# Table of Contents

## Chapter 1: Introducing Create!form Designer .......................... 11
- Who is this Guide For? .............................................. 12
- Create!form Designer Features ................................. 13
- Design Prerequisites ............................................... 14
- Production Prerequisites ........................................ 15
- Getting Help ......................................................... 16
- User Guide Conventions ......................................... 17

## Chapter 2: Installing and Configuring .............................. 19
- Installation Requirements ....................................... 20
- Installing Create!form Designer .............................. 21
- Starting Create!form Designer .............................. 22

## Chapter 3: Getting Started ........................................... 23
- What is a Project? .................................................. 24
- Project Directories ............................................... 25
  - Project Directory ............................................. 25
  - Common Project Directory .................................. 25
- Creating a Project ............................................... 27
  - New Project Wizard ........................................... 27
  - Input File Type ............................................... 28
  - Using an Existing Project as a Template .................... 28
  - DataMap Options .............................................. 29
  - Project Type .................................................. 29
    - Standard Project ......................................... 29
    - Simple Project ............................................ 30
    - Overlay Project ........................................... 30
- Design Flowcharts ............................................... 31
- Saving a Project ................................................. 35
- Opening a Project ............................................... 36
- Printing and Previewing Projects .............................. 37
  - Emulating Production Printing ............................ 38
- Packing Projects ................................................. 39
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transferring Projects to the e-forms Server</td>
<td>40</td>
</tr>
<tr>
<td>Managing Project Files</td>
<td>41</td>
</tr>
<tr>
<td>Chapter 4: Viewing and Navigating</td>
<td>43</td>
</tr>
<tr>
<td>Understanding the Screen Elements</td>
<td>44</td>
</tr>
<tr>
<td>Design Window</td>
<td>46</td>
</tr>
<tr>
<td>Input Window</td>
<td>46</td>
</tr>
<tr>
<td>Project Tree</td>
<td>47</td>
</tr>
<tr>
<td>Input Tree</td>
<td>47</td>
</tr>
<tr>
<td>Thumbnails View</td>
<td>47</td>
</tr>
<tr>
<td>Input File Source View</td>
<td>47</td>
</tr>
<tr>
<td>Status Bar</td>
<td>48</td>
</tr>
<tr>
<td>Navigating in a Project</td>
<td>49</td>
</tr>
<tr>
<td>Jumping Directly to a Set or Page</td>
<td>50</td>
</tr>
<tr>
<td>Moving Between Sections</td>
<td>50</td>
</tr>
<tr>
<td>Chapter 5: Input Design</td>
<td>51</td>
</tr>
<tr>
<td>Before You Start</td>
<td>52</td>
</tr>
<tr>
<td>Using the Create!form Sample Project</td>
<td>52</td>
</tr>
<tr>
<td>Selecting a Different Input File</td>
<td>52</td>
</tr>
<tr>
<td>Viewing the Input File Source</td>
<td>53</td>
</tr>
<tr>
<td>Key Input Design Concepts</td>
<td>54</td>
</tr>
<tr>
<td>What is a DataMap?</td>
<td>54</td>
</tr>
<tr>
<td>What is a Set?</td>
<td>55</td>
</tr>
<tr>
<td>What is a Section?</td>
<td>55</td>
</tr>
<tr>
<td>What is a Data Variable?</td>
<td>55</td>
</tr>
<tr>
<td>What is a Derived Variable?</td>
<td>55</td>
</tr>
<tr>
<td>DataMap Types and Options</td>
<td>57</td>
</tr>
<tr>
<td>Embedded and Shared DataMaps</td>
<td>57</td>
</tr>
<tr>
<td>DataMap Options when Creating a New Project</td>
<td>58</td>
</tr>
<tr>
<td>DataMap Options while Designing a Project</td>
<td>58</td>
</tr>
<tr>
<td>Edit an Embedded DataMap</td>
<td>59</td>
</tr>
<tr>
<td>Copy a DataMap from Another Project</td>
<td>59</td>
</tr>
<tr>
<td>Share a DataMap with Other Projects</td>
<td>59</td>
</tr>
<tr>
<td>Edit a Shared DataMap</td>
<td>60</td>
</tr>
<tr>
<td>Use a Shared DataMap</td>
<td>60</td>
</tr>
<tr>
<td>Embed a DataMap</td>
<td>61</td>
</tr>
</tbody>
</table>
Designing for CSV Input Files ........................................ 62
  CSV File Structure .................................................. 62
  Input File Properties ............................................... 63
  Defining Sets ....................................................... 64
  Defining Sections .................................................. 66
  Working with Data Variables ....................................... 66
    Data Variables in the Input Header Section ....................... 67
    Data Variables in Detail Sections ................................ 67
  Changing the Properties of Columns ................................ 68
  Changing the Input Data Format .................................... 68
  Copying Data Variables to the Project ............................. 68

Designing for Text Input Files ..................................... 70
  Input Sections ....................................................... 72
  Examining How Sections are Populated ............................. 73
  Design Principles .................................................. 73
    Designing from the Top-Down .................................. 73
    Design Flowchart ................................................ 74
  Setting the Input Page Size ....................................... 75
    How the Page Size is Calculated ................................ 77
    Other Input File Settings ........................................ 77
  Defining Sets ........................................................ 80
  Working with Data Variables ....................................... 81
    Set Input Data Formats .......................................... 81
    Create Data Variables .......................................... 82
    Copy Data Variables to the Project .............................. 84
    Create Data Variables in a Detail Section ...................... 85
    Make Data Variables Expandable ................................ 86
  Common Input Section Tasks ....................................... 87
    Display and Edit Input Section Properties ...................... 87
    Controlling When an Input Section is Generated ............... 90
    Working with Multiple Input Sections ........................... 90
    Deleting Input Sections .......................................... 91
  Working with the Input File Header Section ....................... 92
  Working with Input Header Sections ................................ 92
    Adjust the Input Header Section Size ............................ 92
    Make an Input Header Section Expandable ......................... 93
    Add Input Header Sections ....................................... 94
    Rules for Creating Multiple Input Header Sections .......... 96
  Working with Detail Sections ...................................... 96
    About Defining Detail Sections .................................. 97
    Add Detail Sections ............................................... 98
    Make a Detail Section Expandable ................................ 99
    Rules for Creating Multiple Detail Sections ................... 101
  Working with Input Footer Sections ................................ 101
    Adjust the Input Footer Section Size ............................ 101
    Add Input Footer Sections ....................................... 102
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Continuation Header and Footer Sections</td>
<td>136</td>
</tr>
<tr>
<td>Create and Remove Header and Footer Sections</td>
<td>136</td>
</tr>
<tr>
<td>Create Header and Footer Sections from an Input Section</td>
<td>137</td>
</tr>
<tr>
<td>The Display Order of Header and Footer Sections</td>
<td>138</td>
</tr>
<tr>
<td>Alternate Sections</td>
<td>139</td>
</tr>
<tr>
<td>Inserted Sections</td>
<td>141</td>
</tr>
<tr>
<td>Setting Section Properties</td>
<td>142</td>
</tr>
<tr>
<td>Section Positioning Options</td>
<td>142</td>
</tr>
<tr>
<td>Adjusting Section Size</td>
<td>143</td>
</tr>
<tr>
<td>Fixed and Auto-Expanding Sections</td>
<td>143</td>
</tr>
<tr>
<td>About Parent/Child Relationships</td>
<td>144</td>
</tr>
<tr>
<td>Keeping Sections Together</td>
<td>145</td>
</tr>
<tr>
<td>Setting Repagination Control</td>
<td>145</td>
</tr>
<tr>
<td>About Sources</td>
<td>146</td>
</tr>
<tr>
<td>Adding Sources</td>
<td>147</td>
</tr>
<tr>
<td>Chapter 8: Tables</td>
<td>149</td>
</tr>
<tr>
<td>About Tables</td>
<td>150</td>
</tr>
<tr>
<td>Using the Create!form Sample Projects</td>
<td>151</td>
</tr>
<tr>
<td>Working with Tables</td>
<td>152</td>
</tr>
<tr>
<td>Create Tables with the Wizard</td>
<td>152</td>
</tr>
<tr>
<td>Create a Table</td>
<td>152</td>
</tr>
<tr>
<td>Change Table Properties</td>
<td>153</td>
</tr>
<tr>
<td>Delete a Table</td>
<td>154</td>
</tr>
<tr>
<td>Table Properties</td>
<td>155</td>
</tr>
<tr>
<td>Table Levels</td>
<td>155</td>
</tr>
<tr>
<td>Sorting Modes</td>
<td>155</td>
</tr>
<tr>
<td>Header and Footer Sections</td>
<td>156</td>
</tr>
<tr>
<td>Grouping Options</td>
<td>157</td>
</tr>
<tr>
<td>About Parent/Child Relationships</td>
<td>158</td>
</tr>
<tr>
<td>Setting Table Pagination Controls</td>
<td>158</td>
</tr>
<tr>
<td>Creating Borders and Lines over Tables</td>
<td>159</td>
</tr>
<tr>
<td>Chapter 9: Objects</td>
<td>161</td>
</tr>
<tr>
<td>About Objects</td>
<td>162</td>
</tr>
</tbody>
</table>
Chapter 10: Variables ......................................................... 211

About Variables ......................................................... 212
Data Variables and Derived Variables. ......................... 213
User Variables ......................................................... 214
Global User Variables ............................................... 216
System Variables ..................................................... 217
Lookup Variables ...................................................... 219
Archive Variables .................................................... 222
Environment Variables ............................................. 224
Job Ticket Variables ............................................... 225

Chapter 11: Expressions and Conditions ....................... 227

Expressions ............................................................. 228
Building Expressions .............................................. 228
Using the Expression Builder .................................... 228
Navigating and Editing Expressions ......................... 229
Conditions ............................................................. 230
Building Conditions ............................................... 231
Using Expressions as Conditions ............................................. 231
Building Blocks ................................................................. 232
  Boolean Operators .......................................................... 232
  Operators ........................................................................ 233
  Functions ......................................................................... 235
  Variables ........................................................................ 241
  System Conditions ............................................................. 241
Syntax and Order of Precedence Rules ................................. 243
  Mixed Data Types ............................................................. 243
  Addition Operator ............................................................. 244
  Subtraction Operator ......................................................... 244
  Comparing Text Strings ..................................................... 245
  Order of Precedence ......................................................... 245
  Nested Functions .............................................................. 246
  Parsing ............................................................................. 246
  Names of Variables .......................................................... 247
Summarizing Data in Headers and Footers ......................... 248

Chapter 12: Page Styles ......................................................... 251
  About Page Styles ............................................................. 252
  Create Page Styles ........................................................... 253
  Define Page Style Rules ..................................................... 254
  Insert Pages Using Page Styles .......................................... 255
  Attaching Subforms to Inserted Pages ................................. 256
  Create Background Graphics for Page Styles ....................... 257

Chapter 13: Customizing ......................................................... 259
  User Preferences ............................................................. 260
    General Preferences ....................................................... 261
    Section Borders Preferences ........................................ 262
    Input Source Preferences .............................................. 263
    Design Window Preferences ......................................... 263
    Design Window Page Preferences .................................. 265
    Design Window Sections Preferences .............................. 265
    Thumbnails Preferences ............................................... 266
    Input (CSV) Window Preferences ................................... 266
INTRODUCING CREATE!FORM DESIGNER

Create!form Designer is the design component of the Create!form™ Distributed Output Management (DOM) system, which offers advanced output formatting and delivery solutions that allow you to streamline your business communications.

The following topics are covered in this chapter:

• Who is this Guide For?
• Create!form Designer Features
• Design Prerequisites
• Production Prerequisites
• Getting Help
• User Guide Conventions
Who is this Guide For?

This user guide is for people who design projects with Create!form Designer.
Create!form Designer Features

With Create!form Designer you can customize your documents without making modifications in your enterprise server. The following examples introduce some of the features that Create!form Designer offers:

**Invoices**
You can add, remove and reposition data on a page; add graphical elements that improve the look and feel; and change the sort order of tabular data.

**Purchase Orders**
You can compact the data to reduce the overall number of pages, and convert a series of numbers to a barcode.

**Checks**
You can print the bank account number in MICR font and add signatures that are conditional on specified input data values.

**Picking Slips**
You can redesign an invoice as a picking slip, displaying images to represent the parts to be picked.

**Transactional Data**
You can process XML transaction files, displaying the properties of its elements in customized forms.

**Databases**
You can extract and format the contents of any database, either by looking up the records directly in the database, or by processing an extracted CSV data file.
Design Prerequisites

The following are required before you can begin designing your project:

- basic familiarity with Windows concepts
- basic familiarity with word-processing or publishing software
- Adobe Reader version 4 or greater for viewing the user guide and previewing form projects

Refer to Installation Requirements for more information.
Production Prerequisites

Before you can start printing, you must install and configure Create!form Server on the network e-forms server. For more information see the Create!form Server user guide. Depending on your preferred output format, you can also install the Create!archive, Create!fax or Create!email output modules.
Getting Help

Help is available in both Portable Document Format (PDF) and as online help.

To view and print the PDF user guide
- Open the file Create!form Designer 6.3.pdf with Adobe Reader from the \Manuals directory on the installation CD, or
- From the Windows Start menu, select:
  Programs > Create!form > Manuals >Create!form Designer

To view the online help
- Start Create!form Designer and select Help > Help Topics from the Main menu.
User Guide Conventions

Section headings are listed at the front of each chapter to assist you in finding the relevant information. In the electronic version of this document, clicking on these and other cross references will take you to the appropriate section. The following typographical conventions are used:

Table 1.1:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Convention Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bold</td>
<td>Screen items and buttons</td>
</tr>
<tr>
<td><strong>cross reference</strong></td>
<td>Short cut or references to the document, chapter or section</td>
</tr>
<tr>
<td>Numbering 1</td>
<td>Step-by-step procedures. To perform an action, follow these instructions in the sequential order given</td>
</tr>
<tr>
<td>Item &gt; Sub-Item</td>
<td>Menu selection i.e. select the menu Item and then select the Sub-Item from the drop-down list, e.g., File &gt; New</td>
</tr>
<tr>
<td>Note</td>
<td>Additional information relating to the topic</td>
</tr>
</tbody>
</table>
INTRODUCING CREATE!FORM DESIGNER

- User Guide Conventions
INSTALLING AND CONFIGURING

Create!form Designer is installed on a Windows workstation from the Create!form installation CD.

The following topics are covered in this chapter:

• Installation Requirements
• Installing Create!form Designer
• Starting Create!form Designer
Installation Requirements

Create!form Designer requires the following:

- Microsoft Internet Explorer 5.0 or greater
- Adobe Reader 4 or greater (for viewing form project previews and the User Guide)
- Approximately 30 MB of free hard disk space
- A PostScript level 2 or greater printer is recommended
Installing Create!form Designer

For information on upgrading from an earlier version, see the release notes, located in the \Readme directory of the installation CD.

To install Create!form Designer

1. Insert the installation CD. The installation screen should be displayed automatically. If it isn’t, run the CForm.exe program located in the root directory of the installation CD.

2. Select the Install Products button, and then select Create!form Designer from the menu.

3. If no other Create!form product is installed, you will be prompted to select the installation directory. If not, the installation directory is already defined.

4. Specify the default location for saving your project files by specifying the project directory. By default, the WorkDir directory in the Create!form Designer installation directory will be used. If this is the first Create!form product you have installed, it is recommended that you accept the default setting. Refer to Project Directories for more information.

5. The Setup program allows you to choose between a Typical or a Custom installation:
   • Select Typical to automatically install the program, all help files, sample files, and the English-UK and English-US spelling dictionaries.
   • Select Custom to install specific options such as particular spelling dictionaries and Asian fonts.

6. After specifying the location for the program icon on the Start menu, verify that all settings are correct and click Next to complete the installation.

7. Return to the main menu and click Exit when you have finished installing Create!form products.
Starting Create!form Designer

From the Windows Start menu

- Click the Start menu and select Programs > Create!form > Create!form Designer 6

Refer to Chapter Four, Viewing and Navigating for more information on screen elements and toolbars.
Refer to Chapter Three, Getting Started for more information.
GETTING STARTED

Your design is recorded in a project, which contains information about how the input data will be mapped to the design, and the layout of text and graphics on the design page. Each project consists of several files, located in a number of folders, which are required whenever the project is opened, saved, moved or transferred.

The following topics are covered in this chapter:

- What is a Project?
- Project Directories
- Creating a Project
- Design Flowcharts
- Saving a Project
- Opening a Project
- Printing and Previewing Projects
- Packing Projects
- Transferring Projects to the e-forms Server
- Managing Project Files
What is a Project?

A form project (or project) is a record of the way you want to redesign the output of your enterprise server. Because your enterprise server may output a variety of different documents, you will need to create a form project for each of these variations.

A project consists of an input design, which describes the structure of the input file, and an output design, which describes the format and layout of the output form. The input design will be displayed in the Input Window, and the output design will be displayed in the Design Window.
Project Directories

The project files are stored in the following folders:

- Project Directory
- Common Project Directory

The project also uses resources from a number of system folders.

Project Directory

Each project has a project directory. This is the location where you save the project and where its components are stored. When Create!form Designer is installed, a default project directory is created: `<install dir>\WorkDir`. This is where you will be prompted to save all new projects. During the installation or at a later time, you can change this default project directory.

To change the default project directory

1. From the Main menu, select Tools > Preferences. The Preferences dialog will be displayed.
2. Set the Project directory.
3. Click OK.

It is recommended that you use separate directories for design projects and production projects. For example, use \WorkDir for your design work and create a new directory \ProdDir for completed, live projects.

Common Project Directory

Project components that are shared between projects are stored in the common project directory. When Create!form Designer is installed a default common project directory is created: `<Install dir>\CommonProject`. You can specify another location for the common project directory at any time.
To change the common project directory

1. From the Main menu, select Tools > Preferences. The Preferences dialog will be displayed.

2. Set the Common project directory.

3. Click OK.
Creating a Project

Before creating a project, you must choose a suitable input file. The input file should be representative of the files likely to be encountered in a production environment and should contain all possible variations of content and format. The elements and structures of the input file, which you identify during the input design process, are recorded in a DataMap. Refer to Chapter Five, *Input Design*, for more information on creating DataMaps.

The *New Project Wizard* will guide you through the process of creating a new project. You will be given a number of options by the wizard, which may include:

- Input File Type
- Using an Existing Project as a Template
- DataMap Options
- Project Type

To complete your design, follow the steps described in the design flowchart relevant to the input file type you have selected.

New Project Wizard

The New Project wizard will guide you through the process of creating a new project.

**To start the New Project wizard**

1. From the Main menu, select File > New, or click the New Project button. The New Project Wizard dialog will be displayed.

2. From the Type drop-down, select the type of source file from the available file types. Refer to *Input File Type* for more information.

3. From the File/Data source drop-down, select the input source file you wish to open, from the current project directory. If the file you want is not listed, click to display the Select Input File dialog. Locate the input file from this dialog and click Open to select the file.

4. Click *Next*. The sequence and content of dialogs displayed by the wizard will vary depending on the type of input file selected in the first dialog.
5 Continue to follow the wizard instructions and click **Finish** when you have completed all options.

For descriptions of these options refer to:

- Using an Existing Project as a Template
- DataMap Options
- Project Type

---

**Note**

*The project created by the wizard will be configured according to the options you select; you can however change the project by adding or removing features during the design process.*

---

**Input File Type**

The New Project wizard will give you the option of selecting from the available input file types. The input file types supported by Create!form include text, CSV, XML and JDE PDF. Refer to Chapter Five, **Input Design**, for more information.

**Using an Existing Project as a Template**

You can use another project as a template when you create the project with the New Project wizard, or later by selecting the Project > Apply Design Template command. The new project will be an exact copy of the template project including the DataMap and the output design. To successfully apply a template, the input files in both projects must be the same type and have the same structure. When you apply a template to an existing project, the existing DataMap will be unchanged, and the output design will be replaced. To apply an existing DataMap to the new project, use the Project > Change DataMap command. Refer to **DataMap Types and Options** for more information.
DataMap Options

The New Project wizard gives you the option of creating a new DataMap, copying the DataMap from another project, or using a shared DataMap.

Create a New DataMap

Choose the Create a new DataMap option if you have not previously created a DataMap for an input file of this type and structure.

If a suitable DataMap does exist, you can choose to either copy the DataMap, or, if it has already been shared, you can share the DataMap with other projects.

Copy a DataMap from Another Project

Choose the Copy a DataMap from another project option if you need to modify the DataMap in any way.

Select a Shared DataMap

Choose the Select a shared DataMap option if the DataMap can be used without modification.

Refer to DataMap Types and Options for more information.

Project Type

When you are working with a text input file, you can choose the type of project you want to create. The New Project wizard gives you the option to:

- Create a Standard Project
- Create a Simple Project
- Create an Overlay Project

Standard Project

Create a standard project when the input file structure is variable or requires repagination, table sorting, sub-totaling, substantial reformatting, or summary reporting. In a standard project you can utilize Create!form Designer’s many features to map any data from the input file to the project, and then reformat and manipulate the input data in the project.
Output Structure Options
In a standard project you have the option of creating sets and tables in the project.

Set Headers and Footers
Choose the Set headers and footers option if the input file contains groups of pages. For example, a print run might contain multiple invoices for the same customer. The invoices for one customer would form a set. Refer to Defining Sets for more information.

Table with Headers and Footers
Choose the Table with headers and footers option if the body of each page contains repeating line items that you want to sort and subtotal in the project, or if you require floating headers and footers. Refer to Chapter Eight, Tables, for more information.

Simple Project
Create a simple project if only minimal reformatting of the input data is required. In a simple project, one output page is generated for every input page, but you can still selectively map the contents of the input page and then re-position, re-order and reformat as required on the output page.

Overlay Project
Create an overlay project if you don’t want to change the layout, design and formatting of the input file, and need only to add elements such as borders, logos or fixed text to the output pages. In an overlay project the entire input page is mapped directly to the output page.
Design Flowcharts

The following flowcharts list the steps required to create projects for the main input file types.

CSV

Figure 3.1: An overview of the design processes for CSV Input files.
Text

Set the Input Page Size
(refer to Setting the Input Page Size for details)

Define Sets
(see Defining Sets for details)

Create a DataMap for the Input
(see Input Design for details)

Create Data Variables for the Input

Set the Properties and Default Page Styles
(See Project Settings for details)

Organize Sections in the Project
(See Sections for details)

Create Tables for Repeated Data Sections
(See Tables for details)

Add and format Text and Graphics
(See Objects for details)

Figure 3.2: An overview of the design processes for text input files
XML

Figure 3.3: An overview of the design processes for XML input files


**JDE PDF**

Define Sets  
(See Defining Sets for details)

Define Input Sections

Copy Data Variables to the Output

Set the Project Properties and Format Styles  
(See Project Settings for details)

Organize Sections in the Project

Create Tables for Repeated Data Sections

Add and Format Text  
(See Objects for details)

*Figure 3.4: An overview of the design processes for JDE PDF input files*
Saving a Project

You should save your work regularly. You can store projects in any directory, but Create!form Designer looks in only two directories for the files used in each project:

- The <Install dir>\CommonProject directory, which typically contains items common to various projects; or
- A project directory dedicated to particular projects. You can create as many project directories as you want. On install, one project directory exists, called <Install dir>\Workdir.

---

**Note**

*When moving a project to another location, save it as a packed project file type. Refer to Packing Projects for more information.*

---

**To save a project**

From the Main menu, select File > Save, or click the Save icon on the Main toolbar. If you are saving a project for the first time, the Save As dialog will be displayed.

Unless you have selected the Save to common project directory checkbox, Create!form Designer will save the project and the DataMap to the current project directory. Refer to Managing Project Files for more information.
Opening a Project

To view, edit, print or transfer a project, you must open the project with Create!form Designer.

To open a project

1. From the Main menu, select File > Open or click the Open icon from the Main toolbar. The Open dialog will be displayed and show files from the current project directory. To display files in the \CommonProject directory, select the Look in common project directory checkbox.

2. Select the project and click Open.
Printing and Previewing Projects

Projects can be printed locally to a PostScript printer, using the data from the input file. The project can also be previewed on-screen, as it will look when printed, with Adobe Reader. You can emulate your production printing environment by selecting an appropriate merge configuration for local printing. Refer to Emulating Production Printing for more information.

To print a project
1. From the Main menu, select File > Print. The Print dialog will be displayed.
2. Select the required printing options and/or view an on-screen preview:
   - Select a printer.
   - Select the page range and whether to include or ignore Next Copy projects. Refer to Next Copy Project Options for more information.
   - Choose whether to show variable values or names, or whether to hide them.
   - Choose whether you want duplicate copies, or whether you want to use the project settings for duplicates. Refer to Duplicates for information about duplicate settings for next copy projects.
   - Select the production environment you want to emulate from the Merge configuration drop-down. Refer to Emulating Production Printing for more information.
3. Click Print.

Local print settings will not affect how the project behaves in production printing.

To preview a project
From the Main toolbar, click the Print Preview icon or select File > Print Preview from the Main menu.

The settings made on the Print dialog are used when previewing the project. If the preview does not appear as expected, check the settings on the Print dialog.
Emulating Production Printing

You can create merge configurations on the design workstation that emulate the production environment on the e-forms server. The merge configuration contains the same writer properties that you would set up for a Create!form print queue.

Merge configurations are stored in the mergeconfigs.data file in the SystemResource folder, which will be sent to the e-forms server when the project is published. To duplicate the merge configurations on the server, copy mergeconfigs.data from the server into the SystemResource folder on the local machine.

To create a merge configuration

1. From the Main menu, select Tools > Merge Configurations. The Merge Configurations dialog will be displayed. You can also create and edit configurations by selecting Manage from the Merge configuration drop-down on the Print dialog.

2. Click below the last list item, or click the Add Row icon. The Create Merge Configuration dialog will be displayed.

3. Type a name in the Name field.

4. If you have previously created another configuration, which you wish to copy and edit, select the configuration from the Based on drop-down.

5. Click OK. The Merge Configuration dialog will be displayed, which displays either the default configuration settings, or the settings of the configuration you selected above.

6. Select the configuration dialogs from the tree view and set the properties as required. For information on setting properties for a Create!form print queue, refer to the Create!form Server user guide and help.

7. Click OK to return to the Merge Configurations dialog.

8. Continue creating and editing configurations, or click OK to finish.
Packing Projects

Because projects are made up of several different files (the project file, input data file, DataMap, subforms and format styles), Create!form Designer allows you to save all associated files into a single packed project. The entire project can then be easily moved and opened on any computer.

To save a project as a packed project

1. Open the project.
2. From the Main menu, select File > Save As. The Save As dialog will be displayed.
3. In the Save as type field, select Packed form project files (*.fpp6).
4. Click Save. The packed project is saved with the specified name, with an .fpp6 extension. To send the packed project, simply email it as an attachment.

To open a packed project, from the Open dialog, select Packed form project files (*.fpp6) from the Files of type field.
Transferring Projects to the e-forms Server

Before the project can be used for production printing, you must transfer the project and its associated resources to the e-forms server.

To transfer a project to the e-forms server
1. From the Main menu, select Tools > Transfer. The Transfer to Server dialog will be displayed.
2. Select the project to be transferred and its destination.

You will find a complete discussion on transferring projects and the options displayed on the Transfer to Server dialog in the Create!form Server user guide.
Managing Project Files

Create!form Designer creates and uses a number of different file types that can be identified by their icons and filename extensions:

<table>
<thead>
<tr>
<th>File Type</th>
<th>Icon</th>
<th>Extension</th>
<th>Created By</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project</td>
<td>![cfp6 Icon]</td>
<td>cfp6</td>
<td>File &gt; Save and File &gt; Save As</td>
</tr>
<tr>
<td>Backup</td>
<td></td>
<td>cfp6bak</td>
<td>Refer to General Preferences for more information.</td>
</tr>
<tr>
<td>DataMap</td>
<td>![CSV Icon]</td>
<td>.dmp6 (text)</td>
<td>File &gt; Save and File &gt; Save As</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.cmp6 (CSV)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>.xmp6 (XML)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>.jmp6 (JDE PDF)</td>
<td></td>
</tr>
<tr>
<td>DataMap</td>
<td></td>
<td>dmp6bak etc.</td>
<td>Refer to General Preferences for more information.</td>
</tr>
<tr>
<td>Backup</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Packed project</td>
<td></td>
<td>fpp6</td>
<td>Refer to Packing Projects for more information.</td>
</tr>
<tr>
<td>Subform</td>
<td>![cff6 Icon]</td>
<td>cff6</td>
<td>Tools &gt; Create Subform &gt; Design New</td>
</tr>
<tr>
<td>Imported subform</td>
<td></td>
<td>eps</td>
<td>Tools &gt; Create Subform &gt; Import</td>
</tr>
<tr>
<td>Input File</td>
<td></td>
<td>various</td>
<td>external resource</td>
</tr>
</tbody>
</table>
The project files may be located in either the project directory or the common project directory.

Care should be taken when deleting, moving or renaming project files as this may corrupt the project. Whenever you move, copy or send a project to another location, you should pack the project first.
VIEWING AND NAVIGATING

Create!form Designer allows you to interact with and navigate through your projects with a familiar, graphical and easy to use interface, which can be customized to suit your needs.

The following topics are covered in this chapter:

• Understanding the Screen Elements
• Navigating in a Project
Understanding the Screen Elements

The Create!form Designer interface can be customized to suit your needs. The following graphic shows the basic screen elements.
Understanding the Screen Elements

Figure 4.1: The Create!form Designer screen

Note that the appearance and position of each element may vary, depending on screen size and the current view and customization settings.

Create!form Designer
The following topics describe the function of the different windows and views in detail.

**Design Window**

The Design Window has two viewing modes, the Graphics View and the Listing View. To switch between them, click the Graphics View or the Listing View on the Main toolbar. You can also select these commands from the View menu. The sections of the project are displayed as tabs at the bottom of the Design Window.

The Graphics View displays what the project will look like when you print it, and allows you to select, edit and manipulate objects. The Graphics View of the Design Window is where you will perform most of your project design tasks.

The Listing View is a non-graphical representation of all objects in the selected section, along with key information about each object. The Listing View enables you to easily identify objects when the Graphics View is crowded. You can select, edit and work with all objects in the Listing View in the same way as the Graphics View.

You can change the sort order of the Listing View by clicking on the column heading you wish to sort by. If the X-Origin or Y-Origin is displayed in red text, it indicates that part (or all) of the object is off the page layout area.

Refer to Design Window Preferences and Design Window Sections Preferences to customize the look and feel of the Design Window.

**Input Window**

The Input Window displays the input file. When you float the cursor over data in the input file (text and JDE PDF input files only), the name of the data variable will be displayed. This can help you to select the correct data variable when copying to the Design Window.

To show or hide the Input Window, click the Input Window on the Main toolbar or select View/Input Window from the Main menu. Several task shortcuts are available by right-clicking on items in the Input Window.

Refer to User Preferences to customize the look and feel of the Input Window.
Project Tree

The Project Tree displays the logical structure of the project. It contains a list of all sections and all objects, including their properties. Several task shortcuts are available by double-clicking and right-clicking on items in the Project Tree.

To show or hide the Project Tree, click the Project Tree on the Main toolbar or select View > Project Tree from the Main menu.

Input Tree

The Input Tree displays the logical structure of the input file. It provides details about the input file as a whole and individual sections within it. Several task shortcuts are available by double-clicking and right-clicking on items in the Input Tree.

To show or hide the Input Tree, click the Input Tree on the Main toolbar or select View > Input Tree from the Main menu.

Thumbnails View

The Thumbnails View provides a thumbnail of each page of the project, allowing you to scan the formatting of multiple pages.

To show or hide the Thumbnails View, click the Thumbnails on the Main toolbar or select View > Thumbnails from the Main menu.

Refer to Thumbnails Preferences to customize the look and feel of the Thumbnail View.

Input File Source View

The Input File Source View displays the raw input file.

To show or hide the Input File Source View, select View > Input File Source from the Main menu.
Status Bar

The Status bar provides a variety of information about the project as a whole and the current location within the project.

Figure 4.2: The Status bar

Shortcuts are available by double-clicking on the section and project tabs.
Navigating in a Project

The section tabs on the Status bar only appear for sections that are displayed on the current page of the active window. To locate occurrences of some sections using the tabs, you may need to browse the pages of the project.

To navigate to other pages or sections of the project

Click the appropriate button from the Navigation toolbar. The action will be applied to the active window.

These commands are also available through the main menu.

By default, when you browse to a particular section in the Design or Input Window, the corresponding section in the other window will be displayed. Refer to Input (CSV) Window Preferences to turn this synchronization off and on.

Figure 4.3: The buttons on the Navigation toolbar
Jumping Directly to a Set or Page

To go to a specified page number
1 From the Main menu, select Page > Go To. The Go To Page dialog will be displayed.
2 Enter the page number to go to, or select the page number from the drop-down.
3 Click OK. The page selected will be displayed.

To go to a specified set number
1 From the Main menu, select Set > Go To. The Go To Set dialog will be displayed.
2 Enter the set number to go to, or select the set number from the drop-down.
3 Click OK. The first page of the set selected will be displayed.

Moving Between Sections

In order to edit and format sections, they must be selected. You can select sections in the following ways:

- Click on the section.
- Press the arrow keys to move up or down one section in the current window.
- Press the Tab key to move to the next occurrence of that section and Shift+Tab to move to the previous occurrence of that section.
- Click on the section tab on the Status bar.
- All sections that are displayed on the current page will have tabs.
- Select the section in the Project Tree and the Input Tree.
- This is the only way you can edit the properties of a section if it is not set to display.
- Use the buttons on the Navigation toolbar.
INPUT DESIGN

The input design process involves identifying and labelling the different parts or elements of the input file. The way in which you identify, label and organize the data in the input file is recorded in the DataMap.

The following topics are covered in this chapter:

• Before You Start
• Key Input Design Concepts
• DataMap Types and Options
• Designing for CSV Input Files
• Designing for Text Input Files
• Designing for XML Input Files
• Designing for JDE PDF Input Files
• Creating Derived Variables
Before You Start

The following topics contain information about some basic procedures which you might find useful:

- Using the Create!form Sample Project
- Selecting a Different Input File
- Viewing the Input File Source

Before proceeding to the details of designing for a particular file type, you must also be familiar with:

- Key Input Design Concepts
- DataMap Types and Options

Using the Create!form Sample Project

Sample input files have been provided with the installed software so that you can practice performing the tasks described in this User guide. The samples can be found in the `<install dir>\CF6Samples\Tutorials` folder. You can view the text, CSV and XML sample files with any text editor. You should complete each task in sequence as later tasks may assume that earlier tasks have already been completed.

Selecting a Different Input File

If the input file you chose for the input design is unsuitable, or if you want to test your input design on another sample file, you can select a new input file at any time.

Note

If the schema for the XML file has changed, or it has new elements or attributes, we recommend creating a new project and importing the previous project as a design template.
To select a different input file

1. From the Main menu, select Project > Select Input File. The Select Input File dialog will be displayed.
2. Navigate to the folder where the file is located and select the new input file.
3. Click Open. The project will refresh with the new data.

Viewing the Input File Source

To view the input files source, select the View > Input File Source command from the main menu.
**Key Input Design Concepts**

Before you begin your design, you must be familiar with the following key concepts:

- **What is a DataMap?**
- **What is a Set?**
- **What is a Section?**
- **What is a Data Variable?**
- **What is a Derived Variable?**

**What is a DataMap?**

The input design process involves identifying and labelling the different parts or elements of the input file such as header data, line items and footer text. The input file may be an unstructured text file, for which you must create rules that identify each part of the file, or the file may be structured like the CSV and XML formats, which are largely self-defining, but may still require some editing or relabelling. The way in which you identify, label and organize the data in the input file is recorded in the DataMap. You must define a DataMap for every project; however, where the input file structure is the same, you can share the DataMap between several projects.

The information in the DataMap will include:

- The properties of the input file
- How sets are defined
- How sections are defined
- How data variables are defined
- How derived variables are defined

Refer to [DataMap Types and Options](#) for more information about creating, editing and sharing DataMaps.
**What is a Set?**

Where the input file contains sequential related pages or blocks of data, such as account statements for multiple customers, you can group these pages or data together as sets.

During production printing, each set can be treated as a self-contained document. Defining sets also allows you to make use of various pre-defined system conditions, which can be used to apply different formats, or insert extra pages at the start and end of each set.

**What is a Section?**

A section describes a repeated group of data in the input file. For example, in a text file, the rows that contain header text at the top of each page will form a section, while in a CSV file, each row will form a section.

By describing all the different types of data in the input file in this way, the information contained in each section can be more easily reorganized and reformatted in the project.

**What is a Data Variable?**

Input sections can be composed of several different types of information, for example, the input header sections might contain names, addresses, dates, and various reference numbers. A data variable specifies a particular field of data in each section.

Input data cannot be displayed or used in the project unless the data has been identified as a data variable in the input file. The process of defining variables will vary according to the type of input file you are processing.

**What is a Derived Variable?**

You can combine and manipulate data variables in user defined expressions that are called derived variables. Examples of how derived variables can be used include:

- To retrieve a substring of characters from a data variable
Key Input Design Concepts

- To concatenate two or more data variables into one string of characters
- To convert the format of dates
- To find abbreviations and substitute with the expanded text
- To derive values from an arithmetic formula using one or more data variables

Derived variables will appear as data variables in the project. Refer to Creating Derived Variables for more information about the properties of derived variables and how they are defined.
DataMap Types and Options

You can use either an embedded DataMap, or share a DataMap with other projects, which can be selected when you create the project, or later while designing the project.

What would you like to know about?

- Embedded and Shared DataMaps
- DataMap Options when Creating a New Project
- DataMap Options while Designing a Project

Embedded and Shared DataMaps

A DataMap can be either:

- **Embedded**
  An embedded DataMap is associated with a single project, and can be edited and saved without reference to other projects. When you create a new DataMap from another project, the DataMap status is set to be embedded.

- **Shared**
  A shared DataMap can be used by multiple projects. Any changes made to a shared DataMap will affect every project that it uses.

To confirm the status and name of a DataMap

1. From the Main menu, select the Project > Change DataMap command. The Change DataMap dialog will be displayed.
2. The status of the DataMap is displayed, and
   - If the status is shared, the name of the shared DataMap is also shown.
   - If the status is embedded, the name of the DataMap can be determined by adding the appropriate extension to the stem of the project name e.g. if the input file type is CSV, the project MyProject.cfp6 creates an embedded CSV type DataMap with the name MyProject.cmp6. Refer to Managing Project Files for more information.
DataMap Options when Creating a New Project

The New Project wizard gives you the option to:

- **Create a New DataMap**
  Choose this option if you have not previously created a DataMap for the type of input file you are working with.

- **Copy a DataMap from Another Project**
  Choose this option if you have previously created a DataMap in another project for the same input file, which you want to modify for the new project.

- **Select a Shared DataMap**
  Choose this option if you are able to use a DataMap from another project without modification. You must share the DataMap before it can be selected. Refer to Share a DataMap with Other Projects for more information.

DataMap Options while Designing a Project

In an existing project, you can edit the DataMap or change the status of the DataMap in several ways:

**What do you want to do?**

- Edit an Embedded DataMap
- Copy a DataMap from Another Project
- Share a DataMap with Other Projects
- Edit a Shared DataMap
- Use a Shared DataMap
- Embed a DataMap

**Note**

*You must exercise care when changing the DataMap in an existing project, to ensure that the input sections, data variables and derived variables defined in the new DataMap are compatible with your existing design.*
Edit an Embedded DataMap

You can edit an embedded DataMap at any time. The DataMap will change whenever the input design is changed.

Copy a DataMap from Another Project

You can copy a DataMap from another project.

**To copy a DataMap**

1. From the Main menu, select the Project>Change DataMap command.
2. The Change DataMap dialog will be displayed.
3. Select the Copy from another DataMap option.
4. Click Select and locate the DataMap file you want to copy.
5. Click OK.

The existing DataMap will be overwritten and the status remains embedded.

Share a DataMap with Other Projects

You can share a DataMap with other projects.

**To share a DataMap**

1. From the Main menu, select the Project > Change DataMap command. The Change DataMap dialog will be displayed.
2. Select the Share this DataMap as option.
3. The name of the shared DataMap is displayed, and can be edited.
4. Click OK.

The DataMap will be moved to the \CommonProject directory and renamed. The DataMap status will be changed to shared.


Edit a Shared DataMap

You can edit a shared DataMap from any project that uses the DataMap. A warning will be displayed when you open additional projects that share the same DataMap.

When you edit a shared DataMap, the changes will be applied to all projects that use the same DataMap. A warning will be displayed before the edited DataMap is applied to other open projects.

Use a Shared DataMap

Before you can use a shared DataMap, you must first share the DataMap from the original project. Refer to Share a DataMap with Other Projects for more information.

To use a shared DataMap
1. From the Main menu, select the Project > Change DataMap command.
2. The Change DataMap dialog will be displayed.
3. Select the Select a shared DataMap option.
4. Click Select and locate the DataMap file you want to use. Shared DataMaps are stored in either the project directory or the common project directory. Refer to Project Directories for more information.
5. Click OK.

The DataMap status will be changed to shared.

To select a different shared DataMap
1. From the Main menu, select the Project > Change DataMap command. The Change DataMap dialog will be displayed.
2. Select the Select a different shared DataMap option.
3. Click Select and locate the DataMap file you want to use. Shared DataMaps are stored in the \CommonProject directory.
4. Click OK.
Embed a DataMap

In a project that uses a shared DataMap, you can embed either the shared DataMap, or a DataMap from another project.

To embed a DataMap

1. From the Main menu, select the Project > Change DataMap command. The Change DataMap dialog will be displayed.
2. Select one of the following options:
   • Embed DataMap into project
   • Embed copy of another DataMap into project
3. If selecting another DataMap, click Select and locate the DataMap file you want to use.
4. Click OK.

The DataMap is copied into the Project Directory and named according to the project stem and DataMap type. The DataMap status will be changed to embedded. Refer to Managing Project Files for more information.
Designing for CSV Input Files

The input design process for CSV files involves identifying the row and column structure of the CSV input file; this information is recorded in the DataMap. A DataMap for an CSV input file is denoted by the .cmp6 file name extension.

What would you like to know about?

- CSV File Structure
- Input File Properties
- Defining Sets
- Defining Sections
- Working with Data Variables

To create the CSV sample project

1. To start the wizard, click the New Project button on the Main toolbar.
2. From the Type drop-down, select CSV from the available file types.
3. Click Browse to display the Select Input File dialog and select the file Chapter5Sample.csv from the <install dir>\CF6Samples\Tutorials folder.
4. Click Next.
5. Select the Create a new DataMap radio button and click Next. A new DataMap will be created.
6. Ensure the Set headers and footers checkbox and the Table with headers and footers checkbox are cleared and click Next.
7. Click Finish. The new sample project will be displayed in the Input Window.

CSV File Structure

A CSV file contains repeated rows of data, with each field in a row separated by a particular character called the delimiter. Typically fields are separated by a comma, but other characters can be used, including spaces and tabs. The file may also contain a header section, and a column heading row. The important
elements of a CSV file are demonstrated in sample comma delimited file below:

![Figure 5.1: CSV input file structure](image)

Note that text values that contain the delimiter character are enclosed by quotation marks, which is referred to as the text qualifier.

**Paginated CSV Files**

If the CSV input file is paginated and contains page header and footer sections, you must treat the input as a text file, in which you will be able to define the page length and separately identify the detail sections from the header and footer sections. In a text input file, you can create a data variable that contains the entire data row, and then use the retrieve function to extract the delimited field values as user variables. Refer to Designing for Text Input Files for more information.

**Input File Properties**

The input file properties determine how the structure of the CSV input file will be interpreted.

**To change the input file properties**

1. Select the Input Window by clicking anywhere inside the input window frame.
2. From the Main menu, select Input > File Properties. The Input File Properties dialog will be displayed.
Designing for CSV Input Files

3 In the Delimiter section, select a delimiter character and a text qualifier.

4 In the Header section, select the number of header rows and indicate whether the file contains a column heading row.

5 In the Data Format section, select the format used for date and decimal values in the input file.

6 Click OK.

The changes will be applied to the input file and the input window will be refreshed.

Defining Sets

You can use the following means to determine when a new set starts:

- The entire input file can be treated as one set (default setting)
- When the value in one or more columns changes
- Or one set per row
In addition, you can now add conditional settings for each new set, using three options to define Child Tag values.

To define sets
1. From the Main menu, select Set > Define.
2. On the Define Set screen, select a desired tag:
   - Treat entire spool file as one set
   - When values change in marked columns:
     - Select one or more columns

3 (optional) In the Conditional set logic section, you can also configure additional settings contingent on Child Tags values.

- In the Condition set logic section, click Conditionally define sets to activate the Child Tags field box.
- Highlight the Child Tag you want to work with and select one of the desired radio button options. (For example, if you are working with an Invoice set, you might want to select invoice_number and click the “When value changes” radio button).

4 Click OK.

**Defining Sections**

Sections are created automatically from the structure defined by the Input File Properties dialog:

- **InputFileHeader**
  The optional section at the top of the file that contains header text

- **Detail**
  The repeating rows in the body of the file that contain the delimited data

The sections created are displayed in the Input Window.

**Working with Data Variables**

Data variables are created automatically from the structure defined by the Input File Properties dialog. The data variables created are shown in the Input Tree view:
What would you like to know about?

- Data Variables in the Input Header Section
- Data Variables in Detail Sections
- Changing the Properties of Columns
- Changing the Input Data Format
- Copying Data Variables to the Project

Data Variables in the Input Header Section

In the input header section, a data variable will be created for each header line.

Data Variables in Detail Sections

In detail sections, a data variable will be created for each column. The column heading will be used as the data variable name.
Changing the Properties of Columns

You can change the name, type and other properties of columns that define data variables.

To change the properties of a column

1. In the Input Window, click the column heading you wish to edit. The Column Properties dialog will be displayed.
2. To change the column heading and the name of the data variable, type a new name in the Name field.
3. In the Type field, select text, numeric or date as the data type.
4. To repeat the last non-blank value in blank fields, select the Repeat previous value if blank checkbox.
5. Click OK.

Changing the Input Data Format

Create!form Designer interprets dates and numeric data in the input file using the data formats defined by the input file properties. For example, you can change the decimal separator to a comma, or the date order from year-month-day to month-day-year.

To change the input data format

1. From the Main menu, select Input > File Properties. The Input File Properties dialog will be displayed.
2. From the Date order drop-down, select the required date format.
3. From the Decimal symbol drop-down, select the required numeric format.
4. Click OK.

Copying Data Variables to the Project

Data variables defined in the Input Window can be displayed in the project by dragging the data variable into the Design Window.
To copy a data variable to the project

1. In the Input Window, select the data variable.

2. Drag-and-drop the selected variable from the Input Window across into the Design Window.

Refer to Creating Objects for information about adding text variables and other objects to the project.
Designing for Text Input Files

The input design process for text files involves identifying and labelling the different parts or sections of the input file such as header text, line items and footer text. You must create rules that identify each of these parts of the input. These rules are recorded in the DataMap. A DataMap for a text input file is denoted by the .dmp6 file name extension.

The information recorded in the DataMap includes
- Classification of rows on each page into input sections
- Input file page properties
- Grouping of pages as sets
- Mapping of the contents of input sections as data variables
- Definition of derived variables

What would you like to know about?
- Input Sections
- Examining How Sections are Populated
- Design Principles
- Setting the Input Page Size
- Defining Sets
- Working with Data Variables
- Common Input Section Tasks
- Working with Input Header Sections
- Working with Input Footer Sections

To create the text sample project
1 To start the wizard, click the New Project button on the Main toolbar.
2 From the Type drop-down, select Text from the available file types.
3 Click the Open icon to display the Select Input File dialog and select the file Chapter5Sample.txt from the <install dir>\CF6Samples\Tutorials folder.
4 Click Next.
5 Select the Create a new DataMap radio button and click **Next**. A new DataMap will be created.

6 Select the Standard radio button and click **Next**. This option will permit you to fully define the structure of the input file.

7 Ensure the Set headers and footers checkbox is selected, and the Table with headers and footers checkbox is cleared and click Next. Sets will be automatically created in the project. No tables will be created.

8 Click Finish. The new sample project will be displayed in the Input Window.

Refer to **Understanding the Screen Elements** for information about screen elements in the Input Window.
Input Sections

There are four types of sections used to define the structure of the input file:

- **Input file header section** (Refer to Working with the Input File Header Section for more information)
  
  A non-repeating section at the start of the file that contains header text.

- **Input header section** (Refer to Working with the Input File Header Section for more information)
  
  A section at the top of each page that contains header text.

- **Detail section** (Refer to Working with Detail Sections for more information)
  
  Used to define repeating rows within the body of the page.

- **Input footer section** (Refer to Working with Input Footer Sections for more information)
  
  A section at the bottom of each page that contains footer text.

The following graphic displays the way you would define the sections of the sample file that are repeated on each page of the input file. The input file header section occurs once, at the start of the file.

Refer to Display and Edit Input Section Properties for more information.
Examining How Sections are Populated

To assist you in the design process, you can examine all the instances of a section in the input file with the Input > Show All Instances command. In this view mode, all non-blank characters in all instances of the selected section will be marked with an “X”.

Figure 5.6: Examining all instances of a section line in the input file

Design Principles

Create!form Designer can handle a variety of input file structures, from simple tables to complex and irregular forms. This section introduces the design principles that form the foundation for the procedures described later in this chapter.

Designing from the Top-Down

The most important principle, when preparing the input file, is to approach the design from the top-down. As depicted in the Design flowchart, start with the input header section on the first page and make any adjustments needed to the section before looking at the next section. Scan the page from the top-down and locate the first anomaly. Resolve the anomaly, then move down the page to the next anomaly. When the design is correct on the first page, view the second page, again from the top-down. Proceed through the file until the design is complete from start to finish — from top to bottom.
Design Flowchart

The following flowchart describes the main tasks associated with input design.

Define Page Size
(Refer to Setting the Input Page Size for more information)

Define Sets
(Refer to Defining Sets for more information)

Define Input Header Section
(Refer to Working with Input Header Sections for more information)

Define Detail Section
(Refer to Working with Detail Sections for more information.)

Define Input Footer Section
(Refer to Working with Input Footer Sections for more information)

Figure 5.7: Input design flowchart
Setting the Input Page Size

Create!form Designer can handle fixed or variable page lengths, and widths of up to 8192 characters. You can define the input file page length in the following ways:

- By a form feed character (default setting)
- The page has a fixed number of lines
- By a page number
- By a specified character

The input page size is set from the Input File Properties dialog:

![Input File Properties dialog](image)

- Sets the input page length.
- Sets how the end-of-page or start-of-page is identified.
- Only required when a custom end-of-file character is used in the input file.

Refer to Other Input File Settings for more information.

*Figure 5.8: Input page size options on the Input File Properties dialog*
To define the input file page width and length

1. From the Main menu, select Input > File Properties. The Input File Properties dialog will be displayed.

2. In the Lines field, type the page length in lines.

3. In the Columns field, type the line width in characters. If you are unsure of the page length or line width, click the Recalculate button to scan the input file and find the maximum page length and line width in the file. Refer to How the Page Size is Calculated for more information.

4. In the End of page section, select one of the following radio button options:
   - **Form feed**
     A new page will be generated whenever a form feed character is encountered.
   - **Fixed number of lines**
     A new page will be generated according to the Lines field above.
   - **Characters**
     A new page will be generated when the specified characters or character string is encountered.
     In the Characters field, type the character or character string.
     If the string is in a specific location, type the starting position of the string in the at column field; all other occurrences of the string will be ignored.
     Select the appropriate radio button to determine how the character string will be applied:
     select define the first line of a page if the new page begins at the start of the line containing the string;
     select define the last line of the page if the new page begins at the start of the next line following the line containing the string;
     select terminate the page if the new page begins at the next character following the string.

5. Click **OK**.
How the Page Size is Calculated

Once Create!form designer has calculated the page size for a project, it remains unchanged until the Page size fields are edited, or the Recalculate button is selected.

When Creating Standard and Simple Projects

When creating a new standard or simple project, Create!form designer will read the first 10 pages of the file to determine the maximum lines and columns per page. If the number of lines is less than the current default value, the default will be used. If the number of columns is less than the current default value, the default will be used.

When Creating an Overlay Project

When creating a new overlay project, Create!form designer will read the first 10 pages of the file to determine the maximum lines and columns per page. The calculated page size is always used and the default values are ignored.

When Recalculating the Page Size

When you click the Recalculate button, Create!form designer will read the first 10 pages of the file to determine the maximum lines and columns per page. The calculated page size is displayed and the default values are ignored.

To change the default number of lines and columns

1. In the Lines and Columns fields, type the new default values.
2. Review the other settings in the Input File Properties dialog and set default values as appropriate.
3. Click the Set as Default button.

The default settings may be applied when a new standard or simple project is created (see above), or whenever the Reset from Default button is selected.

Other Input File Settings

In addition to page size settings, the following properties are also set from the Input File Properties dialog:

- **Data format** (Refer to Set Input Data Formats for more information)
  Sets the formats used by data variables.
- **Ignore leading blank pages**
  Select to ignore blank pages at the start of the input file.

- **File starts with document header**
  Creates an input file header section (Refer to Working with the Input File Header Section for more information).

- **Ignore bold and underline formatting**
  When processing OS/400 spool files (typically in overlay projects), if the spool file contains bold and underline overstrike control characters, you can choose to either ignore overstriking or convert the overstrike characters into PostScript. Previously this option could only be set in the writer properties on the server.

---

For information on page size settings, refer to Setting the Input Page Size.
To define input file properties

1. From the Main menu, select Input > File Properties. The Input File Properties dialog will be displayed.
2. Select options as required.
3. Click OK.
Defining Sets

Defining sets involves specifying when a new set will start. Typically a new set starts every time the page numbering restarts, but you can also specify that a new set start when data such as the customer number or invoice number changes.

You can use the following means to determine when a new set starts:

- At the beginning of each page in the input file (default setting)
- When the page number in the input file is equal to 1
- When the value of a specified character string on the input page changes
- When a specified condition tests TRUE
- Or, the entire input file can be treated as one set

**To define sets**

1. From the Main menu, select Set > Define. The Define Set dialog will be displayed.
2. Select one of the radio button options to determine when a new set starts:
   - **One set per input page**
• When page number =1 at:
  In the Input Window, click-and-drag a box around the page number. The character position and length of the page number field will be displayed in the Define Set dialog.

• When value changes at:
  In the Input Window, click-and-drag a box around the value. The character position and length of the value field will be displayed in the Define Set dialog.

• On condition:
  Click the Edit button to display the Condition Builder dialog. A new set will start when the condition tests TRUE.

• Treat entire spool file as one set.

3  Click OK.

Working with Data Variables

Set Input Data Formats

Before you start creating data variables, you must set the data formats that Create!form Designer will use to recognize dates and numeric data in the input file. For example, you can change the decimal separator to a comma, or the date order from year-month-day to month-day-year.

To change the input data format

1  From the Main menu, select Input > File Properties. The Input File Properties dialog will be displayed.

2  From the Date order drop-down, select the required date format.

3  From the Decimal symbol drop-down, select the required numeric format.

4  Click OK.
Create Data Variables

To selectively map particular data from the input, for use in the project, you must create data variables around the data.

To create a data variable

1. In the Input Window, select the section containing the data you want.

2. Click-and-drag a box around the area that contains the information you want to store in the variable. The Data Variable Properties dialog will be displayed.

3. In the Name field, type a name for the data variable. The variable name must start with an alphabetic character and then be followed by any combination of alphabetic (a-z, A-Z) and numeric (0-9) characters and underscores. Names are not case sensitive. Naming of variables is entirely at the discretion of the user; however the convention of naming variables to represent the data they contain is recommended. You should think of a name that you will be able to recognize and distinguish from all other data variables in the section.

4. Select a data type from the Type drop-down list.

5. Click **OK**.

6. The area defining the variable will appear highlighted.
Note

*Use the Show All Instances command to examine the extents of a data variable in all instances of the section in the input file. Refer to *Examining How Sections are Populated* for more information.*

---

To change the size and position of a data variable

1. In the Input Window, select the data variable.
2. Click-and-drag the resize handles to change the box size, or click-and-drag inside the box to reposition the box.

Refer to *Selecting and Editing Objects in Graphics View* for more information on Editing Objects.

To display and edit data variable properties

1. Display the Data Variable Properties dialog by doing one of the following:
   - Double-click the data variable in either the Input Window or the Input Tree
   - Right-click the data variable in either the Input Window or the Input Tree and select Properties from the shortcut menu
2 With the Data Variable Properties dialog you can:
   • Change the name of the data variable by typing a new name in the Name field.
   • Select the data type from the Type drop-down. Refer to Set Input Data Formats for more information on how Create!form Designer recognizes date and numeric formats.
   • Change the position and size of the data variable in the Row, Column, Depth and Length fields.
     Row and column positions are defined relative to the origin (top left corner) of the selected section.
   • Select the Trim spaces checkbox to remove leading and trailing spaces from the text string.
   • Select the Expandable checkbox to make the data variable expand with the section. Refer to Make Data Variables Expandable for more information.

To delete a data variable
1 Select the data variable in the Input Window or Input Tree.
2 Press the Del key.

Copy Data Variables to the Project

Data variables defined in the Input Window can be displayed in the project by dragging the data variable into the Design Window.

To copy a data variable to the project
1 In the Input Window, select the data variable.
2 Drag-and-drop the selected variable from the Input Window across into the Design Window.
Figure 5.12: Dragging a data variable into the Design Window

Refer to Adding Text to the Page for more information.

Create Data Variables in a Detail Section

Any data variables created in one occurrence of the detail section will be applied automatically to all other occurrences of the same detail section.

Refer to Create Data Variables for more information.

The area defining the variable will appear highlighted and will be displayed in each occurrence of the same detail section.
Note

Use the Show All Instances command to examine the extents of a data variable in all instances of the section in the input file. Refer to Examining How Sections are Populated for more information.

---

**Figure 5.13: Creating a data variable in a detail section**

**Make Data Variables Expandable**

Data variables defined in expandable sections can be made to expand as the section expands. For example, you could define an expandable data variable on the first row of an expandable detail section so that when the section expands to two or more rows, the data variable will contain the extra rows. An expandable data variable will always expand to the bottom of the section. Refer to Make a Detail Section Expandable and Make an Input Header Section Expandable for more information.
To define an expandable data variable

1. Create a data variable in an unexpanded occurrence of an expandable input section.
2. The Data Variable Properties dialog will be displayed.
3. Type the data variable name in the Name field, and edit other properties as required.
4. Select the Expandable checkbox.
5. Click OK.

Common Input Section Tasks

The following tasks are applicable to all input sections (header, detail and footer).

What do you want to know?

• Display and Edit Input Section Properties
• Controlling When an Input Section is Generated
• Working with Multiple Input Sections
• Deleting Input Sections

For tasks specific to each section type refer to:

• Working with the Input File Header Section
• Working with Input Header Sections
• Working with Detail Sections
• Working with Input Footer Sections

Display and Edit Input Section Properties

Input section properties can be displayed so that you can:

• Change the name of the section
• Change the size of the section
• Control where this type of section is applied in the input file
- Control when this type of section is generated
- Nominate which type of sections can follow this section
- Control the order in which following sections are tested

When you change the properties of a section, the changes are applied to all occurrences of that section in the file.

**To display the Input Section Properties dialog**

Do one of the following:

- With the section active, select Input > Section Properties from the Main menu.
- Double-click the selected section.
- Right-click the selected section and select Section Properties from the shortcut menu.
- Double-click the section tab on the Status bar.
For examples of how to use conditions refer to:
- Add Input Header Sections
- Add Detail Sections
- Add Input Footer Sections

Refer to Chapter 11, Expressions and Conditions, for more information.

Refer to Make an Input Section Expandable and Add Input Header Sections for more information on controlling when a section is generated.

Refer to Controlling When an Input Section is Generated for more information.

Use these controls to add or remove sections from the Followed By list.

Use these controls to change the order in which sections are tested in the Followed By list.

Click to display the sections available to be added to the Followed By list.

The followed by list. Refer to Working with Working with Multiple Input Sections for more information.

Figure 5.14: The Input Section Properties dialog
Controlling When an Input Section is Generated

You can use the Generate Section options in the Input Section Properties dialog to control when an input section is generated in the project. If the section is not generated, the position on the input page where the section would have been displayed is reserved. Changing the Generate Section options will not alter the sequence in which sections are applied to the input page.

To control when an input section is generated

1. In the Input Window, double-click the section. The Input Section Properties dialog will be displayed.

2. Select one of the radio buttons in the Generate Section field:
   - Always — the section will always be generated if valid
   - Never — the section will never be generated even if valid
   - If not empty — the section will only be generated if it contains printable characters

Working with Multiple Input Sections

In cases where input pages begin with different header text, or end with different footer text, or contain different line items, you can define and use additional input sections that accommodate these anomalies. Note that:

- All input header sections are listed in the input headers control list. Refer to Add Input Header Sections for more information.
- All detail sections are listed in the Followed By list on the relevant Input Section Properties dialog.
- All input footer sections are listed in the input footers control list. Refer to Add Input Footer Sections for more information.

For each page, the logic of which section appears where, is summarized in the following flowchart:
Deleting Input Sections

You can delete the selected input section with the Input > Delete Section command. If the section is not visible on the current page, select the section in the Input Tree. You should exercise great care when deleting an input section, as this can disrupt the logic and continuity of how other sections are applied.
Working with the Input File Header Section

You can create an input file header section to contain non-repeating data at the start of the input file.

To create an input file header section

1. From the Main menu, select Input > File Properties. The Input File Properties dialog will be displayed.
2. Select File starts with document header checkbox.
3. Click OK.

To adjust the input file header section size

1. Select the input file header section.
2. Click-and-drag the resize handle on the bottom of the section to reduce or increase the section size.

You can also double-click the section in the Input Window and set the number of rows from the Input Section Properties dialog.

Working with Input Header Sections

What do you want to do?

- Adjust the Input Header Section Size
- Make an Input Header Section Expandable
- Add Input Header Sections

What do you want to know about?

- Rules for Creating Multiple Input Header Sections

Adjust the Input Header Section Size

Typically when you start a new project, the default size of the input header section will be too big or too small for the header text in the input file. You must adjust the input header section so that it contains all the header text and only the header text.
Note

*Use the Show All Instances command to examine all instances of the section in the input file. Refer to Examining How Sections are Populated for more information.*

---

**To adjust the input header section size**

1. Select the input header section.
2. Click-and-drag the resize handle on the bottom of the section to reduce or increase the section size.

**Make an Input Header Section Expandable**

By default, input header sections have a set size. That is, all instances of that section will have the same size regardless of their content. Where the size of the text in an input header section varies between instances, you can make the section dynamically expand around the relevant text. The section becomes “expandable”.

**To make an input header section expandable**

1. In the Input Window, double-click the input header section. The Input Section Properties dialog will be displayed.
2. In the Size field, type the minimum size of the section in rows.
3. Select the Expandable checkbox.
4. Click OK.

The section size will be reset to the minimum size. Select the first detail section below the input header section and identify which data variable can be used to test when this detail section is valid. The data variable will be used to build a condition that tests TRUE in the detail section and FALSE in the expanded rows of the input header section.

5. Double-click the first detail section below the input header section. The **Input Section Properties** dialog will be displayed.
6. Click the Condition Builder icon next to the Condition field. Use the Condition Builder to create a valid condition, using the identified variable, for this detail section. Refer Chapter Eleven, Expressions and Conditions for more information.
7 Click **OK**. The input header section will now automatically expand in size, testing each row below the section, until the condition in a following detail section tests TRUE.

8 Ensure that all sections that are permitted to follow the header section, contain a suitable condition that will distinguish header rows from detail rows.

The sections that are permitted to follow the input header section are listed in the This section can be followed by field on the input header Input Section Properties dialog — also referred to as the Followed By list. The input header section will not expand if any of the sections in the Followed By list do not contain a condition, or the condition always tests TRUE.

---

**Note**

*You can also make data variables expandable in an expandable section.*

---

**Add Input Header Sections**

In cases where the content of the input header section changes between pages, or an expandable header section cannot be used, you can add additional header sections. Only one input header section can be used on a page.

**To create an additional input header section**

1 From the Main menu, select Input > Input Headers. The Input Headers Control List dialog will be displayed.

2 Click the Add Row button. The New Header Section dialog will be displayed.

3 In the Name field, type the new input header section name.

4 In the Size field, type the number of rows required.

5 Using the controls provided, add one or more detail sections to the This section can be followed by list. If there are no sections in the Followed By list, or if the sections listed are not valid when tested, no detail sections will be created on the page.

6 Click **OK** to close the New Header Section dialog. Do not close the Input Headers Control List dialog.
The input headers control list will now have two input header sections defined. Both input header sections are unconditional (the Condition field is empty on the Input Section Properties dialog). If the first input header in the list is unconditional, it will always be created. So the first input header must be conditional. If the condition tests TRUE then the first input header is applied, if the test fails then the second input header is applied.

7 Click the Edit button next to the first input header in the control list. The Input Section Properties dialog will be displayed.

8 Click the Edit button next to the Condition field. Use the Condition Builder to create a valid condition for this input header section. If you wish to use a data variable in the condition, you must create the data variable first. Refer to Chapter Eleven, Expressions and Conditions, for more information.

9 Check that the detail sections in the This section can be followed by list are valid and in the correct order.

10 Click OK to close the Input Section Properties dialog.

11 Click the Edit button on the second (unconditional) input header to display the Input Section Properties dialog.

12 Check that the detail sections in the This section can be followed by list are valid and in the correct order.

13 Click OK to close the Input Section Properties dialog.

14 Click OK to close the Input Headers Control List dialog.

The definition of the new input header section will be applied to the entire file.

Note

The Condition Builder makes available a number of useful system variables and conditions, for example the condition Sys.FirstPageOfDoc will test TRUE on the first page of the document.

To delete an input header section

Care should be taken whenever an input header section is deleted as this will alter not only which input header section is applied, but may also change the way detail sections are applied to the rest of the page. Always review the Followed By lists on the other input header sections, and the order of the input headers control list, before deleting a input header section.
Designing for Text Input Files

1. From the Main menu, select Input > Input Headers.
2. The Input Headers Control List dialog will be displayed.
3. Select the input header section in the control list.
4. Click the Delete Row button.
5. Confirm that you wish to delete this section.
6. Click OK. The change will be applied to the entire file.

Rules for Creating Multiple Input Header Sections

The procedures detailed in Add Input Header Sections can be applied when adding further input header sections. The rules applying to multiple input header sections are:

- The last header in the input headers control list should be unconditional.
- All other headers in the list should be conditional.
- The first header that tests TRUE in the list is applied.
- If none of the headers in the control list test TRUE on a page, then no header will be created on that page.
- Input header sections can be expandable. Refer to Make an Input Header Section Expandable for more information.

The detail section that follows the input header section, will be the first valid detail section in the Followed By list of the input header section.

Working with Detail Sections

What do you want to do?
- Add Detail Sections
- Make a Detail Section Expandable

What do you want to know about?
- About Defining Detail Sections
- Rules for Creating Multiple Detail Sections
About Defining Detail Sections

You can create additional detail sections when the content of line items varies within the page. For example, an invoice may contain subtotal lines, blank lines or description lines in addition to the regular line items. Three different detail section types are shown in the following example.

<table>
<thead>
<tr>
<th>Type</th>
<th>Line Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type 1</td>
<td>APT103 Bamboo Blinds</td>
<td>1020.00 10 10200.00</td>
</tr>
<tr>
<td>Type 1</td>
<td>AT1C39 Umbrellas</td>
<td>30.00 80 2400.00</td>
</tr>
<tr>
<td>Type 2</td>
<td>*** 9 units shipped, balance with next order</td>
<td></td>
</tr>
<tr>
<td>Type 1</td>
<td>APT103 Black Dragon Vase</td>
<td>199.00 10 1990.00</td>
</tr>
<tr>
<td>Type 2</td>
<td>*** hold for collection at Nunawading warehouse</td>
<td></td>
</tr>
<tr>
<td>Type 3</td>
<td>Subtotal</td>
<td>12590.00</td>
</tr>
<tr>
<td>Type 1</td>
<td>APT103 Tea Chests</td>
<td>799.00 4 3196.00</td>
</tr>
<tr>
<td>Type 1</td>
<td>APT103 Black Dragon Vase</td>
<td>199.00 10 1990.00</td>
</tr>
<tr>
<td>Type 2</td>
<td>*** replacement of returned items</td>
<td>***</td>
</tr>
<tr>
<td>Type 2</td>
<td>***</td>
<td>***</td>
</tr>
<tr>
<td>Type 1</td>
<td>AT1C78 Cane Baskets</td>
<td>50.00 5 250.00</td>
</tr>
<tr>
<td>Type 1</td>
<td>AT1C38 Black Dragon Set</td>
<td>1600.00 1 1600.00</td>
</tr>
<tr>
<td>Type 3</td>
<td>Subtotal</td>
<td>7036.00</td>
</tr>
</tbody>
</table>

Figure 5.16: An example of irregular line items

The details of lines are as follows

- Type 1 is the primary line item containing the item details, quantity and price
- Type 2 is a descriptive line relating to the previous type 1 line
- Type 3 is a subtotal line

In particular note that:

- type 1 lines are not always followed by type 2 lines
- type 2 lines can only follow type 1 lines
- type 2 lines always contain “***” at the start of each line
- type 2 can be one or more lines
- type 3 is always two lines
- type 3 lines can follow either type 1 or 2 lines
- type 3 lines can occur at any position
• type 3 lines always contain the text string “Subtotal”

These observations form a set of rules that define the input file structure. The process of building a DataMap involves translating these observations or rules into section properties. You must create a new section for each of the line types you observe in the file.

---

**Note**

Use the Show All Instances command to examine all instances of the section in the input file. Refer to Examining How Sections are Populated for more information.

---

### Add Detail Sections

There are many possible ways of creating DataMaps for the same input file. The type of sections defined will depend not only on the content of the input file, but also on how the input data will be used in the project.

**To define additional detail sections**

1. Identify the first instance where a new section is required.
2. Select the section immediately before (above) where the new section is required.
3. From the Main menu, select Input > New Section After. The New Data Section dialog will be displayed.
4. In the Name field, type the new section name.
5. In the Size field, type the size of the section in rows.
6. Click **OK**. The new section will be displayed in the Input Window.
7. Select the new section.
8. Create the data variables required to define the data in the new section. Refer to Create Data Variables in a Detail Section for more information.
9. Create any additional data variables that are required to assist in distinguishing this section from all other detail sections. You only need to create additional data variables if the data variables created in the previous step are not suitable. For example, you might test when a particular character position is blank, or when a character string contains a particular substring, or a combination of tests using Boolean Operators.
10 From the Main menu, select Input > Section Properties. The Input Section Properties dialog will be displayed.

11 Click to display the Condition Builder dialog and create a condition that tests TRUE when this section is valid.

Refer to Chapter Eleven, Expressions and Conditions, for more information.

12 Define the sections that can follow this section in the This section can be followed by field by adding, removing and ordering sections with the controls provided. The order in which the sections appear in this list is very important: the first valid section in this list will be the next section created after this section. Refer to Display and Edit Input Section Properties for more information.

13 Click OK.

The definition of the new section will be applied to the entire file and displayed in the Input Window.

---

**Note**

*It is good design practice to create a positive condition for every detail section you create, even if the condition is not immediately required to identify the section.*

---

**Make a Detail Section Expandable**

In cases where the number of rows in a detail section is variable, e.g. repeated comment lines, you can make the detail section expandable.

**To make a detail section expandable**

1 In the Input Window, double-click the detail section. The Input Section Properties dialog will be displayed.

2 In the Size field, type the minimum size of the section in rows.

3 Select the Expandable checkbox.

4 Click OK.

The section will now automatically expand in size, testing each row below the section, until a condition in a following detail section tests TRUE.
Ensure that all sections that are permitted to follow the section contain a suitable condition.
Note
You can also Make Data Variables Expandable in an expandable section.

Rules for Creating Multiple Detail Sections

The procedures outlined in Add Detail Sections can be applied when adding further detail sections. The rules applying to multiple detail sections are:

- The detail section that follows the input header section, will be the first valid detail section in the Followed By list of the input header section
- The first valid detail section in the previous detail section Followed By list is applied next
- An unconditional detail section will always be applied if tested in the previous detail section Followed By list
- If there are no valid sections in the last Followed By list, no further detail sections will be applied to that page
- Detail sections can be expandable (refer to Make a Detail Section Expandable for more information)

Working with Input Footer Sections

What do you want to do?
- Adjust the Input Footer Section Size
- Add Input Footer Sections

What do you want to know about?
- Rules for Creating Multiple Input Footer Sections

Adjust the Input Footer Section Size

As with the input header section, you must adjust the size of the input footer section so that it contains all footer text and no text that belongs in a detail section.
Note

*Use the Show All Instances command to examine all instances of the section in the input file. Refer to Examining How Sections are Populated for more information.*

To adjust the input footer section size

1. Select the input footer tab on the Status Bar to select and display the input footer section.

2. Click-and-drag the resize handle on the top border of the input footer section to make the section bigger or smaller.

Add Input Footer Sections

In cases where the content of the input footer section changes between pages, you can add and use additional input footer sections. The additional input footer sections will be added to the input footers control list. Only one can be used on each page.

To define additional input footer sections

1. From the Main menu, select Input > Input Footers. The Input Footers Control List dialog will be displayed.

2. Click the Add Row button to display the New Footer Section dialog.

3. In the Name field, type the footer section name.

4. In the Size field, type the number or rows required for the new input footer section.

5. Click OK to close the New Footer Section dialog. Do not close the Input Footers Control List dialog.

The input footers control list now has two input footer sections defined. Both input footer sections are currently unconditional (the Condition field is empty). If the first in the list is unconditional, it will always be created. So the first footer must be conditional. If the condition tests TRUE then the first input footer is applied, if the test fails then the second input footer is applied.

6. Click the Edit button for the first input footer to display the Input Section Properties dialog.
7 In the Condition section, click to display the Condition Builder dialog and create a condition that tests TRUE when this section is valid. Refer to Chapter Eleven, Expressions and Conditions, for more information on creating conditions.

8 Click OK to close the Input Section Properties dialog.

9 Click OK to close the Input Footers Control List dialog.
The definition of the new input footer section will be applied to the entire file.
Rules for Creating Multiple Input Footer Sections

The procedures detailed in Add Input Footer Sections can be applied when adding further input footer sections. The rules applying to multiple input footer sections are:

- The first input footer that tests TRUE in the control list is applied.
- If none of the input footers in the control list test TRUE on a page, then no input footer will be created on that page.
Designing for XML Input Files

The input design process for XML files involves identifying the different elements of the XML input file; this information is recorded in the DataMap. A DataMap for an XML input file is denoted by the .xmp6 file name extension.

What would you like to know about?

- XML File Structure
- Input File Properties
- Defining Sets
- Defining Sections
- Working with Data Variables

To create the XML sample project

1. To start the wizard, click the New Project button on the Main toolbar.
2. From the Type drop-down, select XML from the available file types.
3. Click Browse to display the Select Input File dialog and select the file Chapter5Sample.xml from the <install dir>\CF6Samples\Tutorials folder.
4. Click Next.
5. Select the Create a new DataMap radio button and click Next. A new DataMap will be created.
6. Ensure the Set headers and footers checkbox and the Table with headers and footers checkbox are cleared and click Next.
7. Click Finish. The new sample project will be displayed in the Input Window.

XML File Structure

To be properly interpreted, the XML file should be “well-formed” and adhere to the published standards. An example of a well-formed XML file is shown below:
The elements and structure of the XML file are interpreted according to an underlying set of rules that determine how the XML elements will be applied in the output project. These initial settings can be changed with the following commands:

<table>
<thead>
<tr>
<th>XML Element Type</th>
<th>Line</th>
<th>XML Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>XML declaration</td>
<td>1</td>
<td>&lt;?xml version=&quot;1.0&quot; encoding=&quot;ISO8859-1&quot; ?&gt;</td>
</tr>
<tr>
<td>comment</td>
<td>2</td>
<td>&lt;!-- Member Updates--&gt;</td>
</tr>
<tr>
<td>root tag with attribute</td>
<td>3</td>
<td>&lt;update total=&quot;2&quot;&gt;</td>
</tr>
<tr>
<td>child element start tag with</td>
<td>4</td>
<td>&lt;member count=&quot;1&quot;&gt;</td>
</tr>
<tr>
<td>attribute</td>
<td></td>
<td></td>
</tr>
<tr>
<td>subchild element tag</td>
<td>5</td>
<td>&lt;ID&gt;1984&lt;/ID&gt;</td>
</tr>
<tr>
<td>subchild element start tag</td>
<td>6</td>
<td>&lt;changes&gt;</td>
</tr>
<tr>
<td>nested subchild element tag</td>
<td>7</td>
<td>&lt;street&gt;1667 Elm Street&lt;/street&gt;</td>
</tr>
<tr>
<td>nested subchild element tag</td>
<td>8</td>
<td>&lt;email&gt;<a href="mailto:fred.krueger@craven.com">fred.krueger@craven.com</a>&lt;/email&gt;</td>
</tr>
<tr>
<td>subchild element end tag</td>
<td>9</td>
<td>&lt;/changes&gt;</td>
</tr>
<tr>
<td>child element end tag</td>
<td>10</td>
<td>&lt;/member&gt;</td>
</tr>
<tr>
<td>child element start tag with</td>
<td>11</td>
<td>&lt;member count=&quot;2&quot;&gt;</td>
</tr>
<tr>
<td>attribute</td>
<td></td>
<td></td>
</tr>
<tr>
<td>subchild element tag</td>
<td>12</td>
<td>&lt;ID&gt;1886&lt;/ID&gt;</td>
</tr>
<tr>
<td>subchild element start tag</td>
<td>13</td>
<td>&lt;changes&gt;</td>
</tr>
<tr>
<td>nested subchild element tag</td>
<td>14</td>
<td>&lt;title&gt;Mr&lt;/title&gt;</td>
</tr>
<tr>
<td>nested subchild element tag</td>
<td>15</td>
<td>&lt;first&gt;Edward&lt;/first&gt;</td>
</tr>
<tr>
<td>nested subchild element tag</td>
<td>16</td>
<td>&lt;last&gt;Hyde&lt;/last&gt;</td>
</tr>
<tr>
<td>nested subchild element tag</td>
<td>17</td>
<td>&lt;email&gt;<a href="mailto:henry.jekyll@stevenson.com">henry.jekyll@stevenson.com</a>&lt;/email&gt;</td>
</tr>
<tr>
<td>subchild element end tag</td>
<td>18</td>
<td>&lt;/changes&gt;</td>
</tr>
<tr>
<td>child element end tag</td>
<td>19</td>
<td>&lt;/member&gt;</td>
</tr>
<tr>
<td>end root tag</td>
<td>20</td>
<td>&lt;/update&gt;</td>
</tr>
</tbody>
</table>

To do this
- Change formats of date and numeric data
  Input > File Properties (Refer to Input File Properties for more information.)
- Ignore selected XML tags in source file
  Input > File Properties (Refer to Input File Properties for more information.)
To do this | Select
--- | ---
Change how sets are created | Set > Define (Refer to Defining Sets for more information.)
Change how sections are created | Input > Tag Properties (Refer to Defining Sections for more information.)
Rename a section | Input > Tag Properties (Refer to Defining Sections for more information.)

Input File Properties

The input file properties determine the input format for dates and numbers, and which tags will be ignored when the input file is processed.

To change the input file properties

1. Select the Input Window by clicking anywhere inside the input window frame.
2. From the Main menu, select Input > File Properties. The Input File Properties dialog will be displayed.
3 In the Data Format section, select the format used for date and decimal values in the input file.

4 In the Ignored Tags section, use the toolbar buttons to add the tags you want to ignore or remove the tags you want to include.

5 Click OK.

The changes will be applied to the input file and the input window will be refreshed.

Figure 5.17: XML Input File Properties dialog
Defining Sets

Defining sets involves specifying when a new set will commence. Typically a new set commences every time the page numbering restarts, but you can also specify that a new set commences when data such as the customer number or invoice number changes.

You can use the following means to determine when a new set starts:

- At the start of each page in the input file (default setting)
- When the page number in the input file is equal to 1
- When the value of a specified character string on the input page changes
- When a specified condition tests TRUE
- The entire input file can be treated as one set

In addition, you can now add conditional settings for each new set, using three options to define Child Tag values.
To define sets
1. From the Main menu, select Set > Define. The Define Set dialog will be displayed.
2. On the Define Set screen, select the desired text:
• Treat entire spool file as one set
• When values change in marked columns:
  Select one or more columns
• When the value contains a certain character

3 (optional) In the Conditional set logic section, you can also configure additional settings contingent on Child Tags values.

4 In the Condition set logic section, click Conditionally define sets to activate the Child Tags field box.

5 Highlight the Child Tag you want to work with and select one of the desired radio button options. (For example, if you are working with an Invoice set, you might want to select invoice_number and click the “When value changes” radio button).

6 Click OK.

Defining Sections

You can select one or more tags to indicate where a new data section will be created.

To define sections
1 In the Input Window, select a tag that defines the start of a new section.
2 From the Main menu, select Input > Tag Properties. The Tag Properties dialog will be displayed.
3 Select the Create section checkbox.
4 In the Name field, type the name of the section.
5 Click OK.

Follow the same procedure to unselect section tags.

To rename a section
1 In the Input Window, select the tag that defines the start of the section.
2 From the Main menu, select Input > Tag Properties.
3 The Tag Properties dialog will be displayed.
Designing for XML Input Files

4 Type the new name in the Name field.
5 Click OK.

Working with Data Variables

Data variables are created automatically from the tag attributes and data elements in the input file. The data variables created are shown in the Input Tree view:

![Input Tree](image)

*Figure 5.18: Data variables created from the sample XML input file*

What would you like to know about?
- Changing the Properties of Data Variables
• Changing the Input Data Format
• Copying Data Variables to the Project

Changing the Properties of Data Variables

To change the properties of data variables

1 In the Input Window, double-click the data variable. The Data Variable Properties dialog will be displayed.

2 With the Data Variable Properties dialog you can:
   • Change the name of the data variable by typing a new name in the Name field.
   • Change the source of the data variable by selecting a different tag from the Data source drop-down.
   • Select the data type from the Type drop-down. Refer to Changing the Input Data Format for information on how Create!form Designer recognizes date and numeric formats.

3 Click OK.

![Data Variable Properties dialog](image)

Figure 5.19: Data Variable Properties dialog
Changing the Input Data Format

Create!form Designer interprets dates and numeric data in the input file using the data formats defined by the input file properties. For example, you can change the decimal separator to a comma, or the date order from year-month-day to month-day-year.

**To change the input data format**

1. From the Main menu, select Input > File Properties.
2. The Input File Properties dialog will be displayed.
3. From the Date order drop-down, select the required date format.
4. From the Decimal symbol drop-down, select the required numeric format.
5. Click OK.

Copying Data Variables to the Project

Data variables can be displayed in the project by dragging the data variable from either the Input Window, or the Input Tree, into the Design Window.

**To copy a data variable to the project**

1. In the Input Window, select the data variable.
2. Drag-and-drop the selected variable from the Input Window across into the Design Window.
Figure 5.20: Dragging a data variable into the Design Window

Refer to Creating Objects for more information on adding text variables and other objects to the project.
Designing for JDE PDF Input Files

The J.D. Edwards PDF format is a self-defining proprietary PDF format generated by PeopleSoft EnterpriseOne. The contents of the PDF file are pre-mapped, which Create!form Designer automatically translates into sections and data variables.

The input design process for JDE PDF files involves defining how sets are created and mapping data variables to the output; this information is recorded in the DataMap. A DataMap for an JDE PDF input file is denoted by the .jmp6 file name extension.

To view the input file elements mapped by the DataMap

- Click the Input Tree on the Main toolbar or select View > Input Tree from the Main menu.

What would you like to know?

- Input File Properties
- Defining Sets
- Input Sections
- Data Variables

Input File Properties

Non-breakable input sections and the date order format are defined by the input file properties. Non-breakable sections are typically used to define page header and footer sections that cannot break over the end of the page when preceded by repeated or broken data sections. The section types that are normally non-breakable are included in the default list of non-breakable sections; these include JDE-PDF section types PH, PF, LF and LH.

To add and remove non-breakable input sections

1. Select the Input Window by clicking anywhere inside the input window frame.
2. From the Main menu, select Input > File Properties.
3 To add section to the Non-breakable sections, click the Add Row button, click the drop-down button on the new row, and select the section from the list.

4 To remove a section, select the section in the Non-breakable sections list, and click the Delete Row button.

5 Click OK.

**To over-ride the input file date format defined in the input file**

1 Select the Input Window by clicking anywhere inside the input window frame.

2 From the Main menu, select Input > File Properties.

3 Select the required format from the Date order field.

4 Click OK.

**Defining Sets**

Defining sets involves specifying when a new set will start. Typically a new set starts every time the page numbering restarts, but you can also specify that a new set start when data such as the customer number or invoice number changes.

You can use the following means to determine when a new set starts:

- At the beginning of each page in the input file (default setting)
- When the page number in the input file is equal to 1
- When the value of a specified character string on the input page changes
- When a specified condition tests TRUE
- The entire input file can be treated as one set

In addition, you can now add conditional settings for each new set, using three options to define Child Tag values.
To define sets

1. From the Main menu, select Set > Define. The Define Set dialog will be displayed.

2. On the Define Set screen, select a desired tag:
   - Treat entire spool file as one set.
   - When values change in marked columns:
     Select one or more columns.
   - One set per row
3 (optional) In the Conditional set logic section, you can also configure additional settings contingent on Child Tags values.

4 In the Condition set logic section, click Conditionally define sets to activate the Child Tags field box.

5 Highlight the Child Tag you want to work with and select one of the desired radio button options. (For example, if you are working with an Invoice set, you might want to select invoice_number and click the “When value changes” radio button).

6 Click OK.

**Input Sections**

Input sections are created automatically from the pre-mapped contents of the input file. The sections defined in the input file are shown in the Input Tree view. The properties of input sections in a JDE PDF input file cannot be changed.

**Data Variables**

Data variables are created automatically from the pre-mapped contents of the input file. The data variables are shown in the Input Tree view. The properties of data variables in a JDE PDF input file cannot be changed.

**To copy a data variable to the project**

1 In the Input Window, select the data variable.

2 Drag-and-drop the selected variable from the Input Window across into the Design Window. Refer to Creating Objects for more information on adding text variables and other objects to the project.
Creating Derived Variables

Derived variables are expressions that are used to combine and manipulate the values of data variables. To create a derived variable, select Input > Derived Variables from the Main menu to display the Derived Variables dialog.

Examples

<table>
<thead>
<tr>
<th>Derived Variables</th>
<th>Expressions</th>
</tr>
</thead>
<tbody>
<tr>
<td>TotalPrice</td>
<td>(UnitPrice+SalesTax)*Qty</td>
</tr>
<tr>
<td>LastChar</td>
<td>substring(ProdDesc,length(ProdDesc),1)</td>
</tr>
<tr>
<td>AddressLine1</td>
<td>Title+FirstName+LastName</td>
</tr>
<tr>
<td>DaysInMonth</td>
<td>Day(date(year(Sys.Today),month(Sys.Today)+1,1)-1)</td>
</tr>
</tbody>
</table>
**PROJECT SETTINGS**

Before you start designing your project, you should review and edit the project properties, the default page style and the project background; otherwise, you may have to redesign your project if these settings are changed later.

The following topics are covered in this chapter:

- Project Properties
- The Default Page Style
- Project Background
Project Properties

Project properties include various print options and default styles, which apply to the whole project.

**What do you want to know about?**
- Collate Options
- Duplicates
- Next Copy Project Options
- About Format Styles

**To edit project properties**
1. From the Main menu, select Project > Properties. The Project Properties dialog will be displayed.
2. Define the collate options. Refer to Collate Options for more information.
3. Define the duplicate options. Refer to Duplicates for more information.
4. Define the Next Copy project, if required. Refer to Next Copy Project Options for more information.
5. Create format styles. Refer to About Format Styles for more information.
6. In the Pages per sheet field, select the number of pages you want to be printed on each sheet. Refer to The Default Page Style for information about setting the page size and orientation.
7. Click OK to save the project properties.

**Collate Options**

There are three ways to order the output of pages:
- Uncollate
- Collate
- (collate) By Set

The dynamic image on the Project Properties dialog gives a graphical representation of the selected collate option.
**Duplicates**

By specifying a value greater than 1 in the Number of duplicates field, you can set the project to process the input file multiple times. By using this option, you can generate multiple copies at the printer, rather than sending all copies from the server.

**Duplicates and Next Copy Projects**

When duplicates are specified in the Project Properties dialog in the first project in the next copy project chain, the duplicates setting is applied to the first project only.

When duplicates are specified in the Print dialog of the first project, the duplicates setting is applied to all projects in the next copy chain.

---

**Note**

*You can also specify that the spool file is processed multiple times using Create!form Director. For more information see the Create!form Director User Guide and the Create!form Server User Guide. The collate options apply to multiple copies, regardless of how the number of copies is specified.*

---

**Next Copy Project Options**

After the input file is processed, you can select that it is processed again using another project, called a Next Copy project. In this way, a chain of projects can be defined. Next Copy projects are used to emulate multi-part printing.

Using the Next Copy project function also allows you to provide summary information of a batch run via a project that produces summary information.

**DataMaps and Next Copy Projects**

Only projects with shared DataMaps can have Next Copy projects, and both the parent and the Next Copy project must use the same shared DataMap. When you make a change to a DataMap in a Next Copy sequence, the change will be applied to all projects.
About Format Styles

You can create format styles that customize the way numbers, dates and currencies are displayed. Format styles allow you to specify such things as:

- Whether a period (.) or comma (,) is used as a decimal symbol
- The currency symbol
- Whether the date will be displayed as day-month-year or month-day-year

Format styles are global for Create!form Designer. Once you have created a style you can apply it in any project.

When you install Create!form Designer, some format styles are automatically created (including the appropriate one for the regional settings of your computer). You can customize these format styles or create new ones.

Create a Format Style

To Create a format style

1. In the Project Properties dialog, select Manage Formats from the Default format style drop-down. The Manage Format Styles dialog will be displayed.

2. Click Create. The Create Format Style dialog will be displayed.

3. Type a name for the style.

4. Choose whether to base the initial style settings on:
   - A Windows locale, and choose a locale
   - A previously defined format style

The locale selected will determine the regional sort mode. Refer to Sorting Modes for more information.

Note

Next Copy projects always adopt the collate and pages per sheet setting of the original project.
Click **OK**. The <name of format style> dialog will be displayed.

Make all required settings for numbers, currency and dates and click **OK**. The Manage Format Styles dialog will be displayed.

When you have finished managing your styles, click Close to return to the Project Properties dialog.

Click **OK** to return to the project.

You can also manage format styles with the Tools > Format Styles command.

### Customize a Format Style

**To customize a format style**

1. In the Project Properties dialog, select Manage Formats from the Default format style drop-down. The Manage Format Styles dialog will be displayed.

2. Select the format style you want to customize and click Edit.

3. Make the necessary changes and return to the project. All objects using the edited format style will be updated.

### Set the Default Format Style for New Objects

When you set the default, it will be the default for all objects created in that project. When you change the default, all objects using the (Project Default) format style will be updated.

**To set the default format style for new objects**

1. In the Project Properties dialog, select the format you want as the default for all new objects in the Default format style field.

2. Click **OK**.

Refer to **Text Properties** for more information on applying different format styles to particular objects.
The Default Page Style

The Page Setup dialog provides the default page settings for the following options:

- Page size
- Orientation
- Duplex settings
- Tray selection
- Inserted pages

These settings will apply to all pages in the project unless you create multiple page styles and apply them conditionally. Refer to About Page Styles for more information.

To set the default page style

1. From the Main menu, select File > Page Setup. The Page Setup dialog will be displayed.

2. In the Page size field, select a pre-defined size from the drop-down list, or choose Custom and specify the Width and Height. The unit of measure will be that specified in your preferences. Refer to Design Window Preferences for more information.

3. In the Orientation section, choose Portrait or Landscape (not available for custom pages).

4. In the Page margins section, type the page and gutter margin dimensions. The unit of measure will be that specified in your preferences. Refer to Design Window Preferences for more information.

5. Set the Tray selection for your printer.

6. In the Duplex section, select whether you want duplex printing and, if so:
   - select the Short edge or Long edge radio button to set the duplex printing style.
   - select the Start on front of page checkbox to start printing the page on the front of the sheet. (Typically used in a conditional page style; for an example of how to control which page is printed on the front or back of a sheet refer to Define Page Style Rules.)
• select the Allow sets to start on back of page to allow printing the start of each set on the back of a sheet.

7 Select the Insert page before checkbox to insert a page before pages with this style. Refer to Insert Pages Using Page Styles for more information.

8 Select the Insert page after checkbox to insert a page after pages with this style.

9 Clear the Include in page numbering checkbox to omit pages with this style from page numbering. (This option is used where you are using multiple page styles.)

---

**Note**

*Changing the page size or orientation will cause objects to be re-positioned. You must manually reposition objects that are affected.*

---

**Tray Selection**

The Input Tray field allows you to specify which printer input tray the paper will be selected from.

The Output Tray field allows you to specify which printer tray the printed paper will be sent to.

The options available from the drop-downs will depend on the options installed on your printer.

---

**Note**

*For details on how to modify the list of available trays and/or media types, refer to Media Selector.pdf available in the \Readme directory of the installation CD.*

---

If the page size is incompatible with the tray’s paper size, the actual results will depend on the printer’s built-in recovery procedures.
Testing with your printer

Due to the variety of printers on the market, you may need to test various selections to see which trays are actually used by your printer. You can re-map the tray selection by creating a Device Properties file specific to a printer device. For detailed information about creating Device Properties Files, refer to Printer Dictionaries.pdf located in the \Readme directory of the installation CD.
Project Background

The page view consists of two superimposed layers:

- **Background Layer**
  Contains objects that do not change with the contents of the page e.g. logo, background graphic and letterhead.

- **Design Layer**
  Contains all variable content e.g. variable text, variable barcode and variable subform objects.

The design layer will overprint the background layer as demonstrated in the diagram below:

*Figure 6.1: Page layers and overprinting*
Background Page Styles

The project background will be displayed on any page that uses The Default Page Style. You can change the background on selected pages by using a different page style. Refer to Create Background Graphics for Page Styles for more information.

Adding Text and Graphics to the Background

To place text, drawing and graphic objects on the background, click the Project Background tab and edit the page. Refer to Chapter Nine, Objects, for more information.

Note

*When you double-click the Project Background tab, the Page Setup dialog will be displayed with which you can alter the page size, orientation and other properties.*
Sections

Sections are defined both in the input and the output. The following topics explain how to create different types of sections in the output.

The following topics are covered in this chapter:

- About Sections
- Data Sections
- Header and Footer Sections
- Alternate Sections
- Inserted Sections
- Setting Section Properties
About Sections

Each type of section displays a particular type of information or occupies a particular space on the page. Sections typically have many occurrences, and where you make layout or format changes to one occurrence, the changes are automatically made to all other occurrences.

In Chapter Five, Input Design, you were introduced to the types of input sections:

- Detail sections, which typically contain information such as line item data
- Input header and input footer sections which typically bracket detail sections

These sections are used to define the input data.

In the project (viewed in the Design Window) you can use the following types of sections to manipulate the display of data:

- Data Sections
- Header and Footer Sections
- Alternate Sections
- Inserted Sections

Using the Create!form Sample Projects

Sample projects have been provided with the installed software so that you can practice performing the tasks described in this user guide. The samples can be found in the <install dir>\CF6Samples\Tutorials folder. You should complete each task in sequence as later tasks may assume that earlier tasks have already been completed.
Data Sections

When you create a project, detail sections in the input file are automatically copied across to the project as data sections; each data section has the same name as the corresponding detail section in the input file.

You cannot delete a data section but you can hide it so it does not display. This is done from the Section Properties dialog.

What do you want to know about?
- Data Section Display Options
- Setting Section Properties

Data Section Display Options

You can specify a data section to display:
- Always
- Never
- On condition

By default, data sections are set to always display.

To set the display properties of a data section

1. From the Main menu, select Section > Properties. The Section Properties dialog will be displayed.
2. Click the Display tab.
3. Select one of the radio button options:
   - Always (to display the section unconditionally)
   - Never (to hide the section unconditionally)
   - On Condition and create the condition (to display the section conditionally). Refer to Chapter Eleven, Expressions and Conditions, for more information.

To prevent data from the section being used in calculations when the section is hidden, select the Ignore data from this section when hidden checkbox.
**Viewing the Properties of Sections that are not Displayed**

If a section is set to never display, or conditionally, and the conditions are never met, the section properties can only be displayed by double-clicking on the section in the Project Tree.
Header and Footer Sections

Creating header and footer sections provides the basic page layout structure and makes designing your project easier.

You can create header and footer sections around the following units:

- The project
- A set
- A page
- A table

Once created, you can format header and footer sections like any other section.

What do you want to know about?

- Continuation Header and Footer Sections
- The Display Order of Header and Footer Sections

What do you want to do?

- Create Set Headers and Footers with the Wizard
- Create and Remove Header and Footer Sections
- Create Header and Footer Sections from an Input Section

Create Set Headers and Footers with the Wizard

When a Standard Project is created, you are given the option of creating set headers and footers. When this option is selected, the first occurrence of the input header section on the first page of the set will be used as the source for the set header. Similarly, the last occurrence of the input footer section in the set will be used as the source for the set footer.
Continuation Header and Footer Sections

As well as header and footer sections, you can also create continuation header and continuation footer sections. Continuation headers and footers appear wherever there is a page break within that header/footer unit. For example a set continuation header appears on every continuation page of a set. Continuation header and footer sections can have the same content and design as the header/footer section, or be totally different.

Refer to Create and Remove Header and Footer Sections for more information.

Create and Remove Header and Footer Sections

This procedure deals with headers and footers for the project, sets and pages. Refer to Header and Footer Sections for information on creating table header and footer sections.

To create header and footer sections for the project, sets or pages
1 From the Main menu, select the appropriate menu option:
   • Project > Headers and Footers
   • Set > Headers and Footers
   • Page > Headers and Footers
2 From the relevant dialog, select the appropriate checkbox to create:
   • A header
   • A continuation header, and if so whether the continuation header will be the same as the header or different
   • A footer
   • A continuation footer, and if so whether the continuation footer will be the same as the footer or different
   • For set headers and footers, you have the option of creating an Additional first page footer. This is useful for things like a payment advice where you may want to print a check on the bottom of the first page
3 To copy all text and graphic objects from another data section, click Create From. Refer to Create Header and Footer Sections from an Input Section for more information.

4 Click OK to close the Header/Footer dialog. The sections you have selected will be created and displayed in the Design Window and the Project Tree. You can now format the newly created sections.

To turn a header or footer section off
Header and footer sections can be turned off (deleted) or turned on (created) at any time. This is done from the relevant headers and footers dialog.

Create Header and Footer Sections from an Input Section

The Create From function lets you copy the contents and size of another section into header and footer sections in the project. When you use this function, the following occurs:

• All text and graphic objects from the specified section and its corresponding original section in the input file are cut and pasted to the new header/footer section. (If objects appear in the specified section and its original section in the input file, Create!form Designer uses the instance from the specified section.)
• If it is bigger than the original, the size of the specified section is adopted by the new header/footer section
• The section you copied the variables and properties from is set to never display

To use the Create From function
1 From the relevant Header/Footer dialog or Table dialog, click Create From. The Create From dialog will be displayed.

2 In the Source field, select the data section from which you want to copy the contents and dimensions.

3 In the Occurrence field, specify the occurrence of the data section you want to map from. For headers, this will usually be the first occurrence, and for footers, the last occurrence.
Click OK to return to the originating dialog.

The Display Order of Header and Footer Sections

It is possible to have multiple header and footer sections on the same page. Their display order is governed by rules. To understand these rules, consider the following hypothetical project that contains:

- Two pages
- One set
- A table that runs over both pages
- Project, set, page and table header/footer sections

The header/footer rules would dictate the following display order:

Table 7.1:

<table>
<thead>
<tr>
<th>Page 1</th>
<th>Page 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>project header</td>
<td>project continuation header</td>
</tr>
<tr>
<td>set header</td>
<td>set continuation header</td>
</tr>
<tr>
<td>page header</td>
<td>page header</td>
</tr>
<tr>
<td>table header</td>
<td>table continuation header</td>
</tr>
<tr>
<td>data section</td>
<td>data section</td>
</tr>
<tr>
<td>table continuation footer</td>
<td>table footer</td>
</tr>
<tr>
<td>page footer</td>
<td>page footer</td>
</tr>
<tr>
<td>set continuation footer</td>
<td>set footer</td>
</tr>
<tr>
<td>project continuation footer</td>
<td>project footer</td>
</tr>
</tbody>
</table>

Create!form Designer
Alternate Sections

Where you have a data section that has a different layout and content in different occurrences, you can create alternate sections to cater for those different occurrences. When certain user-defined conditions are met, the alternate sections will replace the original data section. All alternate sections share the same source and variables as the sections they are alternates for.

There are many possible and equally valid ways of varying the content and format of a section based on its content. Here are some other methods that you may find useful:

- Create multiple detail sections in the input design (Refer to Working with Detail Sections for more information.)
- Conditionally display text and drawing objects (Refer to General Properties)
- Use variable subforms (Refer to Use a variable to Display Different Subforms for more information.)

To create an alternate section

1. Select the data section you want to create an alternate for.
2. From the Main menu, select Section > Alternates. The Alternates <section> dialog will be displayed.
3. To create the condition for displaying the alternate, click the Condition button. (Refer to Conditions for more information on creating conditions.) You can edit this condition at any time. Click OK to return to the Alternates <section> dialog.
4. The Alternate Section Name field contains the default name for the new alternate section. You can type another name in here if you want, but be sure to use one that will clearly associate it with the original data section.
5. If necessary, click the Add Row button to create another condition and another alternate section.
6. If necessary, use the arrow buttons to change the order that Create!form Designer evaluates the conditions.
7. Click OK. Where the conditions are met, the alternate section(s) will replace the data section.
SECTIONS

- Alternate Sections

8 Add data variables (Refer to Adding Text to the Page for more information) and format the alternate section.

To delete an alternate section

1 From the Alternates <section> dialog, select the row that refers to the alternate section you want to delete.

2 Click delete.

3 Click OK. The alternate section will be removed from the project.
Inserted Sections

You can insert one or more sections before and after a detail section. Inserted sections have the following properties:

- Always keeps with the parent detail section
- User variables are defined independently from the parent section
- The primary source is the parent section
- No alternates are permitted

Inserted sections will otherwise act like a normal detail section, and can be edited in the same way.

To insert a section before or after another section

1 In the Design Window, select the section you want to insert the new section before or after.

2 From the Main menu, select Section > Inserted Sections. The Inserted Sections dialog will be displayed.

3 Use the tools provided to add, remove and rearrange inserted sections in the Inserted before <section> and Inserted after <section> list boxes.

4 Click OK.

The Design Window will refresh with the new inserted section(s) displayed.
Setting Section Properties

What do you want to know about?

- Section Positioning Options
- Adjusting Section Size
- Fixed and Auto-Expanding Sections
- About Parent/Child Relationships
- Keeping Sections Together
- Setting Repagination Control
- About Sources
- Adding Sources

To display the Section Properties dialog

- In the Design Window in Graphics View double-click in the active section where there are no objects
- Double-click on the section tab at the bottom of the Design Window
- In the Project Tree, right-click on the section name and select Properties

Section Positioning Options

Depending on the type of section, you may be able to:

- Fix the section in a location on the page
- Float the section immediately after the previous section, regardless of where the previous section is
- Sink the section so that it, along with any sections that follow it, are forced to the bottom of the page

None of these options will change the order of the sections, only their positioning.

The following rules apply to the different section types:

- Data sections, alternate sections and inserted sections can either float or be fixed
• Table header sections can either float or be fixed. Other header sections can only float
• All footer sections can either sink or float
To specify section positioning, click the appropriate button on the General tab of the Section Properties dialog.

To position a fixed section
• In the Design Window, select the section and use the top handle of the section to move it up or down the page.

Adjusting Section Size
To adjust the size of a section do one of the following:
• In the Design Window, drag the bottom handle of the selected section
• Specify the size on the General tab of the Section Properties dialog

Fixed and Auto-Expanding Sections
By default, every occurrence of each section type will be the same size. Alternatively, you can set the following sections to auto-expand:
• Data
• Alternate
• Project headers
• Set headers
• Table headers
• Page headers
When you set a section to auto-expand, you specify that a minimum distance between the bottom of the lowest text object and the bottom of the section will be maintained, regardless of the size of variable objects in the section. If the number of lines in a variable text object increases or decreases, the size of the section will change accordingly.
**To auto-expand a section**

1. On the General tab of the Section Properties dialog, select the Auto-expand checkbox.

2. In the Gap between lowest object and section bottom field, type the minimum distance between the lowest text object and the bottom of the section.

3. In the Size field, specify a minimum overall section size.

4. Click OK.

**About Parent/Child Relationships**

You can define sections as having a parent/child relationship with other sections. A parent section can have multiple child sections. There are several reasons for doing this:

- When you create parent/child relationships, it enables the “keep with” function, so that you can keep sections together, where they might otherwise be forced onto another page.

- To enable the creation of a table involving more than one section type. When the parent is used to define a table, the child section is automatically included in the table. This is the only way a sub-detail section can be included with its parent-detail section in a table.

- From the parent section, you can use section specific variables from the first occurrence of each associated child section that follows the parent.

**Parent/child relationship requirements**

The parent/child relationship will only be preserved if the child follows immediately after the parent or another child of the parent.

All child sections must be set to float.

**Alternate Sections and Parent/Child Relationships**

Alternate sections inherit the parent/child relationships of the section they are alternates for. However the Keep With setting of a parent/child relationship is not inherited.
Keeping Sections Together

Sections can be associated by a parent/child relationship so they keep together. Refer to About Parent/Child Relationships for more information.

Auto-expanding Sections and the Keep With Option

When you set an auto-expanding section to break over the end of a page, the Keep With setting will be ignored.

To define a parent/child relationship

1. Open the Section Properties dialog of the section that will be parent.
2. Click the Repagination tab.
3. For each section you want to be a child:
   - Select the checkbox in the Child Section column
   - Type an alias or leave blank
     The alias is used within expressions and conditions to resolve problems with duplicate names in child or source sections.
4. Select the Keep With checkbox for any children you want to keep with the parent.
5. Click OK.

Setting Repagination Control

To start a new page when a specific section occurs

1. Double-click the section tab to display the Section Properties dialog.
2. Select the Repagination tab.
3. Select the Start new page checkbox.
4. Click OK.

To control repagination of auto-expanding sections

If the section size is auto-expanded, you can specify various pagination controls for the section. Refer to Fixed and Auto-Expanding Sections for more information.
Setting Section Properties

1. Double-click the section tab to display the Section Properties dialog.
2. Select the Repagination tab.
3. To allow the section to break over the page, select the Break auto-expanding section and overflow to next page checkbox.
4. To specify the minimum number of lines kept together at the start or end of a broken text object, select the Orphan control checkbox and type the value in the Minimum lines field.
5. To keep the paragraphs together without splitting, select the Don’t split paragraphs checkbox.
6. Click OK.

About Sources

All data variables, user variables and derived variables used within a section are derived from the section source(s). The section source(s) are other sections. To view a section source(s), open the Source tab of the Section Properties dialog.

The primary source of each section, is the section itself. If you change the name of the section, the change is reflected in the primary source also, but the content remains the same and is a reflection on what type of section it is. The following table explains the content of the primary source for each section type.

<table>
<thead>
<tr>
<th>Section type</th>
<th>Content of primary source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>The corresponding detail section in the input file.</td>
</tr>
<tr>
<td>Alternate</td>
<td>The same primary source as the section it is an alternate for. (The alternate section also inherits any secondary sources that exist in the original section.)</td>
</tr>
<tr>
<td>Inserted</td>
<td>The same primary source as the section it is inserted from. (The inserted section also inherits any secondary sources that exist in the original section.)</td>
</tr>
<tr>
<td>Header/footer</td>
<td>The primary source of a header or footer section in the section itself. You can add source(s) when you are creating the header/footer section using the &quot;create from&quot; function, or from the Source tab.</td>
</tr>
</tbody>
</table>
Adding Sources

At any time you can add data sections from within the project as additional sources. Each additional source can have an alias name used to qualify any data variable used in conditions or expressions in that section. The alias is used within expressions and conditions to resolve problems with duplicate names in child or source sections.

When defining an additional source you must also specify the occurrence. For example, the first occurrence means the source is the first occurrence of that section in the set.

Typically when a section has been defined as a source for another section, to prevent duplication, the source section should be set not to display.

To add a section as a source

1. Open the Section Properties dialog. Refer to Setting Section Properties for more information.
2. Select the Source tab.
3. Click Add Source. A new row of source will be added.
4. From the Section drop-down, select the section you want to add as source.
5. Type an alias or leave blank. The alias is used within expressions and conditions to resolve problems with duplicate names in child or source sections.
6. In the Occurrence column, select which occurrence of the selected section within the set, that you want as source.
7. Click OK.
SECTIONS

- Setting Section Properties
- 
-
Tables

Tables allow you to group repeated data sections so they can be labelled, outlined, sorted and summarized.

The following topics are covered in this chapter

- About Tables
- Working with Tables
- Change Table Properties
About Tables

You can create a table around related repeated sections. Tables provide a robust structure for designing projects. They allow you to:

- Sort data into different groups
- Create header and footer sections for the different groups of data within the table
- Draw table objects, such as lines and boxes, that dynamically fit around the table
- Calculate sub-totals and totals for the different groups of data you have created (Refer to Summarizing Data in Headers and Footers for more information.)

Tables are the best way of handling repeated sections. For example in an invoice project, it is recommended that you define a table for the data sections.

What do you want to know about?

- Working with Tables
- Change Table Properties
Using the Create!form Sample Projects

Sample projects have been provided with the installed software so that you can practice performing the tasks described in this user guide. The samples can be found in the <install dir>\CF6Samples\Tutorials folder. You should complete each task in sequence as later tasks may assume that earlier tasks have already been completed.
Working with Tables

What do you want to do?

- Create Tables with the Wizard
- Create a Table
- Change Table Properties
- Delete a Table

Create Tables with the Wizard

When a Standard Project is created, you are given the option of creating tables. When this option is selected, a table is created for the data sections in each set. The project initially only has one data section defined. The table header and footer positions will change as the set definition is changed.

Create a Table

To create a table

1. Select the section you want to create the table for. If the table will include multiple sections, this will be the parent.

2. From the Main menu, select Table > New. The New Table <section> dialog will be displayed.

3. In the Table name field, type the table name. The table name will form the basis for:
   - Any table header or footer section names
     For example where the table is called “InvoiceData”, the header and footer sections will be called “InvoiceData Header”, “InvoiceData Footer”.
   - The table levels that follow
     For example where the table is called “InvoiceData”, the levels will be called “InvoiceData L1”, “InvoiceData L2” etc.
4 To sort the contents of the table or level, click in the grid or click the Add Row button. A new level will be created. Refer to Table Levels for more information.

In the new row, specify:

• The data variable you want to sort or group by.
• The sort order. If the order is already correct, select None.
• The sort mode. Refer to Sorting Modes for more information.

5 Create headers and footers for the table or a level. Refer to Header and Footer Sections for more information.

6 Select the required grouping options. Refer to Grouping Options for more information.

7 To specify pagination controls for each table level. Refer to Setting Table Pagination Controls for more information.

8 Add further levels.

9 When you have finished adding levels and setting table properties, click OK. In the Design Window, new tabs will be created for each new:

• header section
• footer section
• continuation header section if it is not set to be Same as first header
• continuation footer section if it is not set to be Same as last footer

**Change Table Properties**

To change the properties of a table

1 Click inside any table section.

2 From the Main menu, select Table > Properties. The Table Properties <section> dialog will be displayed.

3 In the Table Name field, type the table name. You can change the name of the table at any time.
4 To change how the table is sorted, click an existing table level or, to add a new level, click the Add Row button. Refer to Table Levels for more information.

In the level row, specify:
- The data variable you want to sort or group by
- The sort order; if the order is already correct, select None
- The sort mode (refer to Sorting Modes for more information)

5 To create or remove headers and footers for the table or a level refer to Header and Footer Sections.

6 Select the required Grouping Options.

7 To specify pagination controls for each table level refer to Setting Table Pagination Controls.

8 When you have finished modifying the table properties, click OK.

Delete a Table

When you delete a table, only the definition of the table and its corresponding header and footer sections is deleted. The data sections the table is built around are preserved in their original state.

To delete a table

1 Select any section contained in the table.

2 From the Main menu, select Table > Delete. A confirmation dialog will be displayed.

3 Click OK to delete the selected table.
Table Properties

What do you want to know about?

• Table Levels
• Sorting Modes
• Header and Footer Sections
• Grouping Options
• About Parent/Child Relationships
• Setting Table Pagination Controls
• Creating Borders and Lines over Tables

Table Levels

Within a table you can create multiple levels. Levels are the mechanism that allows the data to be sorted and grouped.

Refer to Change Table Properties for information on creating a table level.

Examples of how you can use table levels can be found in the following topics:

• Header and Footer Sections
• Grouping Options
• Setting Table Pagination Controls
• Creating Borders and Lines over Tables

Sorting Modes

The following table demonstrates the ASCII, regional and numeric sort modes.
Table 8.1:  

<table>
<thead>
<tr>
<th>Unsor ted Sample</th>
<th>Sort Mode</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(space)</td>
<td>(space)</td>
<td>(space)</td>
<td>(space)</td>
</tr>
<tr>
<td>02</td>
<td>02</td>
<td>02</td>
<td>A</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>1</td>
<td>B</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>10</td>
<td>C</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>a</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>c</td>
</tr>
<tr>
<td>A</td>
<td>A</td>
<td>A</td>
<td>1</td>
</tr>
<tr>
<td>B</td>
<td>B</td>
<td>a</td>
<td>02</td>
</tr>
<tr>
<td>C</td>
<td>C</td>
<td>B</td>
<td>2</td>
</tr>
<tr>
<td>a</td>
<td>a</td>
<td>C</td>
<td>3</td>
</tr>
<tr>
<td>c</td>
<td>c</td>
<td>c</td>
<td>10</td>
</tr>
</tbody>
</table>

The sort order when using regional mode can vary with the Windows locale on which the project’s default format style is based. Refer to About Format Styles for more information.

**Header and Footer Sections**

You can create or remove header and footer sections for the table as a whole and for specific levels within the table.

**To create or remove header and footer sections**

1. In the Table Properties dialog select the level you want to create or remove headers and footers for. Refer to Change Table Properties for more information.

2. Select or clear the appropriate checkbox for:
   - Header before every new grouping in that level
   - Continuation header for a continuation of a group in that level, and if so, whether it will be the same section as the header or different
   - Footer after every grouping in that level
Continuation footer for a continuation of a group in that level, and if so, whether it will be the same section as the footer or different

To create a header or footer section using the content and size of an existing data section, click the Create From button. Refer to Create Header and Footer Sections from an Input Section for more information.

Grouping Options

You can control how sections within a table or table level are grouped and sorted from the following options on the Table Properties dialog. Refer to Change Table Properties for more information.

Selecting the Use only first checkbox and typing a value in the characters for sorting/grouping field can have two functions:

- If you are sorting the data, it will qualify the sorting process to only use the number of characters specified. If you set it to use only the first three characters, it will only sort based on those first three characters of every value
- If you are using headers and footers for the level, it will group the values, based on the number of characters specified

Note

Create a derived variable in the input design if you want to sort or group the data using other criteria.

The example below shows the various sections within a table when you use the group data sections using only the first character and specify level 1 headers and footers.

Table 8.2: An Example of Grouping in Tables

<table>
<thead>
<tr>
<th>Page 1</th>
<th>Page 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table Header</td>
<td>Table Cont. Header</td>
</tr>
<tr>
<td>Level 1 Header</td>
<td>Level 1 Cont. Header</td>
</tr>
<tr>
<td>A........</td>
<td>B........</td>
</tr>
<tr>
<td>A........</td>
<td>Level 1 Footer</td>
</tr>
<tr>
<td>Level 1 Footer</td>
<td>Level 1 Heading</td>
</tr>
</tbody>
</table>
About Parent/Child Relationships

Tables can contain a parent data section and associated children sections. You must create parent/child relationships if:

- the table will contain multiple section types
- you want to keep sections together

Refer to About Parent/Child Relationships for more information.

To create a table containing multiple section types

1. Create a parent/child relationship between the different sections and select the “Keep With” function to keep the child section with the parent section. Refer to Keeping Sections Together for more information.

2. Create a table around the parent. Refer to Create a Table for more information.

Setting Table Pagination Controls

You can specify pagination controls for the table as a whole or individual levels within the table.

To set table pagination controls

1. From the Table Properties dialog, select the level you want to specify pagination controls for.

2. In the Pagination Control section, specify your pagination controls:

Table 8.2: An Example of Grouping in Tables

<table>
<thead>
<tr>
<th>Page 1</th>
<th>Page 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1 Header</td>
<td>C........</td>
</tr>
<tr>
<td>B........</td>
<td>C........</td>
</tr>
<tr>
<td>B........</td>
<td>C........</td>
</tr>
<tr>
<td>B........</td>
<td>C........</td>
</tr>
<tr>
<td>Level 1 Cont. Footer</td>
<td>Level 1 Footer</td>
</tr>
<tr>
<td>Table Cont. Footer</td>
<td>Table Footer</td>
</tr>
</tbody>
</table>
• To start a new page when a new group of the level appears, select the New page checkbox
• To specify a minimum number of occurrences of sections in that level that can appear on the top of the page, select the Top orphan control checkbox, and type the minimum number
• To prevent a group from splitting over a page, select the Keep together checkbox. If the group is too large to fit on a page, it will start on the next page
• To specify a minimum number of sections in the group that can appear at the end of a page, select the Bottom orphan control checkbox and specify the minimum number

Creating Borders and Lines over Tables

To create objects (lines and boxes) that expand and shrink around the contents of a table, you must ensure that:

• The table has header and footer sections bracketing the data you want the dynamic objects around. Refer to Header and Footer Sections for more information.
• When you create the object, the table header section is selected.
• In the object Properties dialog, select the Anchor bottom to footer checkbox. This checkbox will be switched on automatically, if you create the object by clicking and dragging from inside the header to inside the footer. When you do this, the top of the object is anchored to a position relative to the top of the table header section and the bottom of the object is anchored to a position relative to the bottom of the table footer section.

Refer to General Properties for more information.
T A B L E S

- Table Properties

-
You can enhance your project by adding text, graphic, drawing and barcode objects to pages.

The following topics are covered in this chapter:

• About Objects
• Creating Objects
• Editing and Formatting Objects
• Setting Object Properties
• Selecting and Editing Objects in Graphics View
About Objects

Objects can be text, graphic, drawing or barcode. The contents of objects can be static, or you can use variables to control the content of certain objects.

Create!form Designer allows you to create, format and edit objects in many ways; the method you choose will depend on a number of different factors including the type of object you are creating, where the data used to define the object is being sourced from, whether the object is visible, and your own preferences for how you want to work in the Design Window.

Before you begin to create objects, you should be familiar with the following topics:

- About Objects and Sections
- Selecting Objects in Different Views
- Using Variables to Control Objects
- Window Display Preferences

What do you want to know about?

- Creating Objects
- Editing and Formatting Objects
- Setting Object Properties
- Selecting and Editing Objects in Graphics View

Using the Create!form Sample Projects

Sample projects have been provided with the installed software so that you can practice performing the tasks described in this user guide. The samples can be found in the <install dir>\CF6Samples\Tutorials folder. You should complete each task in sequence as later tasks may assume that earlier tasks have already been completed.
About Objects and Sections

When you create an object it will belong to the currently selected section, regardless of where the object is located on the page. You can only select and work with objects belonging to the currently selected section. The object will be displayed in every occurrence of that section. Editing the object in any occurrence of a section changes the object in all occurrences of the same section.

Selecting Objects in Different Views

You can select and edit objects that are visible in the Graphics View directly with the cursor.

When Objects are not Displayed, or are Obscured or Hidden

Alternatively, if the objects are not displayed, or are obscured or hidden, you can select objects in either the Listing View or the Project Tree; both of these views offer a non-graphical list of all objects in a section. Refer to Setting Object Properties for information on how to display and edit the properties of any object from any view, including the position, size, content and format of objects.

When Sections are Not Displayed

If a section is not displayed, the objects it contains can still be edited from the Project Tree. Refer to Edit Objects on Sections that are not Displayed for more information on how to display and edit objects in hidden sections.

Using Variables to Control Objects

You can use variables in a variety of ways to control the content and display of objects on the page, including:

- To provide the content of text objects (refer to Adding Text to the Page for more information)
- To provide data values for barcode objects (refer to Adding Barcodes to the Page for more information)
- To control which image to display (refer to Use a variable to Display Different Subforms for more information)
To control when objects are visible (refer to General Properties for more information)

The types of variables available, and when to use them, are summarized below:

<table>
<thead>
<tr>
<th>Use</th>
<th>To display</th>
</tr>
</thead>
<tbody>
<tr>
<td>data variables</td>
<td>text, number and dates copied directly from input sections</td>
</tr>
<tr>
<td>user variables</td>
<td>text, numbers and dates that manipulated or calculated with functions and other defined expressions</td>
</tr>
<tr>
<td>global user variables</td>
<td>information derived from elsewhere in the project</td>
</tr>
<tr>
<td>system variables</td>
<td>project page numbers, date, time file and user attributes and other system values</td>
</tr>
<tr>
<td>lookup variables</td>
<td>information derived from external databases</td>
</tr>
<tr>
<td>environment variables</td>
<td>information derived from the operating environment</td>
</tr>
</tbody>
</table>

**Window Display Preferences**

You can customize the way objects are edited and displayed by selecting the Tools > Preferences command. With the options displayed in the panes of the Preferences dialog, you can:

- Always display the object Properties dialog whenever an object is created
- Turn on and off rulers and grid
- Change the unit of measure used for object position and size
- Turn on and off outlines of text objects
- Highlight text variables in red
- Show variable names instead of values in text objects
- Control how objects are displayed in selected and unselected sections
Creating Objects

To create an object

1. From the Object toolbar, select the tool for the type of object you want to create.

2. Click-and-drag to create the initial position and size of the object. An object will be created using the default attributes. (Refer to Set Default Properties for more information.)

Object Tools

Table 9.1:

<table>
<thead>
<tr>
<th>Command</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td>Insert static text</td>
</tr>
<tr>
<td></td>
<td>Insert static text containing variables</td>
</tr>
<tr>
<td>Variable Text</td>
<td>Insert a single variable as text</td>
</tr>
<tr>
<td>Subform</td>
<td>Insert an existing subform</td>
</tr>
<tr>
<td></td>
<td>Convert images into subforms</td>
</tr>
<tr>
<td></td>
<td>Create subforms from other objects</td>
</tr>
<tr>
<td>Variable Subform</td>
<td>Use a variable to display different subforms</td>
</tr>
<tr>
<td>Line</td>
<td>Draw lines</td>
</tr>
<tr>
<td>Horz/Vert Line</td>
<td>Draw horizontal and vertical lines</td>
</tr>
<tr>
<td>Box</td>
<td>Draw a box</td>
</tr>
</tbody>
</table>
Adding Text to the Page

There are two types of text objects that you can insert:

- Text objects that can contain both static text and variables
- Variable text objects that can contain only a single variable

Static text is the text that you type directly into the project from the keyboard, whereas variables can contain text, numbers and dates from the input file and other sources. Refer to Using Variables to Control Objects for information on the types of variables you can use.

You can add text and variable text objects to the page in several ways:

By Using the Object Toolbar

- Insert Static Text
- Insert Static Text Containing Variables
- Insert a Single Variable as Text
Creating Objects

- Copy Data Variables from an Input Section

By drag-and-drop from the Input Window
- Drag-and-Drop Variables from the Input Window

Insert Static Text

To insert static text
1. In the Design Window, click on the section where you want to insert the text object.
2. Click the Text button in the Object toolbar. The cursor will change to the text symbol.
3. In the Design Window, click-and-drag the cursor to create a text box. A text box will be displayed in text edit mode.
4. Type in the text you want.
5. Click outside the text object to exit text edit mode. (Refer to Change the Display Properties of any Object and Edit and Format the Contents of a Text Object for more information.)

Insert Static Text Containing Variables

To insert static text containing variables
1. In the Design Window, click on the section where you want to place the text object.
2. Click the Text button in the Object toolbar. The cursor will change to the text symbol.
3. In the Design Window, click-and-drag the cursor to create a text box. A text box will be displayed in text edit mode.
4. Type the required text, and as required, click the Insert Variable button on the Text toolbar. The Insert Variable dialog will be displayed.
5. From the Variable drop-down, select the required variable from the available sources. The Variable drop-down displays all currently available
variables for that section. Refer to About Sources for more information on making variables available from another source.

6  Click OK.

7  Continue typing or add further variables.

8  Click outside the object to exit text edit mode.

Refer to Change the Display Properties of any Object and Edit and Format the Contents of a Text Object for more information.

## Insert a Single Variable as Text

**To insert a single variable as text**

1  In the Design Window, click on the section you want to place the variable text object in.

2  Click the Variable Text button in the Object toolbar. The cursor will change to the variable text symbol.

3  In the Design Window, click-and-drag the cursor to create a text box. A variable text object is created and the Variable Text Properties dialog will be displayed.

4  From the Variable pane, click the Variable drop-down button and select the required variable from the available sources. The Variable drop-down displays all currently available variables for that section. Refer to About Sources for more information.

5  Edit the object properties on the General (Refer to General Properties for more information), Variable (Refer to Variable Properties for more information) and Text panes (Refer to Variable Properties for more information) as required.

6  Click OK.

Refer to Format a Variable Text Object for more information.

## Copy Data Variables from an Input Section

Instead of inserting one variable at a time, you can copy all variables that have been defined in the corresponding section of the input file, or all variables that have been defined in the project. A variable text object will be created for each data variable that is not already copied to the project.
**To copy all variables in a section**

1. In the Design Window, click on the section corresponding to the input section where the variables are located.

2. Do one of the following:
   - From the Main menu, select Section > Copy Data Variables
   - Right-click on the section to display the shortcut menu and select Copy Data Variables

**To copy all variables in the project**

- From the Main menu, select Project > Copy Data Variables

Once the variables have been copied, you can reposition, resize and reformat as required on the page. Refer to Format a Variable Text Object for more information.

---

**Note**

*You can synchronize the display in the Input Window with selections in the Design Window.*

---

**Drag-and-Drop Variables from the Input Window**

You can create variable text objects by using the cursor to drag-and-drop data variables from the Input Window to the Design Window. To copy more than one data variable at a time, select the data variables while holding down the SHIFT key, or drag a box around a group of data variables, and then drag-and-drop the group into the Design Window.
Adding Graphic Objects to the Page

You can create and add graphic objects in several ways:

- Place an image from another program on the page
- Create a graphic object composed of other objects
- Place different images on the page depending on the value of a variable

Images and graphic objects are referred to as subforms.

What do you want to do?

- Insert an Existing Subform
- Convert Images into Subforms
- Create Subforms from Other Objects
- Use a variable to Display Different Subforms
About Subforms

There are two types of subform objects used by Create!form Designer:

- EPS subforms that contain graphic images created by another program, and
- CFF6 subforms that contain other Create!form Designer objects

Subforms are commonly used for items that are repeated in the project or appear in multiple projects such as logos, addresses, signatures and standard images. If a particular graphic appears in several locations in the same project, or in several projects, you only need to create it once. When you modify that subform, all the occurrences where it is used will be automatically updated.

Subform Storage

By default, subforms are stored in the common project directory, however they can also be stored in the current project directory. Refer to General Preferences for information on changing the common project directory and the project directory.

Insert an Existing Subform

1. Click the Subform tool, located in the Object toolbar.
2. Position the cursor on the Design Window and click-and-drag to define the initial subform position. The Subform Properties dialog will be displayed.
3. In the Subform pane, click Select.
Creating Objects

4 Select the subform file (.eps or .cff6) from the common project directory or project directory, and click **Open**.

5 Set other properties as required. (Refer to Setting Object Properties for more information.)

Click **OK**. The subform will be displayed in the project.
Convert Images into Subforms

Image files (.bmp, .jpg, .gif, .dxf and other image formats) must be converted to Encapsulated PostScript (.eps) format before they can be added to the project as subforms. You can either generate your images in .eps format, or use Create!form Designer to convert the image file.

Best results are usually achieved by converting images with Create!form Designer; however if you do create your own .eps files, you should save the file with the lowest compatibility version, while still maintaining quality.

EPS subforms cannot be edited from within Create!form Designer.

To create an EPS subform from an image file

1. Click the Subform tool, located in the Object toolbar.
2. Position the cursor on the Design Window and click-and-drag to define the initial subform position. The Subform Properties dialog will be displayed.
3. In the Subform pane, click Import.
4. Navigate to and select the graphic you want to use.
5. Click Open. The subform will be saved to the common project directory. Clear the Save to common project directory checkbox to save to the project directory. The image will be converted into .eps format.
6. Set other properties as required. (Refer to Setting Object Properties for more information.)
7. Click OK. The subform will be displayed in the project.

To convert multiple images into EPS format

1. From the Main menu, select Tools > Create Subform > Import. The Import dialog will be displayed.
2. Either select or clear the Save to common project directory checkbox. If cleared, the subforms will be saved to the project directory.
3. To add a prefix to each file as it is saved, select the Add prefix checkbox and type the prefix you want; the prefix will help you identify and manage the files in the project folders.
4. Navigate to the folder where the files are stored and select the images you want to convert. Click the Files of type drop down to display the types of
image files that can be converted. You can select multiple files in the same
folder by holding down the CTRL key.

5 If the images are large, you can restrict which files are displayed in the
preview box with the Don’t preview files larger than option.

6 Click Open. Each of the selected images will be converted in turn and saved
to the chosen directory.

The images can then be placed on the page as individual subforms, (Refer to
Insert an Existing Subform for more information) or displayed using a variable
subform object (Refer to Use a variable to Display Different Subforms).

Create Subforms from Other Objects

Subforms that are made of other Create!form Designer objects have the .cff6
file extension.

To create a CFF6 subform

1 Click the Subform tool, located in the Object toolbar.

2 Position the cursor on the Design Window and click-and-drag to define the
initial subform position. The Subform pane of the Subform Properties dialog
will be displayed.

3 In the Subform pane, click Design New. The New Subform dialog will be
displayed.

4 In the Name field, type the subform name and click OK.

5 Use the Create!form Designer tools to create the image or text you want.

6 Select File > Save. The subform file (.cff6) will be saved to the common
project directory.

Close the subform window and return to your project. The subform will be
displayed in the project.

Use a variable to Display Different Subforms

Use the Variable Subform tool from the Object toolbar when you want to
place different subforms on the page depending on the value of a variable.
Refer to Adding Graphic Objects to the Page for more information.
For example, a data variable, which stores the part number for each line item of a product catalog, can be used to display images for each part. If the part number values are, say, AT12P9, AY345W and BRG200F, the variable subform will look for the files AT12P9.eps/cff6, AY345W.eps/cff6 and BRG200F.eps/cff6 in the common project directory and the default project directory, and display the image corresponding to the part number in each occurrence of the data section.

To create a variable subform object

1. Click the Variable Subform tool, located in the Object toolbar.
2. Position the cursor on the Design Window and click-and-drag to define the initial height, width and location for the subform. The Variable Subform Properties dialog will be displayed.
3. From the Variable field, select the variable that will provide the subform name values.
4. Select the Subform Variable pane and click Import to import the image(s) to be converted into .eps format.
5. Set subform properties as required. (Refer to Subform Properties for more information.)
6. Set other properties as required. (Refer to Setting Object Properties for more information.)
7. Click OK.

The variable subform object will be created. If the value in the variable corresponds with the name of a subform, the subform will be displayed.

Adding Drawing Objects to the Page

You can enhance the page with lines, boxes, grids, ellipses and circles using the drawing tools on the Object toolbar.

What do you want to do?

- Draw Lines
- Draw Horizontal and Vertical Lines
- Draw a Box
- Draw a Grid
• Draw Circles and Ellipses

Draw Lines

To Draw lines
1. Click the Line tool, located in the Object toolbar.
2. Position the cursor on the Design Window and click-and-drag to define the start and end points of the line. Hold down the Shift key to draw in 45 degree increments.

Refer to Change the Display Properties of any Object for more information on editing and formatting lines.

Draw Horizontal and Vertical Lines

To draw Horizontal and Vertical Lines
1. Click the Horizontal/Vertical Line tool, located in the Object toolbar.
2. Position the cursor on the Design Window and click and drag to define the start and end points of the line.

Refer to Change the Display Properties of any Object for more information.

Draw a Box

To draw a box
1. Click the Box tool, located in the Object toolbar.
2. Position the cursor on the Design Window and click-and-drag to define any two diagonally opposite corners. Hold down the Shift key to draw a square.

Refer to Change the Display Properties of any Object for more information.
Draw a Grid

To draw a grid

1. Click the Grid tool, located in the Object toolbar.

2. Position the cursor on the Design Window and click-and-drag to define any two diagonally opposite corners.

A 2x2 grid is initially created, which you can change from the Grid pane of the Properties dialog. Refer to Change the Display Properties of any Object for more information.

Draw Circles and Ellipses

To draw circles and ellipses

1. Click the Ellipse tool, located in the Object toolbar.

2. Position the cursor on the Design Window and click-and-drag to define any two diagonally opposite corners of a box containing the ellipse. Hold down the Shift key to draw a circle.

Refer to Change the Display Properties of any Object for more information on editing and formatting ellipses.

Adding Barcodes to the Page

You can define barcodes using either a constant value or variable.

What do you want to do?

- Insert a Barcode with a Constant Value
- Insert a Variable Barcode
Insert a Barcode with a Constant Value

To Insert a barcode with a constant value
1 Click the Barcode tool, located in the Object toolbar.
2 Position the cursor on the Design Window and click-and-drag to define the position of the barcode. The Barcode pane of the Barcode Properties dialog will be displayed.
3 In the Data value field, type the text or numeric value for the barcode.
4 Set the barcode properties as required. Refer to Barcode Properties for more information.
5 Set other properties as required. Refer to Setting Object Properties for more information.
6 Click OK.
The initial size of the barcode is determined by the content and barcode standard used.

Insert a Variable Barcode

To insert a variable barcode
1 Click the Variable Barcode tool, located in the Object toolbar.
2 Position the cursor on the Design Window and click-and-drag to define the position of the barcode. The Variable pane of the Variable Barcode Properties dialog will be displayed.
3 In the Variable field, select a variable to supply the data value from the drop list.
4 Set the barcode properties as required. Refer to Barcode Properties for more information.
5 Set other properties as required. Refer to Setting Object Properties for more information.

6 Click OK.

**Insert a Digital Signature**

When generating PDF output, you can insert a digital signature that will be recognized by Adobe Acrobat/Reader as a PDF signature field.

To insert a digital signature

1 Click the Digital Signature tool, located in the Object toolbar.

2 Position the cursor in the Design Window and click-and-drag to define the position of the signature. The Form Field pane of the Digital Signature Properties dialog will be displayed.

3 In the Field name field, type the name of the signature field that will be used a placeholder tag by Adobe Acrobat.

4 In the Tooltip field, type the text you want displayed in Adobe Acrobat as a tooltip.

5 If you do not want to display the object in Adobe Reader, select the Hidden checkbox.

6 If you want the object to be printed from Adobe Reader, select the Printable checkbox.

7 Set other properties as required. Refer to Setting Object Properties for more information.

8 Click OK.
Copying Objects from Other Resources

You can copy objects already created from existing projects and subforms.

What do you want to do?

- Copy Objects from Other Projects
- Copy Objects from an Existing Subform

Copy Objects from Other Projects

To copy an object from another project, open the project and copy and paste the objects you want from one project document to the other.

Copy Objects from an Existing Subform

If an existing subform file contains objects you want to use in the project, you can copy the objects to the selected section.

To copy objects from an existing subform

1. Select the section where you want to copy the objects to.
2. From the Main menu, select Tools > Copy Objects from. The Copy Objects dialog will be displayed.
3. Select the subform file (*.cff6) containing the objects.
4. Click Open.

All objects contained in the selected subform file will be copied to the current section.

Note

You can also apply the output design from one project to another, which will copy all the objects, sections, tables, variables and page styles designed in that project. Refer to Using an Existing Project as a Template for more information.
Editing and Formatting Objects

When working interactively in the Design Window, a selection of frequently used formatting functions are available from the toolbars. The full range of formatting functions can be accessed from the object Properties dialog. The functions available, and how you find them, are described below.

What do want to do?

- Change the Display Properties of any Object
- Edit and Format the Contents of a Text Object
- Format a Variable Text Object
- Change Number, Date and Currency Formats
- Edit the Contents of a Subform

Change the Display Properties of any Object

The following display options can be accessed from the General pane of the object Properties dialog:

- Origin — displays the position of the object in the section
- Size — displays the horizontal and vertical dimensions of the object
- Scale — displays horizontal and vertical scaling (subforms and barcodes only)
- Description — used to identify the object in other views
- Display condition — applies conditions to when the object is visible
- Rotation — displays the rotation angle of the object

The properties specific to each type of object will be displayed on the Text, Subform, Styles, Grid, Barcode, Variable, Subform variable and Form Field panes of the object Properties dialog, depending on the type of object, or objects, selected. Refer to Setting Object Properties for information on how to display and edit these options.

You can also move, resize, rotate, copy, align, and change the display order of objects interactively; refer to Selecting and Editing Objects in Graphics View for more information.
Edit and Format the Contents of a Text Object

If the text is static, you can edit and format individual characters and words; however for variable text, you can only apply formatting to the entire variable.

**To edit and format the contents of a text object**

1. Select the text object.
2. Click again inside the text object to start text edit mode.
3. Select the characters or words you want to edit, replace or format.
4. Type the new text and use the formatting tools from the Text toolbar to format as required.
5. To format a variable in a text object, select the variable and use the formatting tools from the Text toolbar to format as required.
6. To edit other properties of a variable in a text object, double-click the variable. The Edit Variable dialog will be displayed from which you can change the variable name and edit the properties of the variable, which may include:
   - Format as: — display as text, number, date or currency (Refer to [Change Number, Date and Currency Formats](#) for more information.)
   - Ignore blank lines — removes blank lines from the text
   - Strip leading spaces — trims spaces from start of text string
   - Strip trailing spaces — trims spaces from end of text string
   - Wrap (ignore end of line) — wraps text over multiple lines (refer to [Variable Properties](#) for information on how to use these options)
7. Click **OK** to close the Edit Variable dialog and apply the changes.
8. Click outside the object to exit text edit mode.
To change other text properties
Double-click the object to display the Text Properties dialog. The following options are available from the Text pane, and any changes will be applied to the object as a whole:

- Font — over-rides any formatting you have previously made to the text
- Alignment — text alignment tools
- Line spacing — changes the spacing between lines of text
- Kerning — changes the spacing between characters
- Mirrored text — inverts the display of text
- Format Style — changes how currencies, dates and numbers are displayed (Refer to Variable Properties for more information on how to use these options)

Format a Variable Text Object
Any changes to formatting will be applied to the variable as a whole.

To format a variable text object

1 Select the object.
2 Use the formatting tools from the Text toolbar to format as required.
3 To access the full range of formatting options, double-click the object to display the Variable Text Properties dialog.
4 Select the Text pane to display and edit these properties:
   - Font — text formatting tools
   - Alignment — text alignment tools
   - Line spacing — changes the spacing between lines of text
   - Kerning — changes the spacing between characters
   - Mirrored text — inverts the display of text
   - Format Style — changes how currencies, dates and numbers are displayed.

Refer to Variable Properties for more information.
5 Select the Variable pane to display and edit these properties:

- **Format as** — display as text, number, date or currency (refer to Change Number, Date and Currency Formats for more information)
- **Ignore blank lines** — removes blank lines from the text
- **Strip leading spaces** — trims spaces from start of text string
- **Strip trailing spaces** — trims spaces from end of text string
- **Shrink to object width** — adjusts font size to fit lines to width
- **Wrap (ignore end of line)** — wraps text over multiple lines
- **Keep paragraphs** — preserves paragraph structure of text string (refer to Variable Properties for more information)

**Change Number, Date and Currency Formats**

You can apply customized format styles that control the way numbers, dates and currencies are displayed. For example whether currencies are displayed with a $ or £ symbol, or dates are displayed in month-day-year or day-month-year order. Different format styles can be applied to objects in the same project, allowing you to create output for offices or customers in multiple regions with one design project. Refer to About Format Styles for information on creating and managing format styles.

The following procedures explain how you can format variables as numbers, dates or currencies using a particular format style.

**To select the format style**

1 Double-click the object to display the object Properties dialog.
2 Select the Text properties pane.
3 Select a style from the Format Style drop-down.
4 Click OK.

Refer to Text Properties for more information.
To format a variable text object as a number, date or currency

1. Double click the object to display the object Variable Text Properties dialog.
2. Select the Variable properties pane.
3. Select a format from the Format as: drop-down. Selecting the Text format will leave the input data in its original format.
4. Click OK.

To format a variable in a text object as a number, date or currency

1. Select the text object.
2. Click again inside the text object to start text edit mode.
3. Double-click the variable you want to edit. The Edit Variable dialog will be displayed.
4. Select a format from the Format As: drop-down. Selecting the Text format will leave the input data in its original format.
5. Click OK.
6. Click outside the object to exit text edit mode.

Edit the Contents of a Subform

You can only edit the contents of CFF6 subforms that have been created using Create!form Designer graphic and text objects. Refer to Create Subforms from Other Objects for more information.

To edit a subform

1. Do one of the following:
   - Use the Selection tool to double-click on the subform to display the Subform Properties dialog. In the Subform pane, click Edit. The subform will be displayed.
   - Select File > Open. From the Open dialog, in the Files of type: drop-down, select Subforms (*.cff6). Select the required file(s) and click Open. The subform will be displayed.
2 Use the Create!form Designer drawing tools to edit the subform.

3 Insert variables from the parent form into the subform by creating a variable text field, or by creating a text field and clicking the Insert Variable button in the toolbar. The Variable Text Properties or Insert Variable dialog will be displayed.

4 From the Variable drop-down, select the desired variable from the available sources.

5 Save and close the subform. The changes will be reflected in all projects that use the subform.
Setting Object Properties

What do you want to do?

- Display and Edit Object Properties
- Edit the Properties of Multiple Objects
- Set Default Properties

What do you want to know about?

- About Property Classes

Display and Edit Object Properties

The position, size, rotation, content, format and display properties of objects can be displayed and edited from the <object type> Properties dialog. When you edit and format objects interactively, the content of the object Properties dialog will be updated when the object properties are changed.

You can display the object Properties dialog from the Design Window in either the Graphics View or the Listing View, or from the Project Tree.

To display the object Properties dialog

From any view, do one of the following:

- Double-click on the object
- Right-click on the object and select Properties from the shortcut menu
- Click on the object and select Edit > Properties from the Main menu
- Click on the object and select the Properties button from the Main toolbar

Choose the method that best suits the way you work.
Edit the Properties of Multiple Objects

You can edit the properties of multiple objects, including different object types, at the same time.

To edit the properties of multiple objects

1. Select the objects you want to edit.
2. Click the Properties button from the Main toolbar.
3. The object Properties dialog displays the properties of all selected objects.
4. Edit the properties from the dialog panes displayed.
5. Click OK to apply the changes.

Properties that are common to more than one object will be applied to all objects that share those properties, and properties that are unique to one object will be applied only to that object.
For example, if you simultaneously edit an ellipse, a box and a grid, the object Properties dialog will display the General, Styles and Grid property panes.

![Figure 9.4: Example of editing the properties of multiple objects](image)

### About Property Classes

Objects can belong to several property classes, which can be common to other objects. Each property class will be displayed as a separate pane in the object Properties dialog.

The object Properties dialog displays the property classes for all of the selected objects. For example, if you simultaneously edit a text object, a box and a grid, their properties will be displayed on the General, Styles, Text and Grid property panes:

<table>
<thead>
<tr>
<th>Object Type</th>
<th>General</th>
<th>Variable</th>
<th>Text</th>
<th>Styles</th>
<th>Grid</th>
<th>Subform</th>
<th>Variable Subform</th>
<th>Barcode</th>
<th>FormField</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Variable Text</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subform</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Variable Subform</td>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Line</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 9.2:
### General Properties

The General pane of the object Properties dialog displays the following options that are common to all objects:

<table>
<thead>
<tr>
<th>Object Type</th>
<th>General</th>
<th>Variable</th>
<th>Text</th>
<th>Styles</th>
<th>Grid</th>
<th>Subform</th>
<th>Variable Subform</th>
<th>Barcode</th>
<th>FormField</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horizontal/Vertical Line</td>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Box</td>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Grid</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ellipse</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Barcode</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Variable Barcode</td>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Digital Signature</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>
Setting Object Properties

Origin and Size
You can position objects relative to a section, or to the page. The position and size of the object will be displayed in units of points, cms or inches depending on your specified preference. (Refer to Design Window Preferences for more information.) Object position and size are controlled by the following properties:

- **Origin**
  The object position relative to the section or page origin will be displayed in the X and Y fields, measured from the top left corner of the section, or the page, to the top left corner of the object.

- **Fix to page**
  By default, the object is positioned relative to the section. Select this option to fix the position of the object on the page. Take care not to fix any object that may have more than one occurrence on a page, as it will be redrawn in the same position for every occurrence.

- **Size**
  The object Width and Height will be displayed.

  **Anchor bottom to footer**
  When an object is created in a header section, you can make the object automatically resize; for example, you can draw a box around a table that will adjust to the number of data sections in the table or table level. (Refer to Creating Borders and Lines over Tables for more information.)

Scaling
For subforms and barcodes, you have the option of scaling the size of the object along both the horizontal and the vertical axis. The value of the scale is relative to the original object size when first created. When the object is created, the scaling is set to 1. When you change the scaling value, the size of the object will be factored by the scaling value along each axis respectively. For example, if a subform has an initial size of 2" high by 3" wide, when the scale is changed to 2 on the vertical axis, and 0.5 on the horizontal axis, the size of the subform will change to 4" high by 1.5" wide.

Description
You can type a label in the Description field that will help identify the object in the Listing view mode of the Design Window and the Project Tree.
Display Condition
Objects can be set to display when specified conditions are met. For example, you can create a condition so that a text object will only display when a data variable has a particular value (Refer to Building Conditions for more information.).

Rotation
You can rotate the object by typing an angle in the Rotation field. The angle is shown in degrees anti-clockwise around the origin (top left corner) of the object.

You can also rotate objects using the Rotate button on the Main toolbar and the resulting rotation angle will be displayed in the Rotation field. Note that the Rotate tool applies a rotation angle relative to the current rotated position of the object, which is different to the rotation angle displayed on the General pane of the object Properties dialog, which is always relative to the horizontal axis of the page. For example, if the rotation angle, displayed on the General pane, is initially 90 degrees, and the object is subsequently rotated by -15 degrees with the Rotate tool, the rotation angle, displayed on the General pane, will change to 75 degrees.

Variable Properties
The variable used to control the content of variable text, variable subform and variable barcode objects will be displayed on the Variable pane of the object Properties dialog.

The following additional options are only displayed for variable text objects, and are also available from the Edit Variable dialog (Refer to Edit and Format the Contents of a Text Object for more information.).

Format as
Select whether to treat the variable data as Number, Date or Currency. If you select Text, no format style will be applied.

Ignore blank lines
When selected, blank lines will be removed from the text.

Strip leading spaces
When selected, trims spaces from the start of the text string.

Strip trailing spaces
When selected, trims spaces from the end of the text string.
Wrap (ignore end of line)
When selected, wraps the text over multiple lines according to the object width. If a variable is inserted in a text object and the Wrap (ignore end of line) checkbox is clear, any text that follows the variable will start on a new line.

Keep paragraphs
When selected, preserves new line feeds within the text string.

Text Properties
A selection of text formatting tools are also available from the text toolbar. Refer to Understanding the Screen Elements for more information.

The Text pane of the object Properties dialog displays the following tools and options:

Font Tools

![Font Tools Image]

Alignment Tools

![Alignment Tools Image]

Refer to Advanced Object Selection for more information on aligning objects.
Line Spacing
When the Fixed checkbox is selected, the value in the line spacing field is the spacing, in points, between the bottom of one line and the bottom of the next line. The line spacing interval is independent of font size. A line spacing of "0" makes the lines overprint. When the fixed checkbox is cleared, the value in the Line spacing field is the gap, in points, between the bottom of one line and the top of the next line. The line spacing interval is dependent on font size being the sum of the line spacing value (the gap between lines) and the maximum font size of characters in the line. A line spacing of "0" makes the lines print at the default spacing of the font. A line spacing of "10" adds 10 points to the default line spacing.

Kerning

This is positive kerning
This is normal kerning
This is negative kerning

Mirrored Text
Select this option to display a mirror image of the text.

Format Style
You can apply different format styles to variables in text and variable text objects. Format styles determine how currencies, dates and numbers are displayed. When you apply a format style to an object, it applies to all variables contained in the object.

By default, all new text objects will be created with the current default format style. Refer to About Format Styles for more information.

Select the Manage Formats option from the Format style drop list to create, edit, rename and delete format styles.

Styles Properties
Select the Styles pane of the Properties dialog to display and edit the styles of lines, boxes, ellipses and grids.
Line
Select the line Style, Color and Width.

Fill
To fill the object with color, click the Style drop-down, select Solid, and then click the Color field to choose a color.

Rounded corners
You can round the corners of boxes by selecting the corner or corners, and typing the radius in the Corner radius field.

Grid Properties
Select the Grid pane of the Properties dialog to set the number of columns and rows in a grid object.

Number of Intervals
Type the number of equal width Columns and Rows.

Subform Properties
Refer to Adding Graphic Objects to the Page for more information on creating and using subform objects. The following command buttons are displayed on the Subform pane of the Properties dialog:

Select
Displays the subform files available to be inserted in the project.

Import
Allows you to import one or more image files, and saves the selected images as EPS subforms in the common project directory or the project directory.

Edit
Opens a CFF6 subform for editing in Create!form Designer. Any changes made will be reflected in all projects that use the subform.

Design New
Creates a new CFF6 subform for editing in Create!form Designer.
Subform Variable Properties

**Import**
Allows you to import one or more image files, and saves the selected images as EPS subforms in the common project directory or the project directory.

**Auto fit**
Adjusts the size of the images to fit the width of the variable subform frame.

**Preserve aspect ratio**
When the Auto fit option is active, preserves the aspect ratio of the original image.

**Alignment**
Select the horizontal and vertical alignment of images inside the variable subform frame.

Barcode Properties

The following options are displayed on the Barcode pane of the Properties dialog:

**Check digit**
Select the checkbox to add a check digit to the barcode.

**Letters**
Select the checkbox to display the data value as text below the barcode.

**Line spacing**
Displays the current barcode line spacing. The line spacing is the space between the top of one line of barcode, and the top of the next line of barcode. This spacing is not affected when you vertically scale (resize) the barcode.

**Barcode standard**
Displays the barcode standard. Click the drop list button to display the available standards.

**Data value**
Displays the text or numeric value for the barcode. Alternatively a variable can be selected from the Insert variable drop list. This option is not displayed when viewing the properties of Variable Barcode objects.
Form Field Properties

The following options are displayed on the Form Field pane of the Properties dialog:

Field Name
The Adobe signature field name.

Tooltip
The tooltip displayed in Adobe.

Hidden
Select the checkbox to hide the signature object in Adobe.

Printable
Select the checkbox to make the object printable from Adobe.

The Hidden and Printable settings correspond to the following Adobe signature settings:

Table 9.3:

<table>
<thead>
<tr>
<th>Adobe</th>
<th>Hidden</th>
<th>Printable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visible</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>Visible but does not print</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hidden but printable</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Hidden</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>
Set Default Properties

The following command buttons are displayed on various panes of the object Properties dialog.

Set as Default

Select this button to use the current properties as the default for the current property class. The default properties will be applied when creating new objects, in all projects.

Apply Defaults

Select this button to apply the current default settings for this property class to the selected objects.
Selecting and Editing Objects in Graphics View

What do you want to do?

- Select Objects
- Select Hidden Objects
- Edit Objects on Sections that are not Displayed
- Move Objects
- Resize Objects
- Rotate Objects
- Copy Objects
- Change the Display Order
- Align Objects
- Advanced Object Selection

Select Objects

Use the Selection tool to select one or more objects. (To unselect an object and select the Selection tool, press the Spacebar.)

To select multiple objects
1. Select the first object you want to include in the selection.
2. Hold down the Shift key and click on the next object.
3. Continue to select additional objects in the same way.

To deselect an object
Hold down the Shift key, and click on the selected object. The object will be unselected.
Select Hidden Objects

Objects can be hidden behind each other or difficult to identify because of their content. For example, variable objects may have no content. The following are methods for selecting and identifying these types of object. They all require that the section the object belongs to is selected.

Ctrl+A
Press the Ctrl and A keys together to select all objects in the active section.

Click and drag
If you know the approximate location of the object you are trying to locate, you can use the Selection tool to click-and-drag an area. All objects fully within the area will be selected.

Using the Listing View or Project Tree
The Listing View and the Project Tree both offer a non-graphical list of all objects in a section (Refer to Design Window and Project Tree for more information). By selecting an object in either of these views, the same object is selected in the Graphics View.

Edit Objects on Sections that are not Displayed
You can still view and edit the objects on sections that are not displayed. There can be several reasons why sections may not be displayed in the design:

- The input file does not contain an instance of a particular section
- The conditions for the display of an alternate section are not met
- Continuation headers and footers do not occur

To edit the contents of a section that is not displayed

1 In the Project Tree, right-click the section you want to edit, and select Edit Form from the shortcut menu. The section, and the objects it contains, will be displayed in a separate view pane.

2 Add, edit and delete objects as required.

3 When you have completed your edits, close the section edit pane by clicking the close button at the top right corner of the view pane.
Note

All sections, including possible continuation headers and footers and alternate sections are listed in the Project Tree.

Move Objects

There are several methods of moving objects.

Use the cursor
Use the Selection tool to select and drag an object to the desired location.

- You can constrain movement to increments of 45 degrees from the original location, by holding down the Shift key while dragging
- You can constrain movements to particular points by using snap (refer to Design Window Preferences for more information)

Use the arrow keys
You can move any selected object(s) using the arrow keys.

Use the Move tool
This method is useful where a precise movement is required.

1 Select the object.
2 From the Main toolbar, click the Move Objects button or select Edit > Move from the Main menu. The Move dialog will be displayed.
3 Specify the amount of horizontal and vertical movement that you want. Negative numbers move the object(s) left and up.

You can also
- Set the location of an object from the General pane of the Properties dialog. Refer to General Properties for more information.
Resizing Objects

**To resize one or more objects**

1. Select the object(s) to resize.
2. Position the cursor over one of the object’s sizing handles.
3. Click and drag outwards to increase the size or inwards to decrease the size.
4. Release the mouse button when the object(s) are the size required.

**You can also**

- Set the exact size of an object from the General pane of the Properties dialog (Refer to **General Properties** for more information)
- Use the arrow keys while holding down the SHIFT or CTRL keys to adjust the size of an object
- Make the width and height the same as a control object (Refer to **Align Objects** for more information)

---

**Note**

To resize the object(s) proportionally, hold down the Shift key while dragging a corner handle.

---

Rotate Objects

**To rotate an object**

1. Select the object(s) to rotate.
2. From the Main toolbar click the Rotate button, or select Edit > Rotate from the Main menu. The pointer will change to the rotate pointer.
3. Click on the point you want the object to rotate around (the pivot point). The Rotate dialog will be displayed. The Around fields display the X and Y coordinates of the selected pivot point.
4. Type a rotation angle in degrees in the Rotation Angle field. A negative value will rotate the object in a clockwise direction.
5. Click OK. The object will be rotated around the pivot point.
You can also
Rotate an object from the General pane of the Properties dialog (Refer to General Properties for more information.).

Copy Objects

To copy one or more objects
1 Select the object(s) to copy.
2 From the Main menu, select Edit>Copy, or press Ctrl+C.
3 From the Main menu, select Edit>Paste, or press Ctrl+V. The object(s) will be pasted into the currently selected section in front of, but fractionally offset from the original.

Note
To create a copy of an object and move it at the same time, hold down the Ctrl button and drag a copy of the object away from the original.

Change the Display Order

There are two methods of rearranging the display order of objects:
• Using the order commands, e.g. Send to back, Send to front, common to many graphic design applications.
• Using the paste in relation to a “guide” object method.

Using the order commands and tools
1 Select the object you want to rearrange.
2 Right-click and select Order <command>; or use the order buttons on the Order toolbar.
Using the paste in front or back of object method

1. Cut the object you want to re-arrange.
2. Select the reference object.
3. From the Edit menu or right-click options, select either Paste in Front or Paste in Back. The object will be pasted in front or behind, the object you selected in step 2.

**Align Objects**

The alignment tools enable you to accurately line up objects. In order to use the alignment tools, you must understand the concept of control objects.

The control object

The control object is the object that other objects move around, when you are performing any sort of alignment:

- If you’ve selected the objects using the shift and click method, the control object is the last one selected
- If you’ve selected the objects using the click-and-drag or select all method, the control object is the last object created

Handles around the control object have a thicker black border than the handles on the other selected objects.

**Alignment tools**

The following table explains the alignment tools when used with fully selected objects. These tools will perform different functions when used with the Handle Selection tool. Refer to Advanced Object Selection for more information.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Left</td>
<td>Aligns the left sides of all objects with the left side of the control object.</td>
</tr>
<tr>
<td></td>
<td>Horizontal Center</td>
<td>Aligns the horizontal centers of all objects with the horizontal center of the control object.</td>
</tr>
</tbody>
</table>
To align objects

1. Select the objects you want to align.
2. Select the control object.
3. Click the appropriate tool in the Align toolbar. The selected objects will be aligned with the control object.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Right</td>
<td>Top</td>
<td>Aligns the tops of all objects with the top of the control object.</td>
</tr>
<tr>
<td>Vertical Center</td>
<td>Bottom</td>
<td>Aligns the bottoms of all objects with the bottom of the control object.</td>
</tr>
<tr>
<td>Left to Right</td>
<td>Right to Left</td>
<td>Aligns the left sides of all objects with the right side of the control object.</td>
</tr>
<tr>
<td>Top to Bottom</td>
<td>Bottom to Top</td>
<td>Aligns the tops of all objects with the bottom of the control object.</td>
</tr>
<tr>
<td>Width</td>
<td>Height</td>
<td>Resizes all objects to the width of the control object.</td>
</tr>
</tbody>
</table>

*Create!form Designer*
Advanced Object Selection

The Handle Selection tool is an advanced feature that enables you to simultaneously align and re-size objects. By allowing you to select some object handles and not others, the Handle Selection tool lets you dictate which parts of an object are static and which parts can be moved or re-sized.

When used in conjunction with multiple objects, the Handle Selection tool allows you to fix the spatial relationship between non-selected handles on different objects.

**Anchor points**

When a handle is selected, the handle diagonally opposite the selected handle becomes the anchor point. Anchor points are fixed in position.

![Figure 9.5: A box object with one handle selected](image)

When two handles are selected, the opposite two handles become anchor points.

![Figure 9.6: A box object with two selected handles](image)
When a line object is selected, two handles are positioned on the perimeter of the object. When you select one handle, the other handle becomes the anchor point.

![Diagram of a diagonal object with one handle selected.](image)

**Figure 9.7: A diagonal object with one handle selected**

**To select handles**

1. Activate the Handle Selection tool.
2. Position the cursor on an object’s handle and click.
3. To select other handles (on the same or different objects), hold down the shift key and click on the required handles. To select multiple handles at once, you can click-and-drag an area.

The handles of multiple objects become selected as shown below.

![Diagram showing multiple handles selected from different objects.](image)

**Figure 9.8: A selection containing two handles from one object and one handle from another object.**

**To deselect handles**

1. Using the Handle Selection tool, hold down the Shift key and click the selected handle you want to deselect.
To re-size one object while moving another

1. Activate the Handle Selection tool.
2. Partially select handles of the object to be resized.
3. With the Shift key held down, click in the middle of the second object to select the entire object.

For example:

![Figure 9.9: Selected handles and an entire object](image)

Both objects can now be manipulated simultaneously.
4. Click-and-drag downwards to increase the size of one object while moving the second object.

To align handles of objects

When individual handles are selected with entire objects or handles on other objects, the Alignment toolbar may be used to align selected handles. Objects can be automatically resized and aligned with the edge of a control object.

1. Select the handles of the objects you wish to align.
2. Select the control object last (Refer to Align Objects for more information).
3. Click one of the tools on the alignment toolbar.
For example:

![Diagram of object handles being aligned using the Top tool on the Alignment toolbar]

*Figure 9.10: Aligning object handles using the Top tool on the Alignment toolbar*
OBJECTS

- Selecting and Editing Objects in Graphics View

-
You can use variables to control the content of the project, either as objects on a page, or as arguments in expressions and conditions that you can use to control the display of objects, sections and pages.

The following topics are covered in this chapter:

- About Variables
- Data Variables and Derived Variables
- User Variables
- Global User Variables
- System Variables
- Lookup Variables
- Archive Variables
- Environment Variables
- Job Ticket Variables
About Variables

Variables can be used to create text objects, or as arguments in conditions and expressions. There are several different types of variables that you can use in a project; the type of variable you use will depend on where you want to source your data and how it will be applied in the project:

- Data Variables and Derived Variables
- User Variables
- Global User Variables
- System Variables
- Lookup Variables
- Archive Variables
- Environment Variables
- Job Ticket Variables
Data Variables and Derived Variables

Data variables and derived variables are created in the input design (Refer to Chapter 5, Input Design, for more information.). To view the data variables defined in a section, select Section > Data Variables from the Main menu.
User Variables

User Variables are defined by users and derive their value from functions and expressions using other variables as arguments. User variables are created in and belong to a section. User variables can only derive values from data in the source of the section. They can be used for a wide variety of tasks.

Example

You might create a user variable to display a post-tax total. The user variable can take the pre-tax total (a value obtained by a data variable), and multiply it by the sales tax amount.

<table>
<thead>
<tr>
<th>Net Value:</th>
<th>Tax:</th>
<th>Total:</th>
</tr>
</thead>
<tbody>
<tr>
<td>£5,000.00</td>
<td>10%</td>
<td>£5,500.00</td>
</tr>
<tr>
<td>£26,002.00</td>
<td>10%</td>
<td>£28,602.20</td>
</tr>
</tbody>
</table>

User variable that combines the value of the data variable, with the constant tax rate of 10%.

User variables can be used in other sections, provided the section it belongs to is a source for the section where it will be used (Refer to About Sources for more information).

To create a user variable

1. Select the section you want the user variable to be available in/belong to.
2. From the Main menu, select Section > User Variables. The User Variables dialog will be displayed.
3. If required, click the Add Row button. A new variable is created.
4. To create a condition, click the Condition button in the Condition column and refer to Conditions for more information.
5. In the Variable Name column, type a name for the user variable.
6 Click the Expression button to create the expression that will generate the variable value. The Expression builder will be displayed. Refer to Using the Expression Builder for more information.

7 Click OK. Once you have created a user variable, you can use it in text objects, variable text objects or to build other expressions or conditions.
Global User Variables

Global user variables enable you to create a variety of dynamic fields and use them in any section in the project. Global user variables have an initial value and are passed sequentially from section to section. In each section they have their value changed.

Global user variables are typically used in conjunction with user variables. Global user variables can provide values for user variables, or derive values from user variables.

When to use global variables
Global user variables can be used in the following types of task:

- To perform calculations that involve data from unrelated sections in the project
- For conditional based decision making that involves data from an unrelated section in the project

To create a global user variable
1. From the Main menu select Project > Global User Variables. The Global User Variables dialog will be displayed.
2. Click the Add Row button.
3. In the Variable Name column, type a name.
4. In the Initial Value column, type an initial value.
5. In the Reset column, select whether you want the global user variable reset every page, set or document.
6. Click OK.

Global variables can be used in text object, variable text objects or in conditions and expressions.
System Variables

Create!form Designer provides pre-defined variables that you can use to determine information about the project and the operating environment. The following table describes the system variables available in the input and output of the design.

Table 10.1:

<table>
<thead>
<tr>
<th>System Variable</th>
<th>Description</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sys.PreviousSection</td>
<td>Name of previous detail section.</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Sys.DocPageNum</td>
<td>Current page number in the document.</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Sys.DocSetNum</td>
<td>Current set number in the document.</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Sys.SetPageNum</td>
<td>Current page number in the current set.</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Sys.PageNum</td>
<td>Collated page number sequence in Next Copy projects.</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Sys.SetNumPages</td>
<td>Total number of pages in the current set.</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Sys.Today</td>
<td>Current date value supplied by the system.</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Sys.Time</td>
<td>Time of merge supplied by the system.</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Sys.Username</td>
<td>Username property of spool file.</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Sys.Hostname</td>
<td>Name of host server where the merge occurs.</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Sys.OSVersion</td>
<td>Version of operating system running on host</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Sys.SpoolID</td>
<td>Spool file ID on the merge queue</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Sys.DocName</td>
<td>Name of spool file.</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Sys.QueueName</td>
<td>Name of print queue where merge occurs.</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Sys.DeviceName</td>
<td>Name of destination port, queue or device.</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Sys.ProjectName</td>
<td>Name of current project.</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Sys.ProductVersion</td>
<td>Details about Create!form Server installed on host.</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

System variables can be displayed as variable text objects or used as arguments in expressions and conditions.

**Example**

- This expression tests for odd or even numbered pages in a document:
  \[ \text{round}(\text{Sys.DocPageNum}/2,0)\times2-\text{Sys.DocPageNum} \]
• This expression tests if the date stored in InvDate is in the current month:
  
  \[
  \text{month(InvDate)} = \text{month(Sys.Today)}
  \]
  
  The following diagram demonstrates how the page numbering system variables work with collated and uncollated Next Copy projects:

\[Figure 10.2: \text{Page numbering in a Next Copy project}\]
Lookup Variables

Lookup variables retrieve values from external databases. For example you can use lookup variables to retrieve a customer telephone number from a database. Lookup variables can be inserted directly into the project or used as part of an expression or condition.

Lookup variables are created in and belong to a section. They can be used in other sections, provided the section it belongs to is a source for the section where it will be used.

Requirements

To use lookup variables you require a DSN connection to an ODBC database. To enable production printing, you must also create a permanent connection between the e-forms server and the database(s). For more information see the Create!form Server User Guide.

Note

*MS Access Memo field type lookup is restricted to retrieve the first 255 characters.*

To create a lookup variable

1. From the Create!form Designer main menu, select Section > Lookup Variables. The Lookup Variables dialog will be displayed.
220  VARIABLES

- Lookup Variables

Figure 10.3: The Lookup Variables dialog

2 From the drop-down list under the Database (DSN) field, select the DSN for the database you want to retrieve data from, or click New to create a new DSN. Setup procedures for databases will vary.

3 If multiple tables/views exist in the database, select the appropriate one from the Table/view drop-down.

4 In the Select records where field, define a query by selecting:
   - The Database Field from the database table that will be searched in
   - The Variable from the project containing the value that will be searched with

   Ensure the query will only match with one record. If you need to add further qualifiers, clicking the Add Row button and select further fields and variables.

5 In the Assign to variables field, define the response to the above query by:
   - Typing the Variable Name in the project to assign to
• Selecting the Database Field from the database table to assign from
  It is possible to return more than one response from that particular
  record in the database by clicking the Add Row button and defining
  further Variable Names and Fields.

6 By clicking the Refresh button at the bottom right of the dialog the returned
value(s) from the lookup query will be displayed in the Content column.
Archive Variables

Create!archive is the output module that automatically archives documents after they have been merged by Create!form Server. The documents are archived and indexed using archive variables. For Create!archive to work with Create!form Designer, you must define what the archive variables are, and how they derive their values. The archive variables can use pre-defined, or user-defined names.

---

**Note**

*For detailed information on using and generating archive documents, see the Create!archive User Guide.*

---

Create!form Designer creates a standard set of archive variables that can be viewed and edited from the Archive Variables dialog.

**To set the initial value of an archive variable**

1. From the Main menu, select Project > Archive Variables. The Archive Variables dialog will be displayed.
2. In the Initial Value column, type the initial value for each archive variable.
3. Click **OK**.

**To define an archive variable**

1. Select the section that contains the data you want to use to define the value of the archive variable.
2. From the Main menu, select Section > User Variables. The User Variables dialog will be displayed.
3. Click the Add Row button.
4. To create a condition, click the Condition button in the Condition column. Refer to **Building Conditions** for more information.
5. In the Variable Name column, click the drop list button and select a variable from the list of available archive variables.
6. Click the Expression button to create the expression that will generate the variable value. The Expression Builder will be displayed. Refer to **Using the Expression Builder** for more information.
7 Click OK.

To create a user-defined archive variable

Follow the procedure above for defining archive variables, but instead of selecting an existing archive variable, type a new archive variable name in the Variable Name column. The names of archive variables must begin with “ARC_”. You can also create an archive variable from the Archive Variables dialog.

To delete archive variables

1 From the Main menu, select Project > Archive Variables. The Archive Variables dialog will be displayed.

2 Using the controls provided, delete archive variables in the list.

3 Click OK.
Environment Variables

Environment variables are common to all projects in the production environment. Environment variables may be the same for all output destinations, or can be changed for individual print queues.

Typical uses of environment variables include setting values that are used in all projects, like a company name, or using different page styles in the same project depending on whether the destination is a printer, fax, email or archive server port.

Storage

Environment variables are stored in the EnvironmentVars.data file in the SystemResource folder, which will be sent to the e-forms server when the project is published.

To create an environment variable

1. From the Main menu, select Tools> Environment Variables. The Environment Variables dialog will be displayed.
2. To create a new environment variable, click the Add Row button.
3. Type the name of the environment variable in the Name field.
4. Type the value of the environment variable in the Value field.
5. Click OK.
Job Ticket Variables

All the job ticket values passed through with the spool file, or added by the e-forms server are available wherever system variables are available in the design. The job ticket variables available to a project must be defined in the following configuration files found in the Win32User folder:

- Text, CSV and XML input files: HeaderParams.cfg
- JDE PDF input files: JDEHeaderParams.cfg

You can edit these files to modify existing values, and add additional variables. Any changes you make to configuration files on the design workstation will be transferred to the e-forms server with the project. (Refer to Transferring Projects to the e-forms Server for more information.)

---

Note

If you edit the configuration file, you must restart the program.
VARIABLES

- Job Ticket Variables

-
**Chapter 11: Expressions and Conditions**

You can create expressions and conditions to combine and manipulate text, calculate values, summarize data in tables and test when objects, sections and pages should be displayed.

The following topics are covered in this chapter:

- Expressions
- Conditions
- Building Blocks
- Syntax and Order of Precedence Rules
- Summarizing Data in Headers and Footers
Expressions

What do you want to know about?

- Building Expressions
- Using the Expression Builder
- Navigating and Editing Expressions

Building Expressions

Expressions are built using the Expression Builder dialog (referred to as the Expression Builder), which is opened whenever you click the button from either the Derived Variables dialog (input) or the User Variables dialog (output). The Expression Builder provides a range of operators and functions used to build expressions that evaluate values, called arguments, in a specific order, or syntax.

Expressions are evaluated from left to right subject to the syntax and order of precedence rules. Refer to Syntax and Order of Precedence Rules for more information.

Using the Expression Builder

You must build an expression in a particular order. Generally you will follow the list boxes from left to right. The operators, functions and variables available to you are context sensitive. For example, when you select a variable with a numeric value, only functions that can process numbers will be available to you.

The following procedure explains how to build a simple expression to add the text “Inc.” to a user variable called “Company_Name”

To build a simple expression

1. From the Main menu, select Section > User Variables.
2. From the User Variables dialog, in the Variable Name field, type the name of the variable you want to create.
3. Click to open the Expression Builder.
4 From the Expression Builder, click concat in the Function section.

5 In the Variable section, expand and select the user variable.

6 Click on the Edit Value button and type the text you want to add to the user variable. The Expression Builder will display: concat(Company_Name," Inc.")

7 Click OK to close the Expression Builder.

When you close the Expression Builder, you will be returned to the User Variables dialog where the expression will be displayed in the Value field, and the value the expression returns in the current section will be displayed in the Content field. The same procedure applies when building expressions from the Derived Variables dialog.

Refer to General Preferences for more information on how to enable direct editing of the expression in the dialog that calls the Expression Builder.

Navigating and Editing Expressions

- To move around the expression, use the arrow buttons, keyboard arrow keys and the cursor
- To expand or reduce the selection, use the Level Up button or the Level Down button
- To add text or number values, use the Edit Value button
- To delete parts of an expression, select the part of the expression to be deleted and click the Delete button
- To enclose parts of an expression in parentheses, select the part of the expression to be enclosed and click the Group button

Refer to General Preferences for more information on how to enable direct editing of the expression in the dialog that calls the Expression Builder.
A condition is a special type of expression which produces a TRUE or FALSE value. Conditions allow you to vary the output of the project, depending on data within the input file. For more information about the ways you can use conditions, see the usage examples below.

**Usage examples**

Conditions can be used to determine:

- When input sections are created (Refer to Add Detail Sections and Add Input Footer Sections for more information)
- When sets are created (Refer to Defining Sets for more information)
- When sections are displayed (Refer to Data Section Display Options for more information)
- Which section will be displayed, when using alternates (Refer to Alternate Sections for more information)
- When objects are displayed (Refer to General Properties for more information)
- When defining derived values
- When calculating values or manipulating data in user variables
- When About Page Styles are applied
- When Attaching Subforms to Inserted Pages
- When Archive Variables are used

**What do you want to know about?**

- Building Conditions
- Using Expressions as Conditions
Building Conditions

The Condition Builder is used to build and edit expressions in much the same fashion as the Expression Builder (Refer to Expressions for more information). The Condition Builder can be opened from numerous dialogs, either by clicking the Edit button next to a condition field, or by clicking the button.

The following procedure explains how to build a simple condition to test whether the user variable called “Item_type” contains the word “new”.

To build a simple condition

1. From the Condition Builder, click contains in the Function section.
2. In the Variable section, expand and select the user variable.
3. Click on the Edit Value button and type “new”. The Condition Builder will display: contains(Item_type,"new").
4. Click OK to close the Condition Builder.

When you close the Condition Builder, you will be returned to the originating dialog where the expression will be displayed. Refer to General Preferences for information on how to enable direct editing of the expression in the dialog that calls the Condition builder.

Using Expressions as Conditions

Any expression that evaluates to a non-zero value when true can be used as a condition. For example, the expression A-B is equivalent to the condition A!=B as both expressions return a zero value when A equals B and a non-zero value when A is not equal to B.
Building Blocks

The Expression Builder and Condition Builder dialogs contain list boxes containing the building blocks from which you can build an expression. The content of these dialogs is context sensitive; the building blocks displayed or available will depend on the task you are performing:

- Boolean Operators
- Operators
- Functions
- Variables
- System Conditions

Boolean Operators

Boolean operators are only available in the Condition Builder. The syntax of an expression involving boolean operators begins with the first argument, followed by the operator, and the second argument, separated by spaces:

condition1 OPERATOR condition2

Arguments can be other user defined conditions or system conditions. The following table describes the boolean operators available.
Other boolean operations can be performed by combining AND and OR operators in expressions with the negation operator NOT; for example, a boolean difference operation is performed by the expression (Refer to Operators for more information):

\[
\text{NOT condition1 AND condition2}
\]

### Operators

The usual syntax of an expression involving operators begins with the first argument, followed by the operator, and the second argument, with or without separating spaces:

\[
\text{argument1} \text{OPERATOR} \text{argument2}
\]

The negation operators are the only exceptions to this rule, as they precede a single argument. Arguments can be numbers, text strings, variables or other functions or expressions. The following table describes the operators available.
## EXPRESSIONS AND CONDITIONS

### Building Blocks

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning</th>
<th>Syntax and Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>==</code></td>
<td>equal to</td>
<td>A==B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Compares the values of two numbers or two strings and returns TRUE if equal. Case</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sensitive when comparing strings.</td>
</tr>
<tr>
<td><code>!=</code></td>
<td>not equal to</td>
<td>A!=B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Compares the values of two numbers or two strings and returns TRUE if not equal. Case</td>
</tr>
<tr>
<td><code>&gt;</code></td>
<td>greater than</td>
<td>A&gt;B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Compares the values of two numbers and returns TRUE if A is greater than B.</td>
</tr>
<tr>
<td><code>&lt;</code></td>
<td>less than</td>
<td>A&lt;B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Compares the values of two numbers and returns TRUE if A is less than B.</td>
</tr>
<tr>
<td><code>&gt;=</code></td>
<td>greater than or equal to</td>
<td>A&gt;=B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Compares the values of two numbers and returns TRUE if A is greater than or equal to B.</td>
</tr>
<tr>
<td><code>&lt;=</code></td>
<td>less than or equal to</td>
<td>A&lt;=B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Compares the values of two numbers and returns TRUE if A is less than or equal to B.</td>
</tr>
<tr>
<td><code>*</code></td>
<td>multiply</td>
<td>A*B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Multiplies two numbers.</td>
</tr>
<tr>
<td><code>/</code></td>
<td>divide</td>
<td>A/B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Divides A by B.</td>
</tr>
<tr>
<td><code>+</code></td>
<td>add</td>
<td>A+B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Adds two numbers, or concatenates two strings.</td>
</tr>
<tr>
<td><code>-</code></td>
<td>subtract</td>
<td>A-B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Subtracts B from A.</td>
</tr>
<tr>
<td><code>NOT</code></td>
<td>logical negation</td>
<td>NOT condition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Reverses the logical value of an expression.</td>
</tr>
<tr>
<td><code>- (neg)</code></td>
<td>arithmetic negation</td>
<td>-B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Reverses the sign of a numeric value, i.e. -B=-1*B</td>
</tr>
</tbody>
</table>
Functions

The syntax of a function begins with the function name, followed by an opening parenthesis, the arguments for the function separated by commas, and a closing parenthesis.

function(argument1,argument2,...)

Arguments can be numbers, text strings, variables, functions and expressions. The following tables describe the functions available.

Logical functions

Logical functions can be used as conditions returning a TRUE or FALSE value. When used as arguments in expressions, logical functions return a value of 1 if true, and 0 if false.

<table>
<thead>
<tr>
<th>Function</th>
<th>Syntax and Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>contains</td>
<td>contains(string,find_text)</td>
</tr>
<tr>
<td></td>
<td>Returns TRUE if string contains find_text.</td>
</tr>
<tr>
<td></td>
<td>Example: contains(&quot;Heath;Adele;Mrs;;OK;;&quot;,&quot;OK&quot;) = TRUE</td>
</tr>
<tr>
<td>exists</td>
<td>exists(name)</td>
</tr>
<tr>
<td></td>
<td>Returns TRUE if the variable name exists.</td>
</tr>
<tr>
<td>isnumeric</td>
<td>isnumeric(name)</td>
</tr>
<tr>
<td></td>
<td>Returns TRUE if the variable name contains at least one numeral and any of the characters , . ( ) + - $ % or space. The test will fail if the variable contains any other character.</td>
</tr>
<tr>
<td></td>
<td>Examples:</td>
</tr>
<tr>
<td></td>
<td>isnumeric(&quot;$ 1,234.56&quot;) = TRUE</td>
</tr>
<tr>
<td></td>
<td>isnumeric(&quot;$US1,234.56&quot;) = FALSE</td>
</tr>
</tbody>
</table>
## Text and Numeric Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Syntax and Usage</th>
</tr>
</thead>
</table>
| aswords  | aswords(number,option)  
Converting a number into words. Specifying "a" in the second  
argument will generate a number phrase with “and” e.g. “Five  
hundred and twenty seven”  
Examples:  
aswords("123","") = "One hundred twenty three"  
aswords("123","a") = "One hundred and twenty three" |
| at       | at(row,column,length)  
Returns a string of a specified length, starting at the specified row  
and column position in the current input page.  
Example: if row 5 of the input page contains the letters of the  
alphabet in columns 1 to 26, then  
at(5,12,3) = "LMN" |
| concat   | concat(string1,string2)  
Joins two text strings.  
Example: concat("Adele ", "Heath") = "Adele Heath" |
| find     | find(string,find_text,start_char)  
Finds one text string within another text string, starting from  
character position start_char, and returns the starting position. The  
first character in string is position 1. Returns 0 if find_text is not  
found in string.  
Example: find("Mrs Adele Heath"," ",5) = 10 |
| length   | length(string)  
Returns the length of a text string, including spaces.  
Example: length("Mrs Adele Heath") = 15 |
| lower    | lower(string)  
Converts a text string to lower case. |
| ltrim    | ltrim(string)  
Trims leading spaces from a text string.  
Example: ltrim(" left trim ") = "left trim " |
| mod      | mod(number,base)  
Returns the remainder after number is divided by base.  
Example: mod(51,7) = 2 |
## Expressions and Conditions

### Building Blocks

<table>
<thead>
<tr>
<th>Function</th>
<th>Syntax and Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>num</td>
<td>num(string)</td>
</tr>
<tr>
<td></td>
<td>Converts a text string into a number.</td>
</tr>
<tr>
<td></td>
<td>Examples: num(&quot;12ab3&quot;) = 123</td>
</tr>
<tr>
<td></td>
<td>num(&quot;18+9/3&quot;) = 1893</td>
</tr>
<tr>
<td></td>
<td>num(&quot;nine&quot;) = 0</td>
</tr>
<tr>
<td>pad</td>
<td>pad(string,length)</td>
</tr>
<tr>
<td></td>
<td>If string is shorter than length, extends string to length with spaces.</td>
</tr>
<tr>
<td></td>
<td>If string is longer than length, truncates string to length.</td>
</tr>
<tr>
<td></td>
<td>Examples: pad(&quot;abc&quot;,&quot;6&quot;) = &quot;abc   &quot;</td>
</tr>
<tr>
<td></td>
<td>pad(&quot;abc&quot;,1) = &quot;a&quot;</td>
</tr>
<tr>
<td></td>
<td>pad(&quot;abc&quot;,-2) = &quot;abc&quot;</td>
</tr>
<tr>
<td></td>
<td>pad(&quot;abc&quot;,&quot;def&quot;) = &quot;&quot;</td>
</tr>
<tr>
<td>proper</td>
<td>proper(string)</td>
</tr>
<tr>
<td></td>
<td>Capitalizes the first letter of each word in a text string.</td>
</tr>
<tr>
<td>previous</td>
<td>previous(var_name)</td>
</tr>
<tr>
<td></td>
<td>Returns the value of variable var_name in the previous occurrence of the current section.</td>
</tr>
<tr>
<td>previousat</td>
<td>previousat(row,column,length)</td>
</tr>
<tr>
<td></td>
<td>Returns a string of a specified length, starting at the specified row and column position in the previous input page.</td>
</tr>
</tbody>
</table>
### Building Blocks

<table>
<thead>
<tr>
<th>Function</th>
<th>Syntax and Usage</th>
</tr>
</thead>
</table>
| **retrieve** | `retrieve(string,ordinal_position,delimiter_char)`<br>Extracts text from delimited fields in a text string. When the space character is used as a delimiter, leading spaces are ignored and sequential spaces are treated as a single delimiter.<br>Examples:<br>`retrieve("Abc;De;F;;;;Ghij;;;;6;;;;","") = "Ghij"`<br>`retrieve(" Abc De F"," ") = "De"`<br>Use the following arguments when specifying these delimiters:<br>ArgumentDelimiter<br>

- \`\n`<br>new line<br>- \`\t`<br>tab<br>- \" \"<br>"double quotation mark (")<br>- \`\\`<br>\backslash ()

**Example:**<br>`retrieve(var,1,"\t")` returns the value before the first tab in the text variable var. |
| **rfind** | `rfind(string,find_text,start_char)`<br>Finds one text string within another text string by searching from the right to the left, starting from character position start_char, and returns the starting position. The first character in string is position 1. Position 0 specifies the last character in string. Returns 0 if find_text is not found in string.<br>Example: `rfind("Heath,Adele,Mrs","",0) = 12` |
| **round** | `round(number,num_digits)`<br>Rounds a number to a specified number of decimal places.<br>Examples:<br>`round(1234.567,2) = 1234.57`<br>`round(1234.56,0) = 1234`<br>`round(1234.56,-2) = 1200` |
| **rtrim** | `rtrim(string)`<br>Trims trailing spaces from a text string.<br>Example: `rtrim(" right trim ") = " right trim"` |
| **sentence** | `sentence(string)`<br>Converts a text string to sentence case. |
**Function** | **Syntax and Usage**
---|---
substitute | substitute(string,old_text,new_text)
Replaces all occurrences of existing text with new text in a text string.
Example: substitute("Contact <N> for quote","<N>","Chris") = "Contact Chris for quote"
substring | substring(string,start_char,length)
Returns a portion of a text string.
Example: substring("Telephone: 303-8783 2054",12,3) = "303"
upper | upper(string)
Converts a text string to upper case.

Refer to **Syntax and Order of Precedence Rules** for more information.

**Date Functions**
The result of a date function depends on the current data order setting. Refer to **Set Input Data Formats** for more information.

<table>
<thead>
<tr>
<th>Function</th>
<th>Syntax and Usage</th>
</tr>
</thead>
</table>
date | date(year,month,day)
Returns a date value from arguments representing the year, month and day. Values of day and month outside the normal range are treated on the basis that month=0 is the last month of the previous year, and day=0 is the last day of the previous month.
The following examples assume a date order setting of d-m-y:
date("2003","4","19") = 19-4-2003
date(2000,0,0) = 30-11-1999
date(2004,-1,32) = 2-12-2003
day | day(date)
Returns the day of the month (a number between 1 and 31) from a date value.
Example: day("19/4/03") = 19 when date order is d-m-y
Any non-numeric parsing character is permitted in a string representing a date value, for example, the strings "19 4 03" and "19.04.2003" will return the same value when used as arguments in a date function.

Refer to Syntax and Order of Precedence Rules for more information on addition and subtraction operators with date values.

**Statistical Functions**

Table functions are defined in a header or footer, and perform calculations on variables that occur within the part of the document associated with the header or footer. Refer to Summarizing Data in Headers and Footers for more information on how to used statistical functions to summarize data in tables, pages and sets.

### Table 11.1:

<table>
<thead>
<tr>
<th>Function</th>
<th>Syntax and Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>count(name)</td>
</tr>
<tr>
<td></td>
<td>Counts the number of times the variable occurs within part of a document.</td>
</tr>
<tr>
<td>max</td>
<td>max(name)</td>
</tr>
<tr>
<td></td>
<td>Finds the maximum value of a variable within part of a document.</td>
</tr>
<tr>
<td>min</td>
<td>min(name)</td>
</tr>
<tr>
<td></td>
<td>Finds the minimum value of a variable within part of a document.</td>
</tr>
<tr>
<td>sum</td>
<td>sum(name)</td>
</tr>
<tr>
<td></td>
<td>Sums a variable within part of a document.</td>
</tr>
</tbody>
</table>
**Variables**

The types of variables that can be used in conditions as arguments of operators and functions are summarized below:

<table>
<thead>
<tr>
<th>Variable Type</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>data variable</td>
<td>Input data. Refer to <a href="#">Design Principles</a> for more information.</td>
</tr>
<tr>
<td>user variable</td>
<td>User defined expressions relating to section data. Refer to <a href="#">User Variables</a> for more information.</td>
</tr>
<tr>
<td>global user variables</td>
<td>User defined expressions relating to other sections or the whole project. Refer to <a href="#">Global User Variables</a> for more information.</td>
</tr>
<tr>
<td>system variable</td>
<td>System supplied data. Refer to <a href="#">System Variables</a> for more information.</td>
</tr>
<tr>
<td>lookup variable</td>
<td>Data from an external database. Refer to <a href="#">Lookup Variables</a> for more information.</td>
</tr>
<tr>
<td>environment variable</td>
<td>Data that applies to all projects. Refer to <a href="#">Environment Variables</a> for more information.</td>
</tr>
<tr>
<td>job ticket variables</td>
<td>Job file attributes. Refer to <a href="#">Job Ticket Variables</a> for more information.</td>
</tr>
</tbody>
</table>

The variables that are available to you at any time, will depend on the type of expression you are building, and where you are building it from. User variables and data variables are available only in the section they are defined in.

**System Conditions**

Create!form Designer provides pre-defined conditions that you can use to:
- Conditionally apply an input section (text and JDE PDF input file types)
- Conditionally display text objects or variable text objects
- Conditionally apply page styles
- Conditionally define derived variables

System conditions will only be displayed in the Condition Builder when you are performing one of the above tasks. The following table describes the system conditions available.
## Pre-defined conditions

<table>
<thead>
<tr>
<th>Pre-defined conditions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sys.FirstPageOfDoc</td>
<td>Returns TRUE when the current section is on the first page of the project.</td>
</tr>
<tr>
<td>Sys.ContPageOfDoc</td>
<td>Returns TRUE when the current section is not on the first page or the last page of the project.</td>
</tr>
<tr>
<td>Sys.LastPageOfDoc</td>
<td>Returns TRUE when the current section is on the last page of the project.</td>
</tr>
<tr>
<td>Sys.FirstPageOfSet</td>
<td>Returns TRUE when the current section is on the first page of the set for the project.</td>
</tr>
<tr>
<td>Sys.ContPageOfSet</td>
<td>Returns TRUE when the current section is not on the first page or the last page of the set for the project.</td>
</tr>
<tr>
<td>Sys.LastPageOfSet</td>
<td>Returns TRUE when the current section is on the last page of the set for the project.</td>
</tr>
</tbody>
</table>
Syntax and Order of Precedence Rules

Expressions are evaluated using a specific set of rules, or syntax, that depends on the operators and functions you use in the expression. The Condition Builder and Expression Builder dialogs will assist you in building expressions with a valid syntax; however you must follow a few basic rules to ensure the expression will perform the way you expect:

- Mixed Data Types
- Addition Operator
- Subtraction Operator
- Comparing Text Strings
- Order of Precedence
- Nested Functions
- Parsing
- Names of Variables

Mixed Data Types

Avoid mixing text, numeric and date values in the same expression as this can produce unexpected results. All variable data is stored as a string, but will be interpreted as a number in an arithmetic operation. For example:

"-2"*"3" evaluates to the number -6
2*"3A4" evaluates to the number 68
2*"3-4" evaluates to the number 68
2*"3A4-" evaluates to the number -68
"two"*"three" evaluates to the number 0
Addition Operator

The addition operator (+) is a special case. The rules governing the addition of numeric, text and date arguments are:

• if both arguments of an addition are numeric, an addition is performed. For example: 1+1 evaluates to the number 2.
• if either argument of an addition is a string, the arguments are concatenated. For example: 1+"1" evaluates to the text string "11".
• if one argument is a date and the other is a number, a date addition is performed. For example: if Today is a date variable (m-d-y) with a value of 9-6-2004, the expression Today+1 evaluates to the date value 9-7-2004, whereas the expression "9/6/2004"+1 evaluates to the string "9/6/20041".

Subtraction Operator

The subtraction operator (-) is a special case. The rules governing the subtraction of numeric, text and date arguments are:

• if both arguments of a subtraction are either numeric or text, a subtraction is performed. For example: "a5"-2 evaluates to the number 3.
• if the first argument is a date and the second is a number, a date subtraction is performed. For example: if Today is a date variable (m-d-y) with a value of 9-6-2004, the expression today-1 evaluates to the date value 9-5-2004, whereas the expression "9/6/2004"-1 evaluates to the number 962003.
• if both arguments are dates, a date difference is performed. For example: Sys.Today-InvDate evaluates to the number of days difference between today’s date and the date variable InvDate, whereas the expression "9/6/2004"-"9/6/2003" evaluates to the number 1.
Comparing Text Strings

Avoid using the comparison operators <=, >=, <, > to compare text strings as this can produce unexpected results. The rules governing comparisons of numeric, text and date arguments are:

- if either argument of a comparison is a number, a numeric comparison is performed. For example: "10">2 returns TRUE.
- if both arguments are strings, a string comparison is performed based on the ASCII collating sequence, and is case sensitive. For example: "10">"2" returns FALSE.
- if both arguments of a comparison are dates, a date comparison is performed.

Order of Precedence

Expressions are evaluated in the following order of precedence:

- operations inside parentheses ( ) are evaluated first, followed by
- - (negation)
- *, /
- +, -
- ==, !=, <=, >=, <, >
- NOT
- AND, OR

Operators with the same order of precedence e.g. + and -, are evaluated from left to right. For example, the expression 3*2+6/-2 evaluates to 3, whereas 3*(2+6/-2) evaluates to -3.
Nested Functions

Functions can be used as arguments for other functions, and can be nested. For example, if the variable A has a value of "Andersch,Jorge", then the expression:

\[ \text{length(substring(A,find(A","",1)+1,length(A)-find(A","",1)))} \]

will return the value 5 (the length of the substring "Jorge")

Parsing

Certain characters or combinations of characters, when included in text strings, can have specific meanings in an expression:

Table 11.1:

<table>
<thead>
<tr>
<th>Characters</th>
<th>Description</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;</td>
<td>text delimiter</td>
<td>Quotation marks are used to define the start and end of a text string.</td>
</tr>
<tr>
<td>\</td>
<td>quotation mark</td>
<td>Quotation marks are treated literally when preceded by a backslash.</td>
</tr>
<tr>
<td>\</td>
<td>backslash</td>
<td>The backslash character is treated literally when preceded by another backslash.</td>
</tr>
<tr>
<td>\n</td>
<td>new line</td>
<td>Inserts a line feed.</td>
</tr>
<tr>
<td>-</td>
<td>minus sign</td>
<td>When a minus sign is adjacent to a number, the number is interpreted as a negative in an arithmetic operation. A minus sign embedded inside a number is ignored.</td>
</tr>
</tbody>
</table>

For example,
"the literal \"\" is ignored" evaluates to the literal "\" is ignored
"C:\\Program Files\\Create!form" evaluates to C:\Program Files\Create!form
2*"1A2B3-" evaluates to -246
2*"1A2-3" evaluates to 246
Names of Variables

Avoid using function names as names of variables as this can produce unexpected results (Refer to Functions for more information.).
Summarizing Data in Headers and Footers

You can use the statistical Functions count, max, min and sum for calculations in any header or footer section. The function is performed over the part of the document associated with the header or footer. For example, when defined as a variable in a set header, the expression sum(Price) will sum all occurrences of the data variable Price within the following set.

Rules for calculating statistical functions

• In page header and footer sections, the functions will calculate based on the content of the page
• In set header and footer sections, the functions will calculate based on the content of the set
• In set continuation header and footer sections, the functions will calculate based on the content of the set, up to that point
• In table header and footer sections, the functions will calculate based on the content of the table
• In table continuation header and footer sections, the functions will calculate based on the content of the table, up to that point
• In table level header and footer sections, the functions will calculate based on the content of the level
• In table level continuation header and footer sections, the functions will calculate based on the content of the level, up to that point

To use a statistical function in a header or footer section

1. In the header or footer section, create the user variable (Refer to User Variables for more information.) and use the Expression Builder to build an expression using the count, max, min and sum functions. For example, create a user variable called AvgPrice using the expression sum(Price)/count(Price), where Price is a data variable which occurs in data sections that are repeated within a table.

2. Insert the user variable as a text object or variable text object (Refer to Creating Objects for more information.), in a header or footer section. The header or footer section can be of a table, a table level, a page, or a set.
Example

The following table provides examples of what values will be created in each header and footer section using the count and sum functions.

<table>
<thead>
<tr>
<th>Page 1</th>
<th>Page 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>(count, sum)</td>
<td>(count,sum)</td>
</tr>
<tr>
<td>Set Header (8,20)</td>
<td>Set Cont. Header (4,9)</td>
</tr>
<tr>
<td>Page Header (4,9)</td>
<td>Page Header (4,11)</td>
</tr>
<tr>
<td>Table Header (8,20)</td>
<td>Table Cont. Header (4,9)</td>
</tr>
<tr>
<td>Table Level Header (3,7)</td>
<td>Table Level Cont. Header (1,2)</td>
</tr>
<tr>
<td>Data Section value: 3</td>
<td>Data section value: 2</td>
</tr>
<tr>
<td>Data section value: 3</td>
<td>Data section value: 1</td>
</tr>
<tr>
<td>Data section value: 1</td>
<td>Table Level Footer (3,5)</td>
</tr>
<tr>
<td>Table Level Footer (3,7)</td>
<td>Table Level Header (2,8)</td>
</tr>
<tr>
<td>Table Level Header (3,5)</td>
<td>Data section value: 5</td>
</tr>
<tr>
<td>Data section value: 2</td>
<td>Data section value: 3</td>
</tr>
<tr>
<td>Table Level Cont. Footer (4,2)</td>
<td>Table Level Footer (2,8)</td>
</tr>
<tr>
<td>Table Cont. Footer (4,9)</td>
<td>Table Footer (8,20)</td>
</tr>
<tr>
<td>Page Footer (4,9)</td>
<td>Page Footer (4,11)</td>
</tr>
<tr>
<td>Set Cont. footer (4,9)</td>
<td>Set Footer (8,20)</td>
</tr>
</tbody>
</table>

Note

By definition, the use of calculations within the project will create new data values in your output. Common uses of the calculations feature include page sub-totaling and calculation of line item tax rates. Please note that it is always recommended that critical page data should only be modified from within your originating financial or ERP system. Create!form is not responsible for the misuse of this feature and suggests that any calculated data conditions are tested to ensure proper results.
EXPRESSIONS AND CONDITIONS

- Summarizing Data in Headers and Footers
PAGE STYLES

You can create different page styles and define rules that control when a particular style should be used.

The following topics are covered in this chapter:

• About Page Styles
• Create Page Styles
• Define Page Style Rules
• Insert Pages Using Page Styles
• Create Background Graphics for Page Styles
About Page Styles

You can define additional page styles that will be applied when specific conditions are met. The Default Page Style will otherwise be applied. Page style options include:

• Input and output trays
• Duplex options
• Inserted pages
• Background graphics

You can display the current page style by double-clicking the Background tab, or by selecting Page > Style Properties from the Main menu.

What do you want to do?

• Create Page Styles
• Define Page Style Rules
• Insert Pages Using Page Styles
• Create Background Graphics for Page Styles
Create Page Styles

To create page styles

1. From the Main menu, select Page > Manage Page Styles. The Manage Page Styles dialog will be displayed.

2. Click Create. The New Page Style dialog will be displayed.

3. In the Name field, type a name for the style.

4. From the Based on Page Style drop-down, select the page style you want to base your new style on. By selecting <Project> you will adopt the default page settings (Refer to the The Default Page Style for more information.)

5. Click OK. The Page Style dialog will be displayed.

6. Configure the page setup as you did when you set up The Default Page Style.

7. Click OK.

8. Close the Manage Page Styles dialog or create new styles.

Now that you have created styles, you must create rules defining when the style is applied (Refer to Define Page Style Rules for more information).
Define Page Style Rules

Page style rules are “if, then” rules. They determine when a particular page style is applied. You can use pre-defined conditions such as the page number of the project or global variables to decide when a page style will be applied.

**To define page style rules**

1. From the Main menu, select Page > Page Style Rules. The Page Style Rules dialog will be displayed.
2. Click the Add Row button, or click in the grid. A row will be added.
3. Click the Condition button. The Condition Builder will be displayed.
4. Create the condition you want to determine whether the page style is used and click **OK**. Typically you might apply a page style for the first or last page of a set. Refer to **System Conditions** for more information.
5. From the drop-down in the Page Style column, select the page style you want used when the condition is met.
6. If necessary, click on the page style button and edit the page style.
7. If necessary, create further conditions and use the up and down arrow buttons to ensure that Create!form Designer will evaluate the rules in the correct order.
8. Click **OK**.

Using this same basic procedure, you can control the tray selection, duplex options and format of any page.
Insert Pages Using Page Styles

The Page Style dialog gives you the option to insert a page before and after a particular page style (Refer to Create Page Styles for more information). For example, you can use this feature to add a form letter at the start of each set of documents to the same customer. When you insert a page, a new page style will be created, which you can edit.

To insert pages before or after the current page

1. From the Main menu, select Page > Manage Page Styles, create a new page style based on the current page and, from the Page Style dialog, select one or both options to Insert page before or Insert page after. Refer to Create Page Styles for more information.

2. Click OK to return to the Manage Page Styles dialog. A new page style will appear in the list of page styles for each inserted page.

3. Select the inserted page style and click Edit. The Page Style dialog will be displayed.

4. Select the page style settings as required. Note that inserted page styles have the following additional options:
   - **Display condition**
     Set a condition to control whether the inserted page is displayed. If left blank, the inserted page will always be displayed whenever the original page style is applied. The display of the original page style is controlled by the page style rules. Refer to Define Page Style Rules for more information.
   - **Use attachment**
     Select a subform (.cff6 or .eps) to be displayed on the inserted page; the subform will be positioned at the top left of the print area, inside the margins of the page. You can add objects to the inserted page, with or without an attached subform. Refer to Attaching Subforms to Inserted Pages for more information.

5. Close the Page Style dialog and the Manage Page Styles dialog.

6. From the Main menu, select Page > Page Style Rules, and define the rules for displaying the new page style. Refer to Define Page Style Rules for more information.
The new page style, and its inserted page or pages, will be displayed whenever the rules are satisfied.

To change the style of inserted pages

1. From the Main menu, select Page > Manage Page Styles. The Manage Page Styles dialog will be displayed. You can also open this dialog by double-clicking the background tab of the inserted page, or by double-clicking the inserted page in the Page Styles folder of the Project Tree.

2. In the Page Styles field, select the inserted page style.

3. Click Edit. The Page Style dialog will be displayed.

4. Configure the page style to be applied to the inserted page.

5. Click OK.


Attaching Subforms to Inserted Pages

You can attach one or more subforms to pages inserted using page styles. Refer to Insert Pages Using Page Styles for more information.

The Use attachment field on the Page Style dialog of an inserted page contains an expression, which returns the prefix of either a cff6 or eps subform. If both exist, the cff6 subform will be used. For example, if the value is "letter", Create!form Director will look for the subform files letter.cff6 and letter.eps in the project directory and the common project directory. Refer to Chapter 11, Expressions and Conditions, for more information.

Multiple Subforms

Create!form Director will look for additional subforms according to the naming convention prefix-N.cff6/eps, where “N” is a sequential integer starting from 2.

For example, using the "letter" example above, Create!form Director looks for subform files named letter-2.cff6/eps, letter-3.cff6/eps and so on. An additional page will be inserted for each sequentially named subform.
Create Background Graphics for Page Styles

Where a page style is used, a tab called <name of page style> Background will replace the Project Background tab on the Status bar.

You can add text and graphical objects to the background layer. Refer to Creating Objects for more information. For more information about the Background Layer refer to Project Background.

When you double-click <name of page style> Background tab, the Page Style dialog will be displayed and you can alter the page style.
258 * PAGE STYLES
   - Create Background Graphics for Page Styles
   - 
   -
CUSTOMIZING

You can customize the look and feel of the Create!form Designer interface, set language options, add fonts and other user preferences.

The following topics are covered in this chapter:

- User Preferences
- Spelling and Language
- Fonts
- Colors
User Preferences

You can customize the look, feel and behavior of Create!form Designer to suit the way you work. User preference settings are specific to the local copy of Create!form Designer; changes made to user preferences are not saved with the project.

All user preferences are set from the Preferences dialog, which will be displayed by selecting Tools > Preferences from the Main menu. The categories of settings available are displayed in the tree view on the left of the dialog. Select the category in the tree view to display the corresponding controls on the right.

Figure 13.1: Preferences dialogue
• General Preferences
• Section Borders Preferences
• Input Source Preferences
• Design Window Preferences
• Design Window Page Preferences
• Design Window Sections Preferences
• Thumbnails Preferences
• Input (CSV) Window Preferences
• Input (JDE PDF) Window Preferences
• Input (JDE PDF) Window Sections Preferences
• Input (Text) Window Preferences
• Input (Text) Window Sections Preferences
• Input (XML) Window Preferences

**General Preferences**

Select Tools > Preferences from the Main menu to display the Preferences dialog and select General in the tree view.

The following general preferences settings are available:

**Default project directory**
Click the open file button to change the project directory. Refer to *Project Directory* for more information.

**Common project directory**
Click the open file button to change the common project directory. Refer to *Common Project Directory* for more information.

**Spell check language**
To select a different language for checking spelling, choose a language from the drop down list in the spell check language field. Refer to *Spelling and Language* for more information.
**Events to be recorded**
Select the level required for logging. To use the level set in Create!form Server, select Default to printer settings.

**Allow direct editing of expressions and conditions**
Select the checkbox to enable direct editing of expressions and conditions in dialogs without opening the Expression Builder or the Condition Builder.

**Display dialog on creating objects**
Select the checkbox to display the Properties dialog whenever a new text line, grid, box and ellipse object is created from the Object toolbar (Refer to Creating Objects for more information). You can also display the Properties dialog of an object when you:
- Double click the object
- Right click the object and select Properties from the shortcut menu
- Select Edit > Properties from the Main menu

**Make backup on save**
Select the checkbox to make a backup copy of the project and DataMap files when the project is saved. Refer to Saving a Project for more information.

**Restore Suppressed Messages**
Select this option to restore messages that have previously been suppressed. A message is suppressed when the Do not show this message again option is selected on a message dialog.

**Generate log file**
Select this option to record events in a log file.

---

**Section Borders Preferences**
Select Tools > Preferences from the Main menu to display the Preferences dialog and select General > Section Borders in the tree view. The following preferences settings are available:
- The color of a section when it is selected
- The color of the corresponding selection in an inactive window
- The thickness of the border
Input Source Preferences

The following settings apply when View > Input File Source is selected.
Selected Tools > Preferences from the Main menu to display the Preferences dialog and select General > Input Source in the tree view. The following preferences settings are available.

Display Font
- Select the input file source view display font in the drop down
- Select the font size
- Select the Monospace only checkbox to restrict the fonts available in the font drop down

Colors
Select the colors for the text and the background.

Scroll bars
Display and hide the horizontal and vertical scroll bars.

Design Window Preferences

Select Tools > Preferences from the Main menu to display the Preferences dialog and select Design in the tree view. The following preferences settings are available.

View
To set the default page view magnification choose Fit to Width, Fit in Window or Actual Size from the drop down list.

Rulers
Horizontal and vertical rulers are available to assist in object layout and placement. Use the Ruler options to:
- Hide or show ruler
- Snap objects to the ruler. By snapping objects to the ruler, you are snapping them to the ruler units that are displayed. This will vary depending on the zoom. You can snap objects to the ruler or the grid, but not both.
Grid
A grid can be displayed in the Design Window to assist in object layout and placement. Use the Grid options to:

- Hide or show a grid
- Specify the grid interval
- Snap objects to the grid.

Turning on snap to grid will turn off snap to ruler.

Unit of Measure
To set the page units of measure, select the Points, Units or CM radio button. The unit of measure you select will be used for:

- The units displayed in the ruler and grid of the Design Window
- The page coordinates displayed in the Status Bar
- Setting a page size
- Setting a section size
- Setting the size of an object
- Moving objects using the Objects button

Nudge Offset
Type the offset distance in the current units of measure applied when nudging objects with the arrow keys. Refer to Move Objects for more information.

Show Background Forms
Select the checkbox to display graphics or text set in the project background (through the page style).

Synchronize with Input Window
Select the checkbox to synchronize the page and section display in the Design Window with selections made in the Input Window. You can also separately synchronize the display in the Input Window with selections made in the Design Window. Refer to Input (CSV) Window Preferences for more information.

Show Variables in Red
Select the checkbox to highlight variable text objects in red in the Design Window.
Show Text Object Outline
Select the checkbox to display the frame of all text and variable objects.

Show Variable Names
Select the checkbox to display the actual variable names on screen in place of the values.

Enable Tooltips
Select the checkbox to display tooltips.

Design Window Page Preferences
Select Tools > Preferences from the Main menu to display the Preferences dialog and select Design > Page in the tree view. The page size, margins and orientation settings defined by this dialog will be applied whenever a new project is created; changing page preferences will not change the page settings in the current project. The following preferences settings are available:

Page size
Select a standard page size from the drop down.

Page margins
Type the dimension of each margin in each margin field.

Orientation
Select Landscape or Portrait page style.

Gutter background
Select the color for display of the page gutter.

Line Style
Select the line styles for display of the page and gutter margins.

Design Window Sections Preferences
Select Tools > Preferences from the Main menu to display the Preferences dialog and select Design > Sections in the tree view. The following preferences settings are available:
Section Borders

- Select the Show checkbox to display section borders in the Design Window.
- Select the section border line style from the Style drop down.

Selected Section

- Select the Highlight background checkbox to display the selected section in the background color shown. To change the background color, click the background color field to display the Color Picker dialog. Refer to Colors for more information.
- Select the Show repeats checkbox in the Objects section to display all objects in all occurrences of the selected section.

Unselected Sections

- Select the Highlight background checkbox to display unselected sections in the background color shown. To change the background color, click the background color field to display the Color Picker dialog. Refer to Colors for more information.
- Select the Show checkbox in the Objects section to display all objects in unselected sections.
- Select the gray checkbox in the Objects section to gray out objects in unselected sections.

Thumbnails Preferences

Select Tools > Preferences from the Main menu to display the Preferences dialog and select Design > Thumbnails in the tree view.

To change the background color, click the color drop down to display the Color Picker dialog.

Input (CSV) Window Preferences

Select Tools > Preferences from the Main menu to display the Preferences dialog and select Input (CSV) in the tree view. The following preferences settings are available:
**Input file size**

To limit the size of an input data file to be used as a sample, select the Limit to checkbox and enter the maximum number of lines you want to use. To most accurately represent the sample data, it is recommended that you don’t limit the number of lines unless the number of lines affects the processing speed.

Whenever you open a project where the number of lines exceeds the limit, you will be prompted to truncate the input file.

**Header**

- Select the Show checkbox to display the CSV input file header in the Input Window
- To change the background color, click the Background color field to display the Color Picker dialog. Refer to Colors for more information

**Detail**

- Select the Show row numbers checkbox to display the row number in the Input window
- To alternate the background color between consecutive sets, select different colors for odd and even sets

**Synchronize with design window**

Select the checkbox to synchronize the display in the Input Window with selections made in the Input Window. Refer to Design Window Preferences for more information.

**Input (JDE PDF) Window Preferences**

Select Tools > Preferences from the Main menu to display the Preferences dialog and select Input (JDE PDF) in the tree view. The following preferences settings are available:

**View**

To set the default page view magnification, choose Fit to Width, Fit in Window or Actual Size from the View drop down list.

**Input file size**

To limit the size of an input data file to be used as a sample, select the Limit to checkbox and enter the maximum number of lines you want to use. To most
accurately represent the sample data, it is recommended that you don’t limit the number of lines unless the number of lines affects the processing speed. Whenever you open a project where the number of lines exceeds the limit, you will be prompted to truncate the input file.

**Show rulers**

Horizontal and vertical rulers are available to measure column (character) and row (line) positions on the input page.

**Enable tooltips**

Select the checkbox to display names as the cursor moves over data variables in the Input Window.

**Synchronize with design window**

Select the checkbox to synchronize the display in the Input Window with selections made in the Design Window. You can also separately synchronize the display in the Design Window with selections made in the Input Window.

### Input (JDE PDF) Window Sections Preferences

Select Tools > Preferences from the Main menu to display the Preferences dialog and select Input (JDE PDF) > Sections in the tree view. The following preferences settings are available:

**Section borders**

- Select the Show checkbox to display section borders in the Input Window
- Select the section border line style from the Style drop down

**Selected section**

- Select the Highlight background checkbox to display the selected section in the background color shown. To change the background color, click the background color field to display the Color Picker dialog. Refer to Colors for more information.
- To change the variable background color, click the Variable background color field to display the Color Picker dialog
- To change the selected variable frame color, click the Selected variable frame color field to display the Color Picker dialog
Unselected Sections

- Select the Highlight background checkbox to display unselected sections in the background color shown. To change the background color, click the background color field to display the Color Picker dialog. Refer to Colors for more information.

Input (Text) Window Preferences

Select Tools > Preferences from the Main menu to display the Preferences dialog and select Input (Text) in the tree view. The following preferences settings are available:

View

To set the default page view magnification, choose Fit to Width, Fit in Window or Actual Size from the View drop down list.

Input file size

To limit the size of an input data file to be used as a sample, select the Limit to checkbox and enter the maximum number of lines you want to use. To most accurately represent the sample data, it is recommended that you don’t limit the number of lines unless the number of lines affects the processing speed.

Whenever you open a project where the number of lines exceeds the limit, you will be prompted to truncate the input file.

Grid

A grid can be displayed in the Input Window to assist in mapping sections and data variables. The grid lines correspond to the column and row numbering displayed on the rulers.

User the Grid options to:

- Hide and show a grid
- Show grid as points
- Change the grid line style

Rulers

Horizontal and vertical rulers are available to measure column (character) and row (line) positions on the input page. User the Rules options to:

- Hide or show ruler
CUSTOMIZING

User Preferences

- Change the cursor indicator color

Display font
- Select the Monospace only checkbox to restrict the fonts available in the font drop down
- Select the input file display font in the drop down

Enable tooltips
Select the checkbox to display names as the cursor moves over data variables in the input window.

Synchronize with design window
Select the checkbox to synchronize the display in the Input Window with selections made in the Design Window. You can also separately synchronize the display in the Design Window with selections made in the Input Window. Refer to Design Window Preferences for more information.

Input (Text) Window Sections Preferences
Select Tools > Preferences from the Main menu to display the Preferences dialog and select Input (Text) > Sections in the tree view. The following preferences settings are available.

Section Borders
- Select the Show checkbox to display section borders in the Input Window
- Select the section border line style from the Style drop down

Selected Section
- Select the Highlight background checkbox to display the selected section in the background color shown. To change the background color, click the background color field to display the Color Picker dialog. Refer to Colors for more information.
- To change the selected variable frame color, click the Selected variable frame color field to display the Color Picker dialog

Unselected Sections
- Select the Highlight background checkbox to display unselected sections in the background color shown. To change the background
color, click the background color field to display the Color Picker dialog. Refer to Colors for more information.

- To change the variable background color, click the Variable background color field to display the Color Picker dialog

### Input (XML) Window Preferences

Select Tools > Preferences from the Main menu to display the Preferences dialog and select Input (XML) in the tree view. The following preferences settings are available:

**Tooltips**

You can choose to display tooltips when the cursor hovers over an XML element, and optionally, the tooltip will display values, variable and section names, and the types of node.

**Initial Document View**

When the document is opened, choose if the whole tree is expanded, or only the first level.

**Colors**

Select a component of the tree to display the current color setting. To customize colors, clear the Use system color checkbox and click the color field to display the Color Picker dialog.

**Synchronize with Design Window**

Select the checkbox to synchronize the display in the Input Window with selections made in the Design Window. You can also separately synchronize the display in the Design Window with selections made in the Input Window. Refer to Design Window Preferences for more information.
Spelling and Language

Create!form Designer is able to check each section for spelling errors occurring in Create!form Designer text objects. You cannot check the spelling of variable data that is mapped from the input data file.

- Checking Spelling
- Languages
- User Dictionaries

Checking Spelling

To check spelling
1. Select the section you want to check spelling for.
2. From the Main menu, select Tools > Spelling. To only check specific text objects in a section, select those text objects to spell check. If no text objects are selected, Create!form Designer will check the entire section.

Check Spelling dialog options

Table 13.1:

<table>
<thead>
<tr>
<th>Dialog Item</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not in Dictionary</td>
<td>Indicates that a misspelled word was found. The word is considered misspelled because it was not located in any open dictionaries, or was found in an exclude dictionary.</td>
</tr>
<tr>
<td>Ignore</td>
<td>Causes this occurrence of a misspelled word to be skipped.</td>
</tr>
<tr>
<td>Ignore All</td>
<td>Causes this and all further occurrences of a misspelled word to be skipped.</td>
</tr>
<tr>
<td>Change To</td>
<td>Contains a word which will replace a misspelled word when you select the Change or Change All buttons.</td>
</tr>
<tr>
<td>Suggestions</td>
<td>Contains a list of suggested replacements for the word reported as misspelled. A word selected in this list will automatically be copied to the Change To field, where it can be substituted for the misspelled word by clicking the Change button.</td>
</tr>
<tr>
<td>Change</td>
<td>The misspelled word will be replaced with the word in the Change To field.</td>
</tr>
</tbody>
</table>
Languages

Create!form Designer can check spelling in a project using any installed language. For a Typical installation, English - US and English - UK are installed:

<table>
<thead>
<tr>
<th>Dialog Item</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change All</td>
<td>This and all following occurrences of the misspelled word will be replaced with the word in the Change To field.</td>
</tr>
<tr>
<td>Suggest</td>
<td>Causes a group of suggested replacements for misspelled words to be displayed in the Suggestions list.</td>
</tr>
<tr>
<td>Add</td>
<td>Causes the misspelled word to be added to the dictionary selected in the Add words list.</td>
</tr>
<tr>
<td>Add Words to</td>
<td>Indicates which user dictionary words will be added to when you click the Add button.</td>
</tr>
<tr>
<td>Dictionaries</td>
<td>Display the Dictionaries dialog. Use this feature to open or close user dictionaries and to edit the contents of user dictionaries. For more information on the dictionary, click Help in the Dictionaries dialog.</td>
</tr>
<tr>
<td>Options</td>
<td>Display the Options dialog to customize the way the dictionary works.</td>
</tr>
<tr>
<td>Lock Pos.</td>
<td>Locks the position of the Check Spelling dialog in place.</td>
</tr>
</tbody>
</table>

To install another language from the installation CD

1. Insert the installation CD and navigate to the \utilities\Language directory. This directory contains a separate sub-directory for each of the available languages.
2. Locate the sub-directory for the language required.
3. Copy the language directory from the installation CD into the <install dir>\Win32User\MainLexicon directory.
4. The language(s) will be available the next time you start Create!form Designer. To use the new language(s), make the settings in the Preferences dialog.
User Dictionaries

There are four types of dictionary. When you create a new dictionary, you must specify the type. The type indicates what happens when a word is found in the dictionary during a spelling check as explained below:

<table>
<thead>
<tr>
<th>Dictionary type</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto Change</td>
<td>Words found in an Auto Change dictionary are automatically replaced with other words. Typically, Auto Change dictionaries hold frequently misspelled words and their correct replacements. Each entry in an Auto Change dictionary contains a word and the replacement, separated by a colon (e.g., “recieve:receive”).</td>
</tr>
<tr>
<td>Conditional Change</td>
<td>Words found in a Conditional Change dictionary are presented to you as candidates form replacement and are replaced with other words if you request.</td>
</tr>
<tr>
<td>Exclude</td>
<td>Words found in an Exclude dictionary are always considered misspelled, even if they are defined in other dictionaries.</td>
</tr>
<tr>
<td>Ignore</td>
<td>Words found in an Ignore dictionary are considered correctly spelled.</td>
</tr>
</tbody>
</table>
Fonts

Create!form Designer supports a pre-defined set of PostScript fonts. The fonts supported are based on the fonts which are typically released and supported on the majority of PostScript printers. Create!form Designer allows you to extend the list of supported fonts.

Each font used in Create!form Designer is mapped to a Windows font so it can be displayed on the screen. If you have other Windows fonts, for example AvantGarde or Bookman, you can configure Create!form Designer to use that Windows font for display instead.

Special fonts, such as MICR for check printing, will need to be ordered directly from a Create!form distributor.

To install Asian fonts, you must select them using the Custom install. Or, if you have already installed Create!form Designer, you can install specific components from the installation CD, by selecting Modify from the Installshield Wizard.

What do you want to do?

- Add Fonts
- Change Font Mapping
- Remove Fonts

Add Fonts

Only PostScript fonts with .afm or .pfm files can be added. A PostScript font can be added even if it is not installed in Windows, so long as the .afm and .pfm files for the font are available.

Adding fonts is a two-step process. You should first install the font to the Windows font directory and then to the Create!form Designer font table. This section takes you through this process.

To add a font in Create!form Designer

1. Add the font into Windows, using the standard Microsoft Windows font installation. This step is optional as you can still print the font without it being installed in Windows.
2 From the Create!form Designer Main menu, select Tools > Fonts > Add. The Import Fonts dialog will be displayed.

3 Change the directory to the location of the .afm or .pfm files for the font.

4 Select the font(s) to install and click Open.

When a font is installed into Create!form Designer it requires a Windows font to be associated with it. This allows it to be displayed properly. If you have correctly installed the font on Windows you will be prompted to associate the font with the Windows font name. If you do not associate the font with the Windows font name, Windows will choose the most appropriate font for display in the Design Window of Create!form Designer.

Change Font Mapping

Create!form Designer allows you to change the Windows fonts that will be used to display text for PostScript typeface. Changing the display font does not change the typeface that will be printed nor does it change the font metrics on the screen for that text. This command is provided so that the most appropriate installed Windows font can be used to display the PostScript typeface in the form Design Window.

---

**Note**

*If the font mapping between the on-screen and printed fonts are not compatible, the on-screen layout and positioning of text objects may be different to the printed version.*

---

**To change font mapping**

1 Select Tools > Fonts > Mapping. The Map Fonts dialog will be displayed.

2 Select the PostScript Font to change the Window mapping for.

3 Select the font you will map to in the Uses Windows Font list. To increase the size of the sample font, select a different size from the Size of Sample field.

4 Click Apply to test the font mapping.

5 Click **OK** when finished.
Remove Fonts

Create!form Designer also allows you to remove PostScript fonts which are not required. If you remove a font from within Create!form Designer it will prevent the use of that font when creating or editing a text of variable object. If you open a project containing a reference to a font that has been removed, Create!form Designer will warn you that the font is available. When the project is opened, all the affected text and variable text objects will have their font style defaulted to Courier.

To remove a font

1. Close any open projects.
2. From the Main menu, select Tools > Fonts > Remove. The Remove Fonts dialog will be displayed.
3. Select the font(s) to remove. You can select multiple fonts by holding down the Ctrl or Shift keys as you point and click.
4. Click Remove.
Colors

Create!form Designer allows you to work with the colors you use most often in your project and mix alternative colors. The default color palette provides a standard range of colors to choose from. The current color selection on a dialog option will be shown on the color button.

To display the Color Picker dialog, click the color drop down. The default color palette is shown on the Basic tab. To customize the color palette, select the Custom tab on the Color Picker dialog and use the controls to mix the colors you want.

![Color Picker Dialogues](image)

*Figure 13.1: The Color Picker dialogues*
GLOSSARY

This appendix provides a list of commonly used Create!form Designer terms.
Glossary

The following table explains many of the terms specific to Create!form Designer.

Table A.1:

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>alias</td>
<td>An alternative name for a source or child section used to resolve problems with duplicate variable names in child and source sections.</td>
</tr>
<tr>
<td>alternate section</td>
<td>A section that alternates with a data section when user-defined conditions are met. The characteristics and content of an alternate section are independent of the section they alternate for.</td>
</tr>
<tr>
<td>archive variable</td>
<td>A variable defined in the project for use by the Create!archive output module.</td>
</tr>
<tr>
<td>argument</td>
<td>The component of a function or expression that defines a value.</td>
</tr>
<tr>
<td>Common Project directory</td>
<td>The directory used to store shared project resources.</td>
</tr>
<tr>
<td>condition</td>
<td>A user-defined logical test that results in either a true or false value. Conditions enable the project to behave differently according to the content of the input file or the production environment.</td>
</tr>
<tr>
<td>data section</td>
<td>A section that is brought into the project from the input file.</td>
</tr>
<tr>
<td>DataMap</td>
<td>A DataMap records how the different parts of an input file are identified and labelled.</td>
</tr>
<tr>
<td>data variable</td>
<td>A variable defined in the input file.</td>
</tr>
<tr>
<td>data copying</td>
<td>The creating of variable text objects in the project, using data variable from the input file.</td>
</tr>
<tr>
<td>derived variables</td>
<td>A user-defined expression, defined in the input.</td>
</tr>
<tr>
<td>detail section</td>
<td>A section in the input file containing repeating line items.</td>
</tr>
<tr>
<td>e-forms server</td>
<td>The system that hosts Create!form Server which merges the project with incoming spool files.</td>
</tr>
<tr>
<td>enterprise server</td>
<td>ERP system or other application that is the originator of the input files.</td>
</tr>
<tr>
<td>expression</td>
<td>A user-defined statement that performs calculations and derives values from the input data.</td>
</tr>
<tr>
<td>form project</td>
<td>See project.</td>
</tr>
<tr>
<td>function</td>
<td>A pre-defined formula that performs a calculation from user defined arguments</td>
</tr>
<tr>
<td>global user variable</td>
<td>A user-defined variable, global to the project.</td>
</tr>
<tr>
<td>input file</td>
<td>A sample file from the enterprise server, used to build the DataMap.</td>
</tr>
</tbody>
</table>
### Glossary

**Table A.1:**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>inserted section</td>
<td>An inserted section is created by the user and attached to a data section.</td>
</tr>
<tr>
<td>lookup variable</td>
<td>A variable that derives its value from an external database.</td>
</tr>
<tr>
<td>standard project</td>
<td>A project that utilizes all the design features of Create!form Designer object.</td>
</tr>
<tr>
<td>object</td>
<td>Text, graphic or barcode object inserted in the project.</td>
</tr>
<tr>
<td>operator</td>
<td>A component of an expression or condition that defines an arithmetic, logical or text operation on adjacent arguments.</td>
</tr>
<tr>
<td>overlay project</td>
<td>A project where the data from the input file will be displayed in the project without change.</td>
</tr>
<tr>
<td>project</td>
<td>A project is a record of the way you want to redesign the input file. Also referred to as a form project.</td>
</tr>
<tr>
<td>project directory</td>
<td>The directory where the project resources are stored.</td>
</tr>
<tr>
<td>repagination</td>
<td>The process of expanding or condensing pages in the project depending on the space required.</td>
</tr>
<tr>
<td>repeated section</td>
<td>A data section that appears more than once in a set.</td>
</tr>
<tr>
<td>set</td>
<td>A user-defined construct that divides the input file into its basic document units, that the project is designing.</td>
</tr>
<tr>
<td>simple project</td>
<td>A project where only limited reformatting of the input file is required.</td>
</tr>
<tr>
<td>source</td>
<td>Where a section derives its data from. The source of a section can be a section from the input file, or another section in the project.</td>
</tr>
<tr>
<td>subform</td>
<td>A subform is a user defined graphical object.</td>
</tr>
<tr>
<td>system variable</td>
<td>System variables are predefined and relate to the project. They include such items as the number of pages in the set or document.</td>
</tr>
<tr>
<td>table</td>
<td>A table is a user created structure that enables you to better organize and display a series of consecutive data sections.</td>
</tr>
<tr>
<td>template</td>
<td>A project used to define other projects.</td>
</tr>
<tr>
<td>user variable</td>
<td>User variables are user-defined expressions that relate only to the section they were created in.</td>
</tr>
</tbody>
</table>
KEY INFORMATION WHEN UPGRADING

This chapter provides a reference to the information you need when upgrading from earlier versions of Create!form Designer.

The following topics are covered:

• What’s New in Create!form Designer?
What’s New in Create!form Designer?

This history lists the changes since the release of Create!form Designer 6.0.

Basic Features

The user interface has been given a face-lift (improved in 6.1)

In-line with recent changes to user interface design standards, Create!form Designer has a new look. Apart from the new application and file icons, the most noticeable change in 6.1 are the new style toolbars. We have updated the look of most icons and changed the appearance of a few to help you to quickly find the tool you need. The toolbars have some new features which will help you locate your favorite tools when space is restricted on your screen. You will also find that the icons in the various tree views have been improved and extended, providing you with better visual clues to the meaning and purpose of each item.

Online help (improved in 6.1)

Help information is now available in both printable PDF and Windows Help formats. The contents of the PDF user guide and the online help are the same, so you can choose the format that best suits the way you work.

Input File Source view (new in 6.1)

Instead of viewing the input file in an external program like Notepad, you can now view the input source file directly in the project.

The new features have been incorporated into the New Project wizard (improved in 6.1)

The New Project wizard has been updated so that you can now select the following additional options:

Choose between Text, CSV, XML and JDE-PDF input file types.

Choose to create a new DataMap, or copy a DataMap from another project, or select a shared DataMap.
Working with Other Create!form Components

Local file printing emulates server operation (new in 6.1)
You can now test your design without having to transfer the project to the server. Form merge options like scale, shift, attach header/trailer files, remove/keep headers, and PostScript-PDF-PCL output selection can now be set from the Print dialog.
Project Files and Folders

**UserResource folder has been removed (removed in 6.1)**

The contents of the UserResource folder have been moved to the SystemResources folder, and the folder has been removed.

**Common project directory has been renamed (changed in 6.1)**

Formerly called the “common form project” directory, this is now called the “common project” directory and has been renamed from \CommonFormProject to \CommonProject. When 6.1 is installed, if CommonFormProject exists, it will be automatically renamed.

**Project Template File type has been removed (removed in 6.1)**

The option to save as a Project Template File type has been removed from the Save As dialog.

**Templates folder has been removed (removed in 6.1)**

Formerly, templates were located in the <install dir>\CommonFormProject\Templates folder. Now any project, in any folder can be used as a template.

**There are now multiple types of DataMap files (extended in 6.1)**

A different DataMap file is created for each input file type and is identified by its filename extension:

- .dmp6  DataMap for a text input file
- .cmp6  DataMap for a CSV input file
- .xmp6  DataMap for an XML input file
- .jmp6  DataMap for an JDE PDF input file

**Preview filename includes project and user name (changed in 6.1)**

When the project is previewed in Adobe Reader, the PDF is given an extended filename, which includes both the project name and the user name.
KEY INFORMATION WHEN UPGRADING

Projects

DataMaps can now be shared between projects (new in 6.1)

One of the advantages in separating the input design from the output design has been realized with the introduction of sharable DataMaps. This modularity means that DataMaps can now be shared not just between Create!form projects, but also with Create!form Director and Create!form Transform projects."

This has resulted in changes to numerous areas of Create!form Designer:

- New Project wizard
- Change DataMap command
- Saving new projects - checks for name conflicts with shared DataMaps
- Next Copy projects
- Apply Design Template command

Apply Design Template command (changed in 6.1)

Previously, when a project was applied as a template, both the input (DataMap) and output design (sections, variables, tables, objects and page styles) were applied to the new project. The application of the input and output designs have now been separated. The Apply Design Template command will now only change the output design. To apply the input design, use the new Change DataMap command (see below).

Change DataMap command (new in 6.1)

With this command, you can choose to copy another DataMap, or use a shared DataMap from another project, or share the DataMap in the current project with other projects.

Next Copy projects and shared DataMaps (changed in 6.1)

The ability to share DataMaps has resulted in some changes to the selection of Next Copy projects. Namely, only projects with shared DataMaps can have Next Copy projects, and both the parent and the Next Copy project must use the same shared DataMap.
Input Data

CSV is now a special type of input file (new in 6.1)
Although CSV (comma separated values) files could previously be handled as text files by using the retrieve function, CSV is now a special input file format. The properties of CSV files are used to simplify how headers, columns and rows are mapped to the project. The commands available from the Input menu now depend on the type of input file used by the project.

XML is now a special type of input file (new in 6.1)
This new input file type makes it easy to process any XML file. The structure of the XML file, and the data it contains, are quickly translated into data variables, sections and sets in the output design.

JDE PDF is now a special type of input file (new in 6.1)
The JDE PDF format supported by earlier versions of Create!form is now also supported in Create!form 6.1.

Ignore bold and underline formatting (new in 6.1)
Text only. When processing OS/400 spool files (typically in overlay projects), if the spool file contains bold and underline overstrike control characters, you can choose to either ignore overstriking or convert the overstrike characters into PostScript. Previously this option could only be set in the writer properties on the server.

Show All Instances command (new in 6.1)
Text only. A new command on the Input menu that examines all instances of a section type, displaying an X where the section is populated with text. This command will help you to quickly assess your section definitions and the positioning and size of data variables.

Input file banners and headers (new in 6.1)
CSV and Text only. An additional input section (InputFileHeader) can be enabled from the Input File Properties dialog to capture non-repeating file banners or header lines.
Sets

Allow set to start on back of page (new in 6.1)

A new property has been added that controls whether a set can start on the back or front of the page.
Sections

**Inserted section (new in 6.1)**
A new section type that can be inserted either before or after a detail section. The inserted section always keeps with the parent section, but has independent user variables. An inserted section has no alternates.

**Start new page (new in 6.1)**
A new section property that will force a page break at each instance of the section.

**Editing hidden sections (new in 6.1)**
When a section is not displayed, the contents of the section can now be edited even though the input file does not contain an instance of the section, or the condition for display of an alternate section is not met, or the file pagination does not create all types of continuation headers and footers.
Tables

No changes.
Objects

Digital signatures (new in 6.1)

A signature placeholder can be inserted for PDF output projects. For more information see + Section 9.2.5."
Variables

System variables (extended in 6.1)
System variables have been extended to include time, user, host, file, print queue and project values. For the full list of system variables, refer to + Section 10.5.”

Environment variables (new in 6.1)
A new class of variables that are common to all projects in the production environment. Environment variables may be the same for all output destinations, or can be changed for individual print queues. Typical uses of environment variables include setting values that are used in all projects, like a company name, or using different page styles in the same project depending on whether the destination is a printer, fax, email or archive server port. For more information, refer to + Section 10.8.”

Job ticket variables (new in 6.1)
All the job ticket values passed through with the spool file, or added by the e-forms server are now available wherever system variables are available in the design. The job ticket variables available to a project must be defined in a configuration file in the Win32User folder. For more information, refer to + Section 10.9.”

Banner and trailer variable functionality
The extensions and additions made to variables (see above), and the new feature for attaching subforms to inserted pages (see also + Section C.1.12), means that the full functionality of banner and trailer variables, available in earlier versions of Create!form, is now provided in Create!form 6.1.
Expressions and Conditions

Boolean operators
No changes

Operators
No changes

Logical functions
Changed (new in 6.1):
Returns true if the value of a variable changes
isnumeric (new in 6.1):
Returns true if the argument is numeric

Numeric functions
aswords (new in 6.1):
Converts a number into words
mod (new in 6.1):
Returns the remainder when dividing one number by another

Text functions
pad (new in 6.1):
Pads the end of a text string with spaces to the same length as a reference string
rfind (new in 6.1):
Finds a text string by searching from right to left in another string
upper (new in 6.1):
Converts a text string to uppercase
lower (new in 6.1):
Converts a text string to lowercase
sentence (new in 6.1):
Converts a text string to sentence case
previousat (new in 6.1):
Returns value from previous page or section.
proper (new in 6.1):
Converts a text string to title case

**Statistical functions**
No changes

**System conditions**
No changes
Page Styles

Page gutters and margins (new in 6.1)
A binding gutter and printer margins can be defined for each page style, and will take into account duplex printing settings for left and right face pages. For more information, refer to +Section 6.2."

Subforms can be attached to inserted pages (new in 6.1)
Subforms can be attached to an inserted page. Each additional subform page generates an additional inserted page, which are recognized in page numbering, duplexing and guttering (see above). Attachments may be conditional, which, in combination with the new environmental variables (see also +Section C.1.10), permits selective attachment of pages based on a print queue name. This will permit the same project to be used for printed, email, fax and archive output, amongst other applications.
Customizing

Changes to the user interface are reflected in the Preferences dialog.
KEY INFORMATION WHEN UPGRADING

- Customizing

Create!form Designer
INDEX

A
About Page Styles 252
About Sections 132
About Tables 150
About Variables 212
Add Detail Sections 98
Add Fonts 275
Add Input Header Sections 94
Adding Drawing Objects to the Page 175
Adding Graphic Objects to the Page 170
Addition Operator 244
Adjust the Input Header Section Size 92
Alternate Sections 139
  Creating 139
  Deleting 140
Archive Variables 222

B
Background
  Adding Text and Graphics 130
Background Page Styles 130
Boolean Operators 232
Building Blocks 232
Building Conditions 231
Building Expressions 228

C
Change Font Mapping 276
Changing the Input Data Format 68, 114
Changing the Properties of Columns 68
Changing the Properties of Data Variables 113
Checks 13
Collate Options 122
Colors 278
Common Input Section Tasks 87
Common Project Directory 25
   To change the common project directory 26
Comparing Text Strings 245
Conditions 230
Controlling When an Input Section is Generated 90
Copy a DataMap from Another Project 59
Copy Data Variables to the Project 84
Copying Data Variables to the Project 68, 114
Create Background Graphics for Page Styles 257
Create Data Variables 82
Create Data Variables in a Detail Section 85
Create Page Styles 253
Create!form Designer Features 13
Creating a Project 27
Creating Derived Variables 120
CSV File Structure 62

D
Data Sections 133
   Display Options 133
Data Variables 119
   Working With 112
Data Variables and Derived Variables 213
Data Variables in Detail Sections 67
Data Variables in the Input Header Section 67
Databases 13
DataMap Options 29
DataMap Options when Creating a New Project 58
DataMap Options while Designing a Project 58
DataMap Types and Options 57
DataMaps and Next Copy Projects 123
Date Functions 239
Default Page Style 126
Define Page Style Rules 254
Defining
   Sections 111
Defining Sections 66
Defining Sets 64, 80
Deleting Input Sections 91
Design Flowchart 74
Design Flowcharts 31
Design Prerequisites 14
Design Principles 73
Design Window 46
Design Window Page Preferences 265
Design Window Preferences 263
Design Window Sections Preferences 265
Designing for CSV Input Files 62
Designing for JDE PDF Input Files 116
Designing for Text Input Files 70
  Working with the Input File Header Section 92
Designing for XML Input Files 105
Designing from the Top-Down 73
Detail Sections
  Defining 97
Display and Edit Input Section Properties 87
Drag and Drop Variables from the Input Window 169
Duplicates 123
Duplicates and Next Copy Projects 123

E
Edit a Shared DataMap 60
Edit an Embedded DataMap 59
Edit Objects on Sections that are not Displayed 200
Editing and Formatting Objects 181
Embed a DataMap 61
Embedded and Shared DataMaps 57
Emulating Production Printing 38
  To create a merge configuration 38
Environment Variables 224
Examining How Sections are Populated 73
Expressions 228

F
Fonts 275
Format Style
  Setting the Default for New Objects 125
Format Styles
  About 124
  Creating 124
  Customizing 125
Functions 235

G
General Preferences 261
Getting Help 16
  To view and Print the PDF User Guide 16
  To view the online help 16
Global User Variable 216
Glossary 280
Glossary of Terms 279

H
Header and Footer Sections 135
  Continuation 136
  Create and Remove 136
  Create from an Input Section 137
  Create with the Wizard 135
  Display Order 138
How the Page Size is Calculated 77

I
Input (CSV) Window Preferences 266
Input (JDE PDF) Window Preferences 267
Input (JDE PDF) Window Sections Preferences 268
Input (Text) Window Preferences 269
Input (Text) Window Sections Preferences 270
Input (XML) Window Preferences 271
Input Design
   Before You Start 52
Input File Properties 63, 107, 116
Input File Source View 47
Input File Type 28
Input Footer Sections
   Adding 102
   Adjusting Size 101
   Rules for Creating Multiples 104
   Working with 101
Input Sections 72, 119
Input Source Preferences 263
Input tree 47
Input Window 46
Insert Pages Using Page Styles 255
Inserted Pages
   Attaching Subforms 256
Inserted Sections 141
   Inserting before or after another section 141
Installation Requirements 20
Installing Create!form Designer 21
   To install Create!form Designer 21
Invoices 13

J
Job Ticket Variables 225
Jumping Directly to a Set or Page 50

K
Key Information When Upgrading
   Expressions and Conditions 294
   Input Data 288
   Objects 292
   Sections 290
Create!form Designer

Sets 289
Tables 291
Variables 293

Key Information when Upgrading
   Customizing 297
   Page Styles 296
   Projects 287

Key Input Design Concepts 54

L
Languages 273
Lookup Variables 219

M
Make a Detail Section Expandable 99
Make an Input Header Section Expandable 93
Make Data Variables Expandable 86
Managing Project Files 41
Mixed Data Types 243
Moving Between Sections 50

N
Names of Variables 247
Navigating and Editing Expressions 229
Navigating in a Project 49
Nested Functions 246
New Project Wizard 27
Next Copy Project Options 123

O
Objects
   About 162
   About Objects and Sections 163
   About Property Classes 189
   About Subforms 171
   Adding Barcodes to the Page 177
Adding Text to the Page 166
Adding text to the page by copying all Data Variables from an Input Section 167
Adding text to the page by drag and drop from the input window 167
Adding text to the page by using the toolbar 166
Advanced Object Selection 206
Align Objects 204
Barcode Properties 196
Change Number, Date and Currency Formats 184
Change the Display Order 203
Changing the Display Properties 181
Converts Images into Subforms 173
Copy 203
Copy Data Variables from an Input Section 168
Copy from an Existing Subform 180
Copying from other Projects 180
Copying from Other Resources 180
Create Subforms Other Objects 174
Creating 165
  Draw Horizontal and Vertical Lines 176
Display and Edit Object Properties 187
Draw a Box 176
Draw a Grid 177
Draw Circles and Ellipses 177
Draw Lines 176
Edit and Format the Contents of a Text Object 182
Edit the Contents of a Subform 185
Edit the Properties of Multiple Objects 188
Form Field Properties 197
Format a Variable Text Object 183
General Properties 190
Grid Properties 195
Insert a Barcode with a Constant Value 178
Insert a Digital Signature 179
Insert a Single Variable as Text 168
Insert a Variable Barcode 178
Insert an Existing Subform 171
Insert Static Text 167
Insert Static Text Containing Variables 167
Move Objects 201
Resize 202
Rotate 202
Select Hidden Objects 200
Selecting and Editing Objects in Graphics View 199
Selecting in Different Ways 163
Set Default Properties 198
Setting Object Properties 187
Styles Properties 194
Subform Properties 195
Subform Variable Properties 196
Use a Variable to Display Different Subforms 174
Using the Create!form Sample Projects 162
Using Variables to Control 163
Window Display Preferences 164
Opening a Project 36
  To open a project 36
Operators 233
Order of Precedence 245
Other Input File Settings 77

P
Packing Projects 39
  To save a project as a packed project 39
Parsing 246
Picking Slips 13
Printing and Previewing Projects 37
  To preview a project 37
  To print a project 37
Production Prerequisites 15
Project Background 129
Project Directories 25
Project Directory
   To change the default project directory 25
Project Files and Folders 286
Project Properties 122
Project Tree 47
Project Types 29
   Overlay 30
   Simple 30
   Standard 29
Purchase Orders 13

R
Remove Fonts 277
roject 58
Rules for Creating Multiple Detail Sections 101
Rules for Creating Multiple Input Header Sections 96

S
Saving a Project 35
   To save a project 35
Section Borders Preferences 262
Selecting a Different Input File 52
Selecting and Editing Objects in Graphics View
   Select Objects 199
Sets
   Defining 109, 117
Setting Section Properties 142
   About Sources 146
   Adding Sources 147
   Adjusting Size 143
   Auto-expanding and the Keep With Option 145
Fixed and Auto-Expanding Sections 143
Keeping Sections Together 145
Parent/Child Relationships 144
Section Positioning Options 142
Setting Repagination Control 145
Setting the Input Page Size 75
Share a DataMap with Other Projects 59
Spelling
  checking 272
Spelling and Language 272
Starting Create!form Designer 22
Statistical Functions 240
Status Bar 48
Subtraction Operator 244
Summarizing Data in Headers and Footers 248
Syntax and Order of Precedence Rules 243
System Conditions 241
System Variables 217

T
Tables
  Changing Properties 153
  Creating 152
  Creating Borders and Lines 159
  Deleting 154
  Grouping Options 157
  Header and Footer Sections 156
  Parent/Child Relationships 158
  Properties 155
  Setting Pagination Controls 158
  Sorting Modes 155
  Working with 152
Thumbnails Preferences 266
Thumbnails View 47
Transactional Data 13
Transfer Projects to the e-forms Server
  To transfer a project to the e-forms server 40
Transferring Projects to the e-forms Server 40
Tray Selection 127
Understanding the Screen Elements 44
Use a Shared DataMap 60
User Dictionaries 274
User Guide Conventions 17
User Preferences 260
User Variables 214
Using an Existing Project as a Template 28
Using Expressions and Conditions 231
Using the Create!form Sample Project 52
Using the Create!form Sample Projects 132, 151
Using the Expression Builder 228

Variable Properties 192
Variables 241
Viewing the Input File Source 53

What is a Data Variable 55
What is a DataMap? 54
What is a Derived Variable? 55
What is a Project? 24
What is a Section? 55
What is a Set? 55
What’s New in Create!form Designer
  Basic Features 284
What’s New in Create!form Designer? 284
Who is this Guide For? 12
Working with Data Variables 66, 81
  Set Input Data Formats 81
Working with Detail Sections 96
Working with Input Header Sections 92
Working with Multiple Input Sections 90
Working with Other Create!form Components 285
Create!form Designer

X

XML File Structure 105