The Dunning Notice Sample

Adobe® LiveCycle Designer 7.0

About the sample

The Dunning Notice sample demonstrates the behavior of a dynamic PDF form. For information about creating and using dynamic forms, see the Designer Help.

Getting started

To get started, you need the following files:

Form:	Dunning Notice.xdp	The Designer form file located in the Forms folder.
Data:	Dunning Notice Level1.xml Dunning Notice Level2.xml Dunning Notice Level3.xml	The XML data files located in the Data folder. The data file for the Level 3 Dunning notice contains enough invoices to overflow onto a second page.
Image:	Dunning Notice.tif	The image file located in the Images folder.
Output:	Dunning Notice Level1.pdf Dunning Notice Level2.pdf Dunning Notice Level3.pdf	The rendered forms with merged data located in the Outputs folder.

Running the sample

You can run the sample by following these steps.

1. Open the form, Dunning Notice.xdp, in Designer.

2. To specify the preview options, select File > Form Properties. Click the Defaults tab and do the following:

   - In Preview Type, select Print Form.

   - In Data File, browse to the Data folder and select one of the sample data files, for example Dunning Notice Level1.xml.

   - Click OK.

3. To preview the form, click the PDF Preview tab.

Results

The output sample PDF files in the Outputs folder show the rendered form merged with each of the three sets of data. Use these files to compare to the results of the previewed form.

About the form

Requirements

Three dunning levels are available. The information appropriate for each level should be printed along with a list of the unpaid documents.

Features

Form hierarchy structure

The Dunning Notice form is based on the data structure in order to take advantage of the implicit data binding process.

Master page

Two master pages are required. The first one should display the company logo, form title, and static text. This page is the first page to print and has a maximum occurrence value of 1. The second master page is used for every subsequent page. The same static text is displayed as well as the page numbering.

Page numbering

The page numbers are calculated values obtained by inserting run-time properties in a Static Text object.

Flowed content

When data is merged with the form, the subforms are placed one below the other by setting the subforms' parent (dunningNotice) type to Flow Content. The flow direction is Top to Bottom. The subform margins add the extra spacing required between two subforms.

The header, level1, level2, level3, and closing subforms have a flowed content as well. The fields margins add the extra spacing required between two objects.

Subform occurrence

The form includes a number of subforms that work together to properly accommodate the data. The subforms themselves are a mixture of repeating and non-repeating subforms. The minimum occurrence values of all of the subforms (with the exception of the detailHeader and closing subforms) are set to 0 to specify that only the required subform will print. The maximum value of the non-repeating subforms is set to 1. The repeating subforms do not have a maximum occurrence value because the number varies for each set of data. The minimum value of the closing subform is set to 1 because no data is available to instantiate the subform.

Detail header

The detail header displays the column headings and should print before the first detail line. This detail header information should repeat at the top of the page when the detail lines can no longer fit on the current page and flow on to a new page. Setting the minimum occurrence value of the detailHeader subform to 1 enables us to ensure the header is always printed at least once.

Overflow leader

The detailHeader subform is associated with the document subform as its overflow leader. This feature will force the overflow leader subform of the document subform to print every time the

document subform flows onto a new page. Because there are no fields in the detailHeader subform, the binding match is set to None.

Global fields

The currency value is provided once in the data file. By making the currency field global and by using this field in multiple locations, the value is replicated in each occurrence of the field.

Image field

Since the company logo may vary, the logo file is embedded in the data file. An Image Field object is used to display the image.

Maximum number of characters for each field

The maximum number of characters for each field is specified as per data specifications.

Expandable fields and anchor position

To accommodate data values of varying length, some fields have been marked as Expand to Fit, which is the case for the billToAddress, salutation, and closing fields. The anchor position of these fields is set to allow the proper expand direction.

Variables

The closing of every paragraph is the same. For easier maintenance, its value has been defined as a variable, and a script is used to reference this value and display it. Because the field's value is calculated, the binding match is set to None.

Scripting

The address values should be displayed as a block. A JavaScript is used to concatenate the data values. Because the billToAddress field's value is calculated, the binding match is set to None.

Multiple line field

The billToAddress field's Allow Multiple Lines property is enabled so that the calculated value can print on several lines.

Floating fields

The level3 subform contains both text and fields. Using floating fields results in a natural flow of the information. A display pattern can be specified directly on the field and displayed in the paragraph.

Picture patterns

Some values are easier to read by applying a display picture pattern. For example, the level3 subform's deadline value is formatted with a medium-style date format. A binding picture pattern is required because the data value is not provided in a canonical format.

Licenses and Copyrights