

# Introduction to XPages Development Workshop



**TLCC.COM**

## Instructor

---

- **Howard Greenberg**
  - Teaching Notes and Domino since 1994
  - Partner at TLCC, the Leader in Notes and Domino Training
  - And Now, the Leader in XPages Training



## Agenda

---

- Understand XPages Architecture
- Use Domino Designer to create XPages
- Create your first XPage
- Display Domino views and documents on XPages
- Use Core controls to create and edit Domino documents
- Demonstrations of advanced XPages techniques

**Take the mystery out of XPages!**



## TLCC XPages Offerings

- Four XPages Courses
  - JavaScript for XPages Development
  - Developing XPages Using Domino Designer 8.5
  - XPages Development 2 for Notes and Domino 8.5
  - Mobile XPages for Domino 8.5
- Choose your learning method
  - Self-paced
  - Instructor Led Online
  - We come to you

**Filter a Data View based on a Selection Control**

One approach to filtering the contents of a Domino view is to provide a selection control to the mobile user to set the criteria for filtering the contents of the view.

**This is a lesson from TLCC's Mobile XPages course!**

**Planning Sketch for Demonstration Application**

There are two mobile pages in this demonstration. The two images on the left are of the same view page, before and after the user selects a customer from a Combo Box control. The dataView on the view page is rendered after a customer is selected, and is filtered to display only the sales orders for the selected customer. When an order is selected on the view page, the details for the order are displayed on the document page.

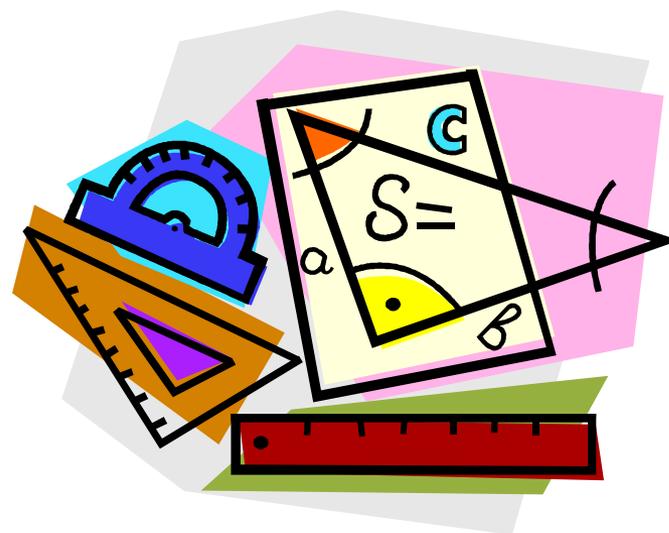
Number	Part ID/Qty	Part Name/Unit Price/Sub-total
1	GSdeluxLR	Golf Set Complete Deluxe Ladies Right
	1	755.00 755.00
2	GSdeluxMR	Golf Set Complete Deluxe Mens Right
	3	730.00 2,190.00

**20% off for Attendees - [www.tlcc.com/entwickler](http://www.tlcc.com/entwickler)**

## Prerequisites

---

- Experienced Notes/Domino Developers
- Minimal to no experience with XPages
- Goal: Convert a basic Notes app to XPages
  - Show a view
  - Display documents
  - Create/edit documents



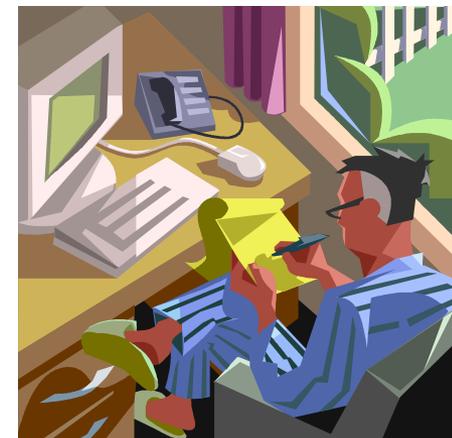
## Hands-on Activities

---

# Will Have Time for Activities In Class

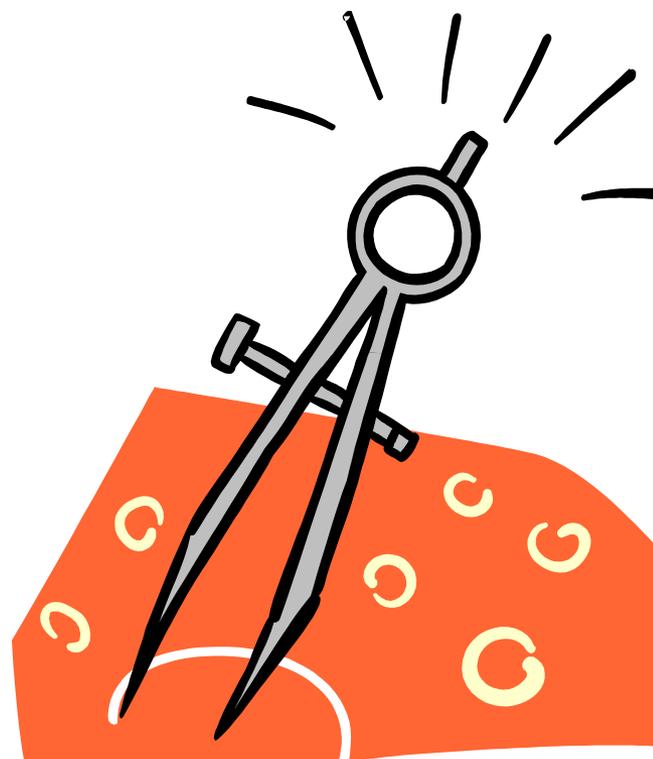
- TLCC's free *Introduction to XPages Development*
  - self-paced course

❖ [www.tlcc.com/free-xpages-training](http://www.tlcc.com/free-xpages-training)



## The Notes Application to be Converted

---



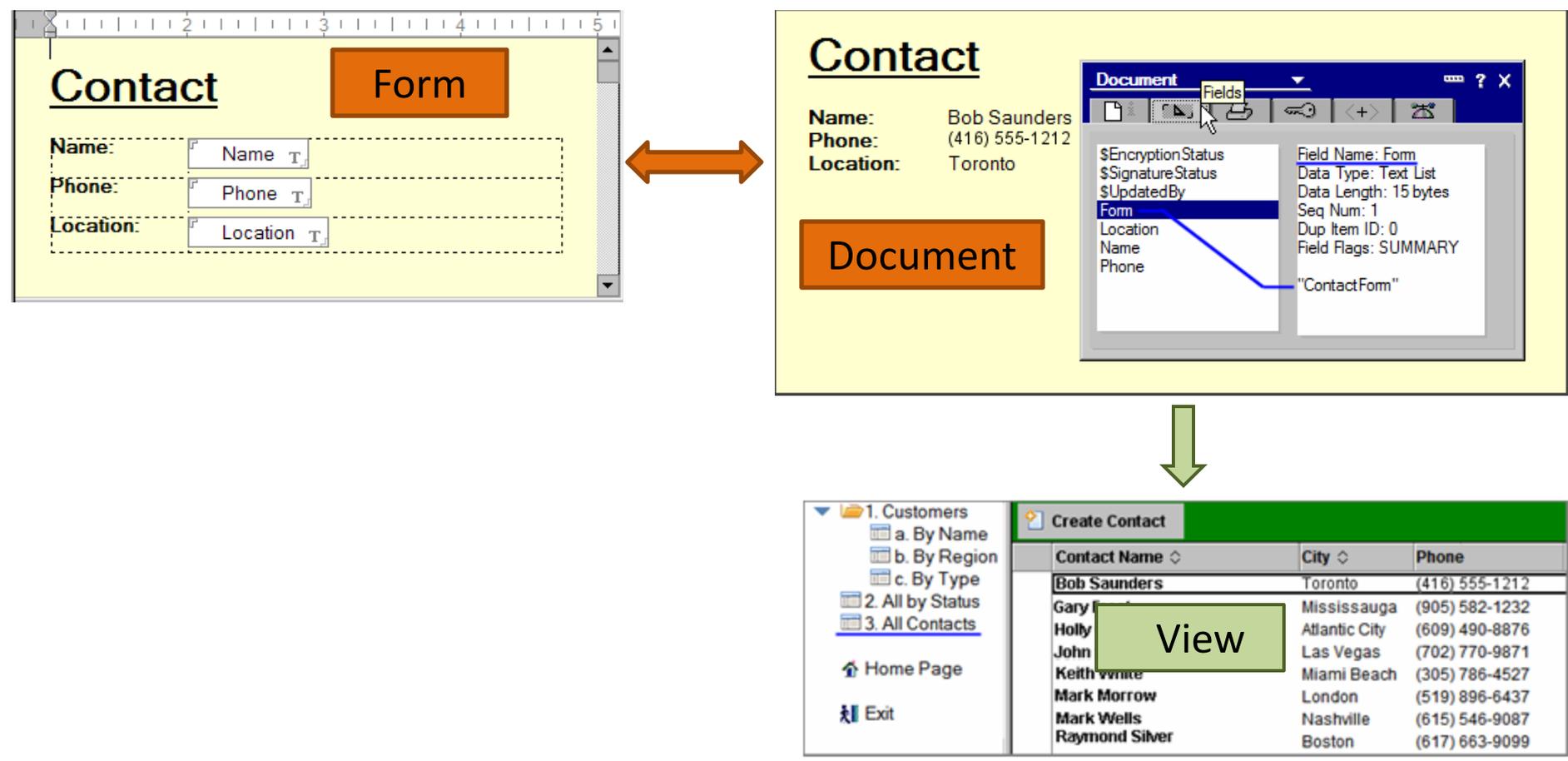
## Setup and Activity Databases

---

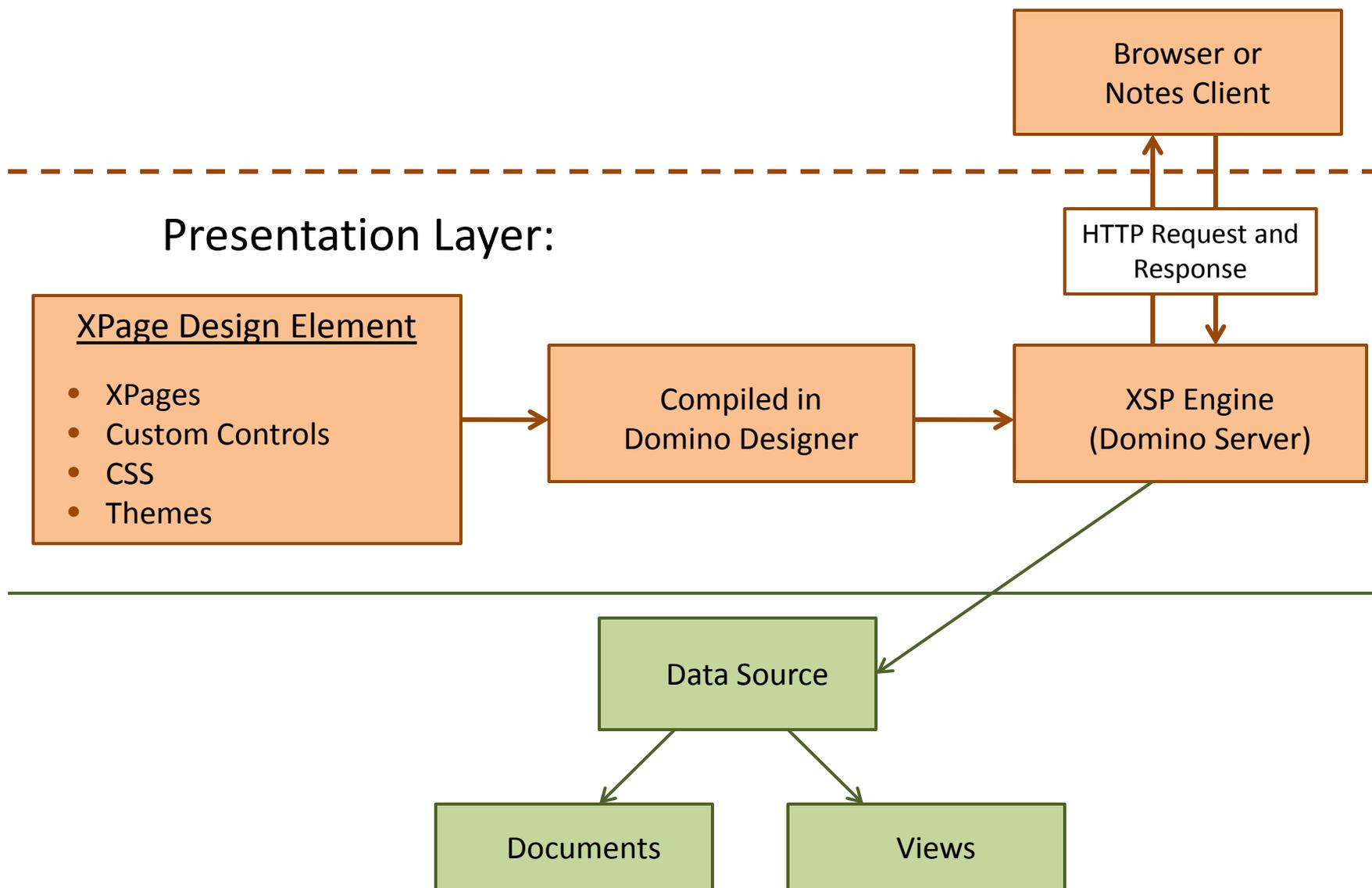


# Traditional Domino Design

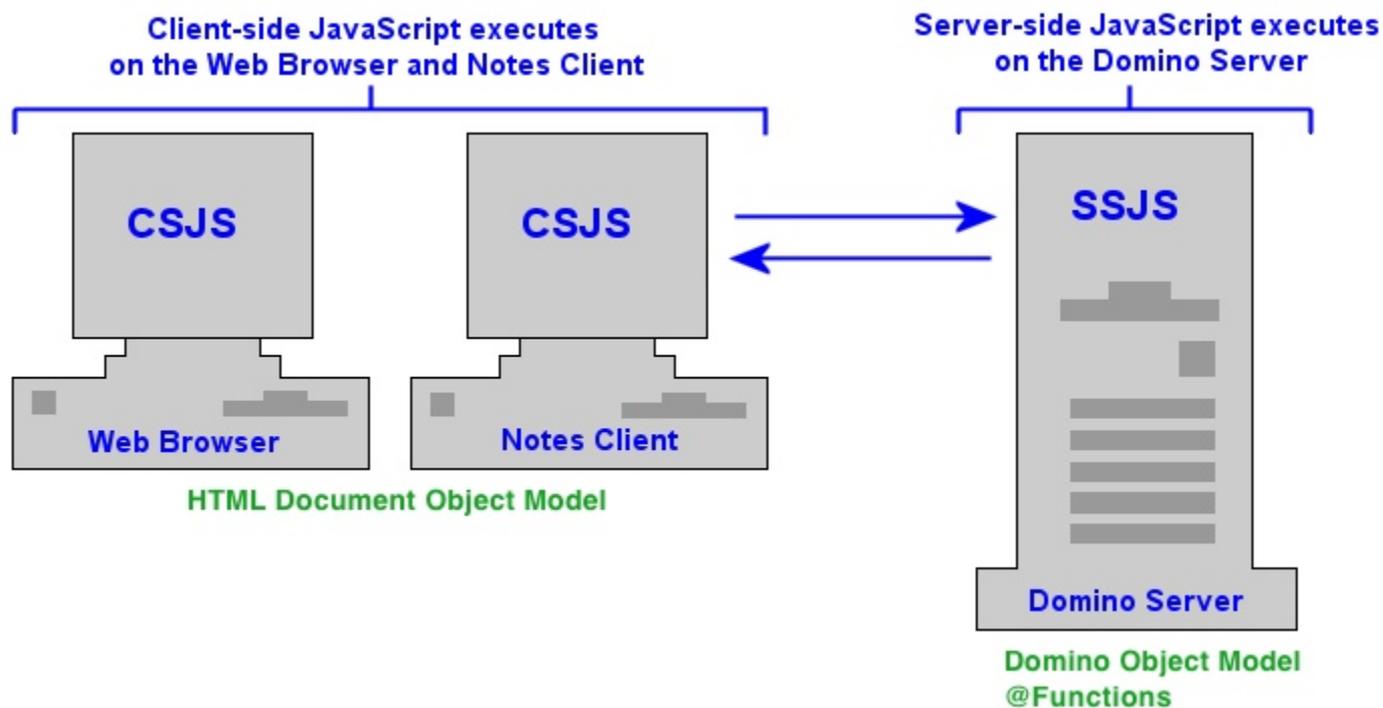
Form and Document have a 1 to 1 relationship



# XPages Architecture



## Client-Side and Server-side JavaScript...



# Programmable Events

The screenshot shows the IBM Design View interface for an XPage. At the top, there is a form with two fields: "Region:" with a dropdown menu and "Sales Representative:" with a text input field containing "SalesRep". Below the form, the "Events" tab is selected, showing a tree view on the left with "onchange" highlighted under the "Select & Change" category. The main area displays the "Script Editor" for the selected event, containing the following JavaScript code:

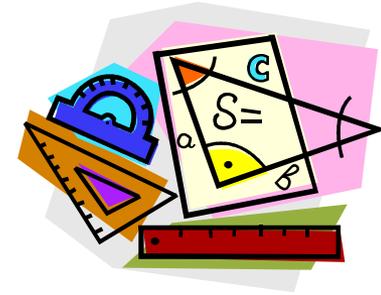
```
var lReg = getComponent("Region1").getValue();
var dbName = new Array("", "tlcc\\85DEV\\ND85XPGS\\XPG85cst.nsf");
var lRep = @DbLookup(dbName, "SalesRepView", lReg, 2);
getComponent("SalesRep1").setValue(lRep);
```

An orange arrow points from a callout box below to the `@DbLookup` function in the code. The callout box contains the text:

**@DbLookup @Function in a Server-side JavaScript event**

## Lesson: Creating an XPage

---



- Domino Designer Perspective for XPage development
- Creating an XPage
- Adding and formatting text on an XPage
- Adding a control to an XPage
- Use the Outline view
- Saving and previewing an XPage
- Troubleshooting errors

# Domino Designer Perspective for XPage Development

**Applications Navigator:**

- Access and organize projects (nsf applications)
- Expand and collapse applications and design elements
- Double-click a design element (XPage) to open

**Applications Navigator**

**Outline Viewer**

**Source Editor**

**Events View**

**Properties View**

**Controls Palette**

**XPage Editor**

**Data Palette**

# Domino Designer Perspective for XPage Development

The screenshot displays the IBM Lotus Domino Designer interface in the XPage Editor perspective. The central workspace shows a form titled "Big Oak Golf Customer" with a computed field `{computedField1}`. Below the title, there is a table of form fields:

Customer:	Customer
Address:	Address
City:	City
State:	State
Zip:	Zip
Other products:	OtherProducts
Phone:	Phone
Region:	Region
Sales potential:	SalesPotential
Sales rep:	SalesRep
Status:	Status

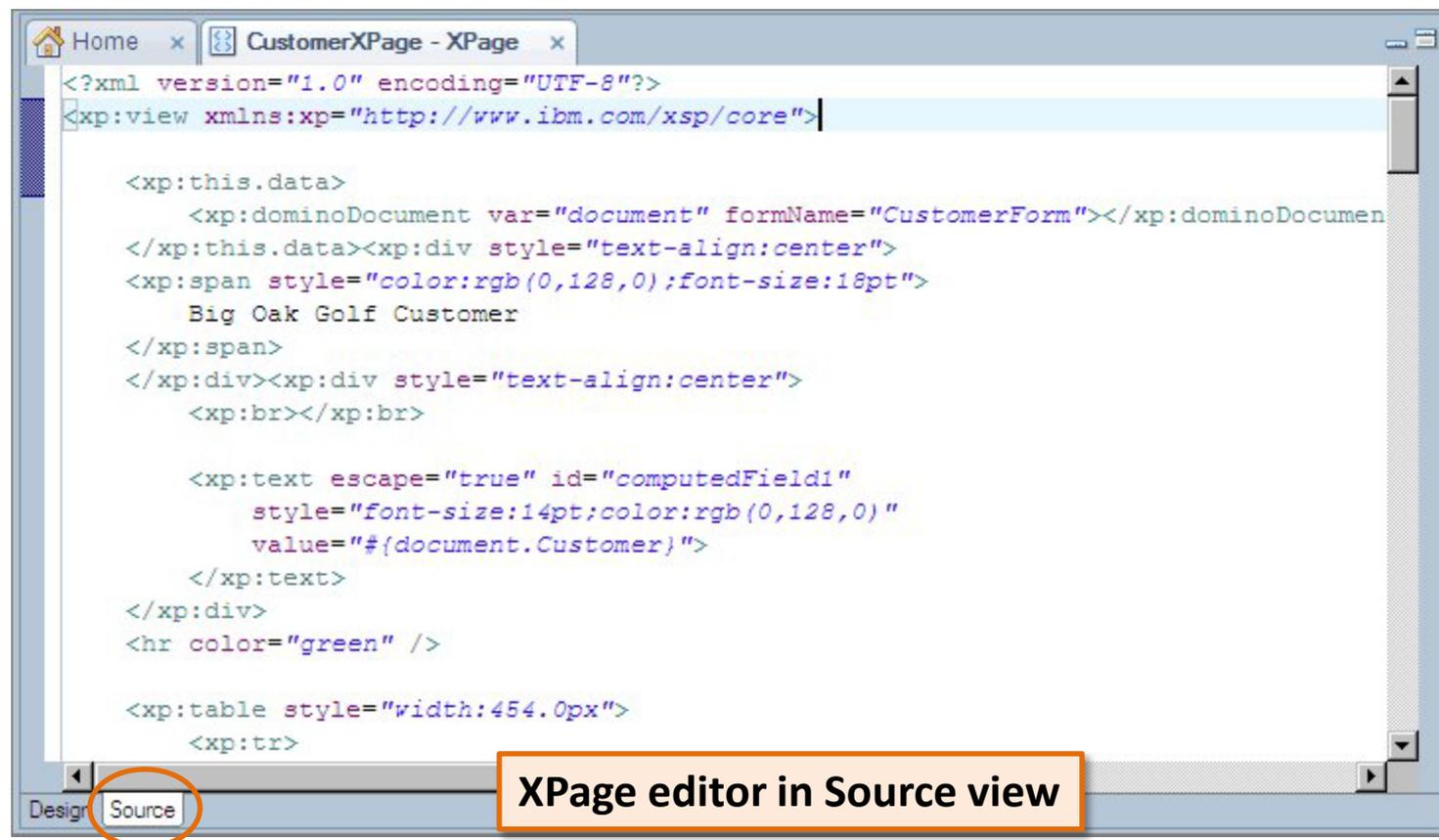
The interface includes several key components:

- Applications Navigator:** Located on the left, showing the project structure for "TLCC" and "ND85XPGS Customers".
- Outline Viewer:** Located at the bottom left, showing a hierarchical view of the form's table structure.
- Design Source Editor:** The central workspace where the form is designed, with tabs for "Design" and "Source".
- Controls Palette:** Located on the right, providing a library of controls such as Edit Box, Rich Text, Button, List Box, and Combo Box.
- Data Palette:** Located at the top right, used for defining data sources and relationships.

**XPage Editor:**

- Controls are dragged from the Controls palette
- Design tab – view and edit design layout
- Source tab – view and edit the source code

## Domino Designer Perspective for XPage Development



# Domino Designer Perspective for XPage Development

## Other Views:

- Outline view – Shows all elements in a hierarchical layout
- Properties view – Shows the properties for the selected element
- Events view – Add CSJS or SSJS to the selected control/element
- Controls Palette – Core, Container, Custom Controls and user-added. Drag & drop to the XPage editor
- Data palette – Lists data elements for the current data source (Document or View). Drag & drop to the XPage editor

The screenshot displays the IBM Lotus Domino Designer interface for XPage development. The main workspace shows a form with fields like 'Sales rep' and 'Status'. Several views are highlighted with blue boxes and labels:

- Applications Navigator:** Located in the top-left pane, showing a hierarchical tree of the project structure, including folders for Forms, Views, Folders, XPages, and Custom Controls.
- Outline Viewer:** Located in the bottom-left pane, showing a hierarchical tree of the current page's structure, including a Table with rows and cells.
- Properties View:** Located in the bottom-center pane, showing the properties for the selected 'Edit Box' control, such as Name, Height, Width, and Tab order.
- Events View:** Located in the bottom-center pane, showing the events for the selected control, with 'Properties' and 'Events' tabs visible.
- Data Palette:** Located in the top-right pane, showing a list of data elements for the current data source (document1), including Customer, Address, City, State, Zip, Phone, Contact, Type, SalesPotential, Region, SalesRep, Status, and OtherProducts.

## Procedure: Creating an XPage

1. Provide a name for the XPage and optional comment.
2. Optionally select the data source for the XPage (Document or View).
3. The required fields to define the data source are displayed.
4. Click [OK] to create the new XPage.

**New XPage**

Create a new XPage.

Name: MyNewXPage

Comment: Use this XPage for testing

Application: Customers : Customers.nsf

Add data source to the page

Data source: Domino View

Data Source: Domino View

Application:

Current Customers.nsf

Other

View: 1. Customers\A. By Name - CustomersByNameView

Data source name: customersByNameView

OK Cancel

## Adding and Formatting Text on an XPage

The screenshot displays the IBM Design Center interface for editing an XPage. The main design area shows the text "Big Oak Golf" in a large, bold, green font, with "Customer Information" below it in a smaller, bold, black font. The "Properties" panel on the left is active, showing the following settings:

- Text
- Style
- Font**
- Background
- Margins
- All Properties

Font settings:

- Family: [Empty]
- Font: Verdana
- Size: 14
- Color: Green
- Style: Bold (b), Italic (i), Underline (u), Strikethrough (ABC), and a refresh icon.
- Alignment: Left, Center, Right, and a refresh icon.

The right-hand side of the interface features a "Controls" palette with two sections:

- Core Controls:** Select, Other..., Edit Box, Rich Text, Multiline Edit Box, Button, List Box, Combo Box, Check Box, Radio Button, Link.
- Container Controls:** Panel, Repeat, Include Page, Table, View, Data Table, Tabbed Panel, Section.

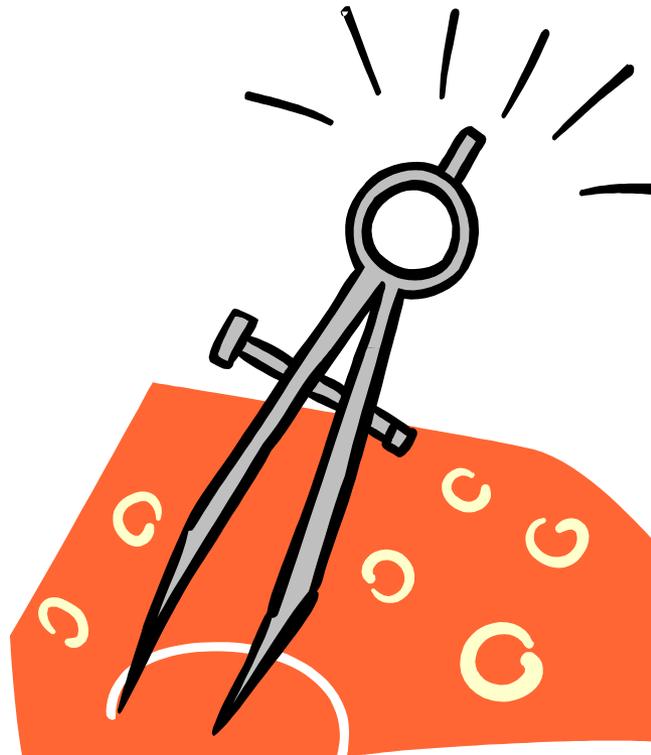
The top of the window shows several tabs: Home, ticc185DEVND85ADUIND85Ucst.nsf - D..., \*NewXpage - XPage, Controls, and Data.

## Procedure: Creating an XPage

---

## Procedure: Adding and Formatting Text on an XPage

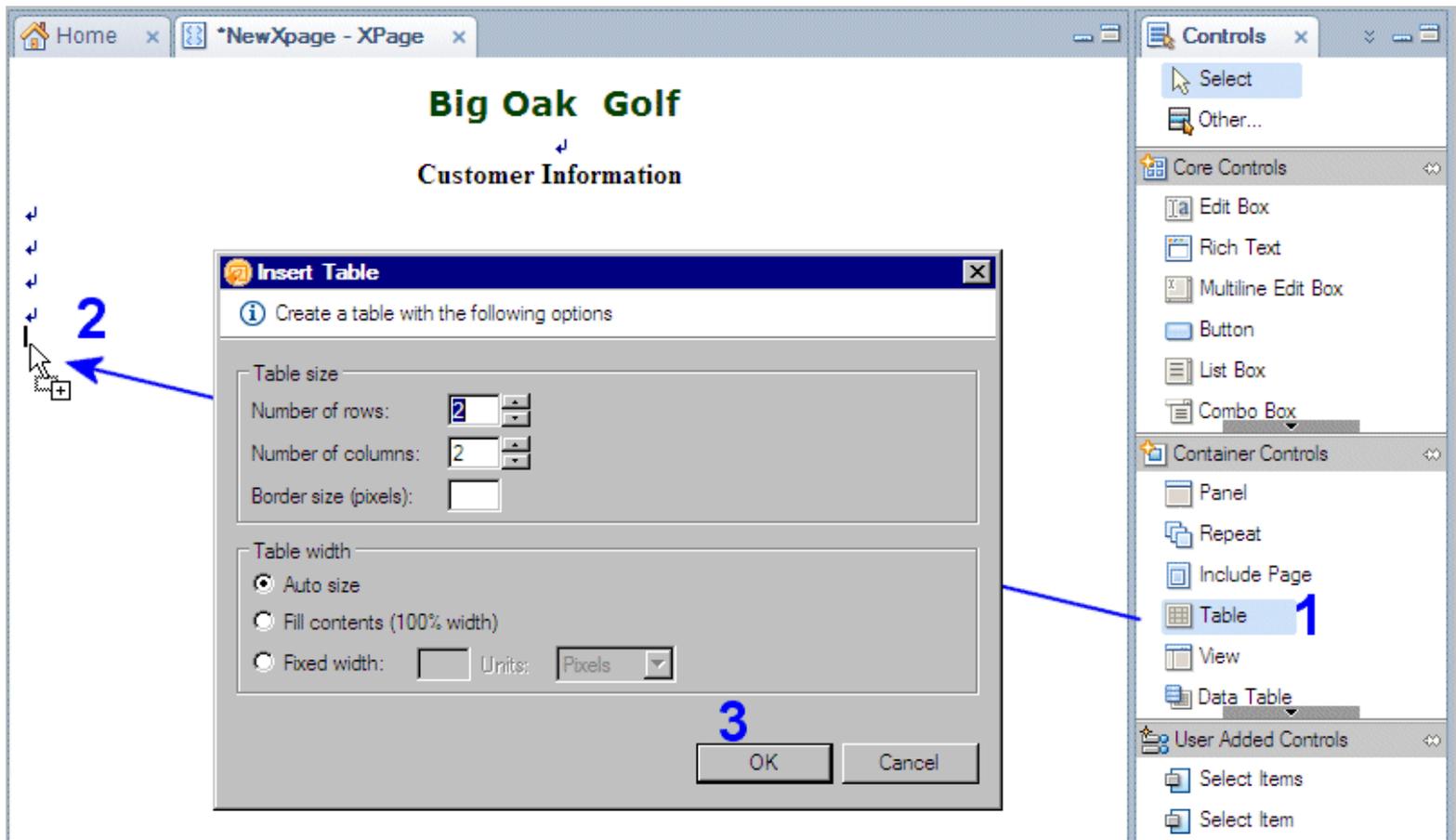
---



## Adding a Control to an XPage

Use any of the following methods to add a control to an XPage:

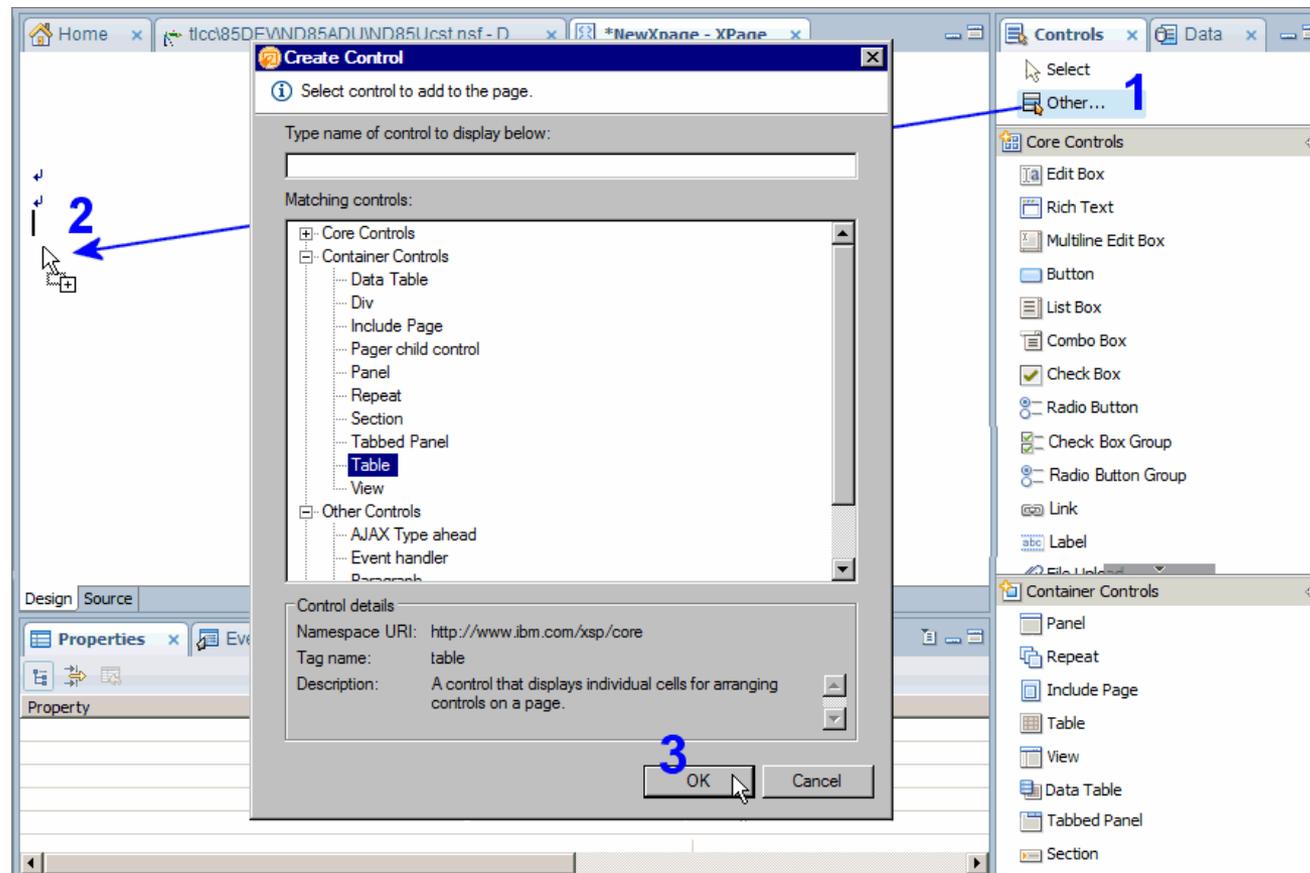
- Drag & Drop from Controls palette



## Adding a Control to an XPage

Use any of the following methods to add a control to an XPage:

- Drag & Drop from Controls palette
- Choose the control from the **Create Control** dialog



## Use the Outline View to Select and Organize XPage Elements

The screenshot displays the IBM WebSphere Designer interface for editing an XPage. The main workspace shows a form titled "Big Oak Golf" with "Customer Information" fields for "Customer:" and "Phone:", and a "Submit" button. The Outline View on the left shows the document structure, with "Edit Box: inputText1" selected. The Properties View at the bottom shows the configuration for the selected "Edit Box", including its name "inputText1", height and width settings, and options like "Visible", "Read-only", and "Disabled".

**Outline View Structure:**

- Root Document
  - XML Processing Instruction
  - XPage
    - Div
    - Div
    - Line break
    - Line break
    - Table
      - Table row
        - Table cell
        - Table cell
          - Edit Box: inputText1
      - Table row
        - Table cell
        - Table cell
          - Edit Box: inputText2
    - Line break
    - Line break
    - Button: button1

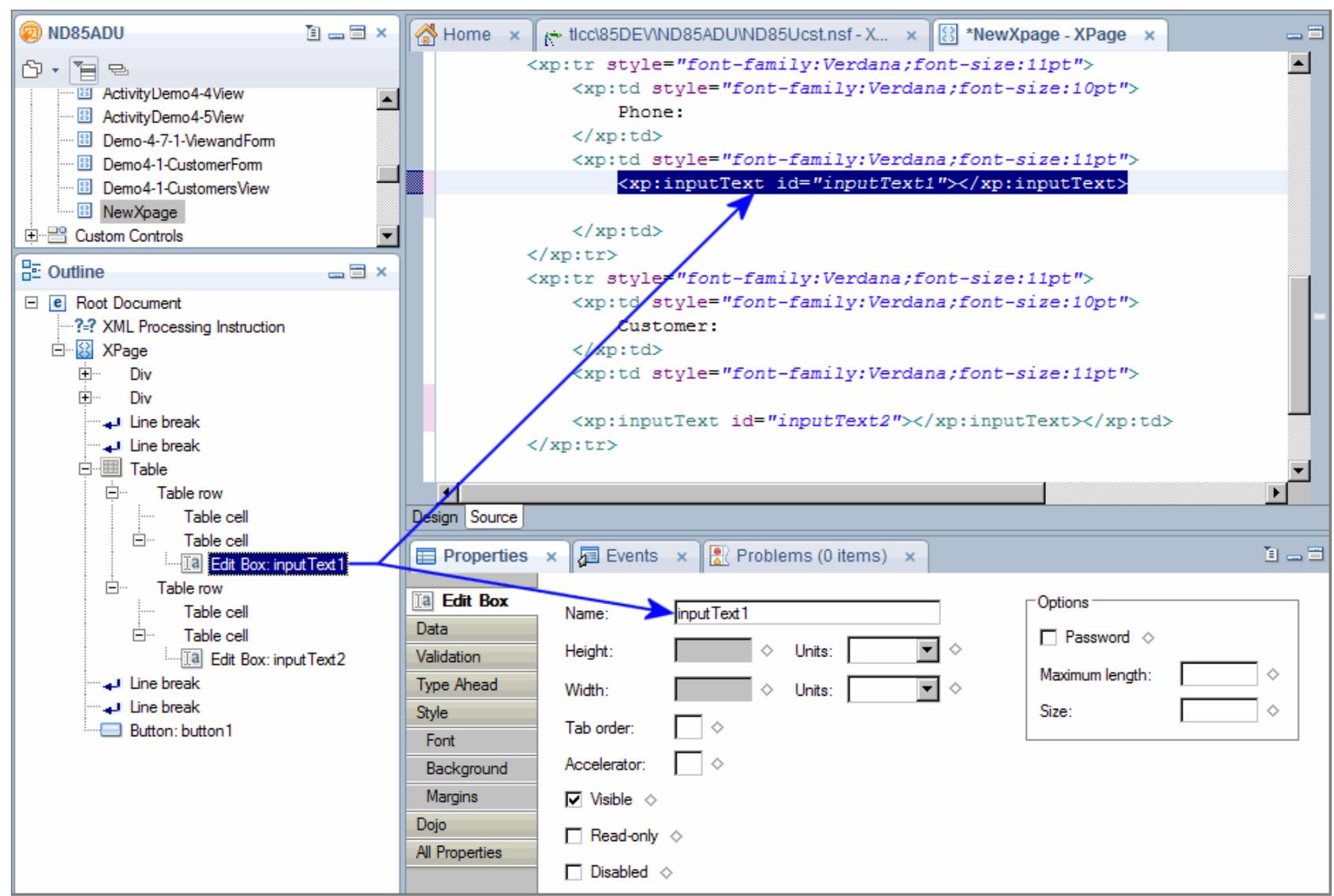
**Properties View for Edit Box:**

- Name: inputText1
- Height: [ ] Units: [ ]
- Width: [ ] Units: [ ]
- Tab order: [ ]
- Accelerator: [ ]
- Visible:
- Read-only:
- Disabled:

**Options:**

- Options:  Password
- Maximum length: [ ]
- Size: [ ]

# Use the Outline View to Select and Organize XPage Elements



## Use the Outline View to Select and Organize XPage Elements

The screenshot displays the IBM WebSphere Designer interface. The top window shows the design view of an XPage titled "Big Oak Golf Customer Information". The page content includes a title, a "Customer:" label with an input field, a "Phone:" label with an input field, and a "Submit" button. The left sidebar contains the Outline View, which shows a tree structure of the page elements. A context menu is open over the "Line break" element in the Outline View, with the "Delete" option selected. The bottom right pane shows the Properties View, which lists various properties for the selected element.

**Outline View Structure:**

- Root Document
  - XML Processing Instruction
  - XPage
    - Div
    - Div
    - Line break (Selected)
    - Table
      - Table cell
        - Edit Box: inputText1
      - Table row
        - Table cell
        - Table cell
          - Edit Box: inputText2
    - Line break
    - Line break
    - Button: button1

**Properties View Table:**

Property	Value
<b>accessibility</b>	
title	
<b>basics</b>	
binding	
id	
loaded	
rendered	
rendererType	
<b>dojo</b>	
dojoAttributes	
dojoType	
<b>styling</b>	
disableTheme	
style	

## Use the Outline View to Select and Organize XPage Elements

Customer:

Phone:

Submit

Second row in table is moving up one row.

Outline View:

- Root Document
  - XML Processing Instruction
  - XPage
    - Div
    - Div
    - Line break
    - Line break
    - Table
      - Table row
        - Table cell
        - Table cell
          - Edit Box: inputText2
    - Line break
    - Line break
    - Button: button1

Property	Value
<b>accessibility</b>	
role	
title	
<b>basics</b>	
binding	
dir	
id	
lang	
loaded	
rendered	
rendererType	
<b>dojo</b>	
dojoAttributes	
dojoType	

# Troubleshooting Errors

The screenshot displays the IBM Lotus Domino Designer interface for editing an XPage named 'NewXpage.xsp'. The code editor shows the following XML structure:

```

<xp:br></xp:br>
<xp:table>
  <xp:tr>
    <xp:td>Phone:</xp:td>
    <xp:td>
      <xp:inputText id="inputText1"></xp:inputText>
    </xp:td>
  </xp:tr>
  <xp:tr>
    <xp:td>Customer:</xp:td>
    <xp:td>
      <xp:inputText id="inputText2"></xp:inputText>
    </xp:td>
  </xp:tr>
</xp:table>
<xp:br></xp:br>
<xp:br></xp:br>
<xp:button value="Submit" id="button1">
  <xp:eventHandler event="onclick" submit="true"
    refreshMode="complete" immediate="false" save="true">
</xp:button>
  
```

The error message in the Problems pane is:

Description	Resource	Path	Location	Type
Errors (1 item)				
The end-tag for element type "xp:table" r	NewXpage.x...	XPG85cst.nsf/XP...	line 24	com.ibm.desi...

Blue arrows labeled 'Error Indicators' point to red error markers on the right side of the code editor, corresponding to the error in the Problems pane.

**Procedure: Adding Controls to an XPage**

---

**Procedure: Working with the Outline View**

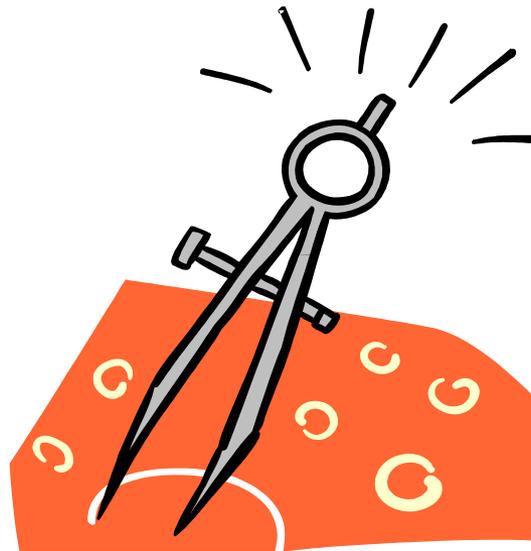
---

**Procedure: Previewing an XPage in a Browser and the Notes Client (XPiNC)**

---

**Procedure: Troubleshooting Errors**

---

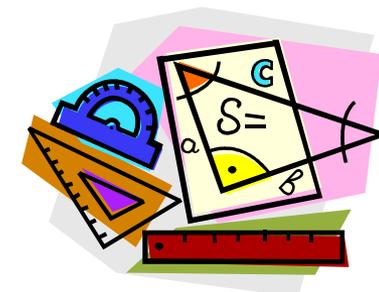


## ... Activity - Creating and Previewing an XPage

---



## Lesson: Display a Domino View on an XPage



- Adding a View control to an XPage
- Setting the View control properties
- Setting the Pager control properties
- Setting View Column Header and View Column properties

Customers By Name << First | < Previous | Page 1 | 2 | 3 | 4 | Next > | Last >>

Customer	City	Contact	Phone
<a href="#">Atlantic City Golf &amp; Country</a>	Atlantic City	Harold Jones	(609) 550-9824
<a href="#">Columbus Golf &amp; Country</a>	Columbus	Michael Morris	(614) 695-9324
<a href="#">Dusty Road Golf Course</a>	Toledo	Mark Leigh	(419) 550-9853
<a href="#">Fox Run Golf Course</a>	Burlington	Mark Morrow	(802) 582-1232
<a href="#">Gallop's Department Store</a>	Northfield	Arnold Gallop	(609) 455-7899
<a href="#">Green Meadows Golf Course</a>	Park Rapids	Dave Gallop	(218) 333-9876
<a href="#">Green Oaks Golf Course</a>	San Jose	Wendy Wase	(408) 298-8693
<a href="#">Greg Nettle's Golf</a>	St. Louis	Greg Nettle	(314) 789-6676
<a href="#">Las Vegas Golf Course</a>	Las Vegas	Mark Wells	(702) 770-9871
<a href="#">Long Bow Golf &amp; Country</a>	Denver	Tan Williams	(303) 559-7765

Internet | Protected Mode: Off 115%

## Procedures: Adding a View Control to an XPage

---

Two methods to create a View Control:

- Method 1: Use the Controls palette
- Method 2: Use the Data palette

## Method 1 - Use the Controls Palette

The screenshot shows the IBM Lotus Domino Designer interface. The main window displays the 'CustomersByNameXPage' XPage. A dialog box titled 'Select Data Source For View' is open, allowing the user to configure the data source for a view control. The dialog includes the following fields and options:

- Show data from:** Domino view
- Data Source:** Domino View
- Application:**
  - Current: Customers.nsf
  - Other: [Browse]
- View:** [Dropdown]
- Data source name:** view1

Below these fields is a table for selecting columns to display:

Programmatic Name	Title

At the bottom of the dialog are buttons for 'OK', 'Select Data Source Later', and 'Cancel'. The Controls Palette on the right shows the 'View' control selected, indicated by a blue arrow labeled '1'. The Outline pane on the left shows the 'CustomersByNameXPage' XPage selected, indicated by a blue arrow labeled '2'. The 'Data Source: Domino View' section of the dialog is highlighted with a blue arrow labeled '3'.

## Method 2 - Use the Data Palette

The screenshot illustrates the process of binding data to an XPage in IBM Domino Designer. The interface is divided into several key areas:

- Design Area:** The top-left pane shows the XPage design. A blue arrow labeled "Drag & Drop" indicates the action of moving data elements from the palette to the page.
- Data Palette:** The top-right pane, titled "Data", shows the "Data source" set to "CustomersByName". It lists available data elements:
  - Customer (Customer) [Text]
  - City (City) [Text]
  - Contact (Contact) [Text]
  - Phone (\$1) [Text]
  - SalesOrderLookupColumn (\$2)
- Properties Window:** The bottom-left pane shows the "Properties" window for the XPage.
  - The "Data sources" list contains "CustomersByName", marked with a blue "1".
  - The "Data Source: Domino View" configuration panel is visible, showing:
    - Application: Customers.nsf
    - View: Name - CustomersByName View
    - Data source name: CustomersByName
    - Filter by category name: (empty)
    - Filter by column value: (empty)

# View Control Properties

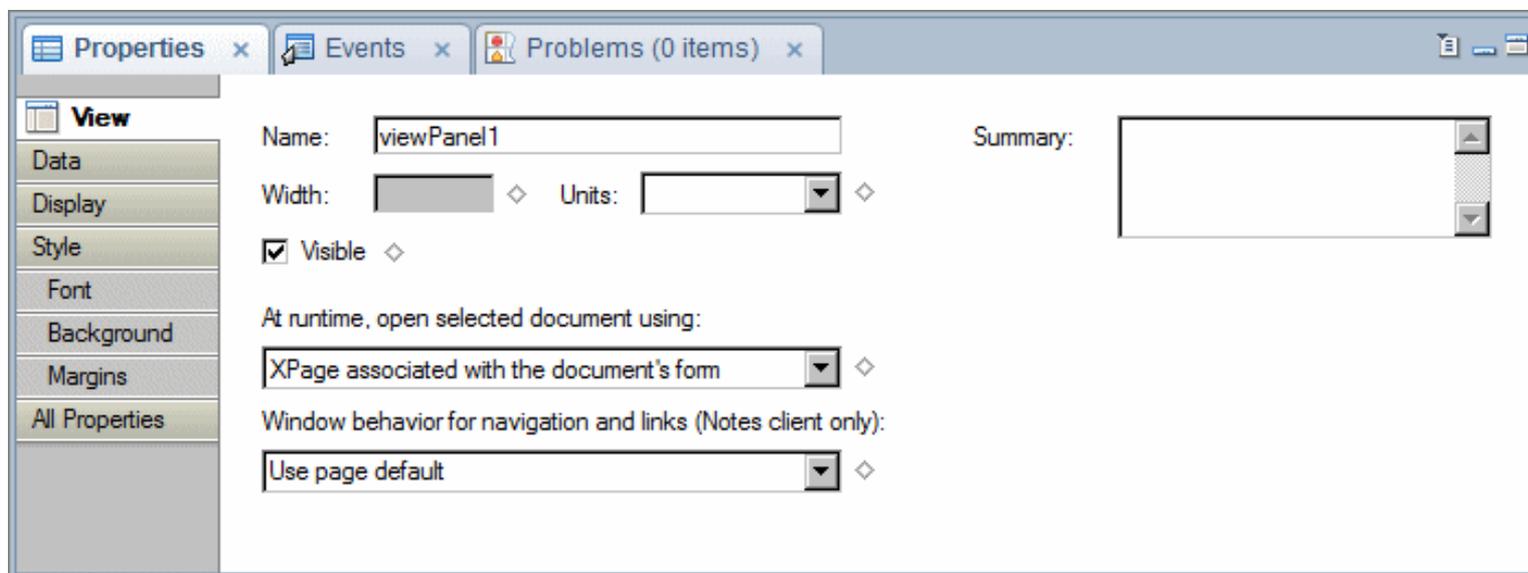
The screenshot displays the IBM Domino Designer interface. On the left, the 'Applications' tree shows the project structure, with 'DemoXPage231View' selected. The 'Outline' pane shows the document structure, with 'View: viewPanel1' selected. The main design area shows a table with columns 'Customer', 'City', 'Contact', and 'Phone'. A blue arrow points from the table to the 'Properties' pane.

The 'Properties' pane for 'View: viewPanel1' is shown below the design area. It includes the following settings:

- Name: viewPanel1
- Width: [ ] Units: [ ]
- Visible
- At runtime, open selected document using: XPage associated with the document's form
- Window behavior for navigation and links (Notes client only): Use page default

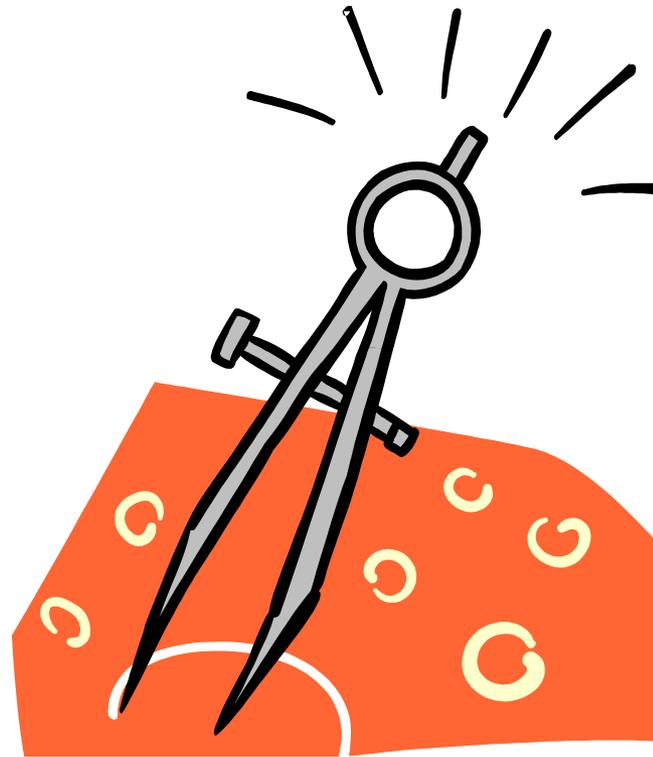
## View Tab

- Basic View control properties are set on the View tab
- Default values for these settings are shown below



## Procedure: Add a View Control to an XPage

---

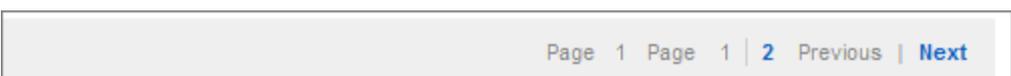
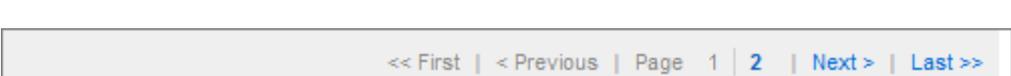


# Pager Control Properties

The screenshot displays the IBM Domino Designer interface for editing an XPage. On the left, the Project Explorer shows a tree structure under 'XPages' with various view files. The Outline pane shows the document structure, including 'XPPage' and 'View: viewPanel1'. The Design view shows a table with columns 'City', 'Contact', 'Customer', and 'Phone'. A pager control is positioned above the table, displaying 'Previous 1|2|3|4|5 Next'. The Properties panel on the right is open to the 'Pager' control, showing the following settings:

- Name: pager1
- Pager style: Sample 1
- Attach to: (empty dropdown)
- Alignment: Default
- Partial refresh:
- Visible:

## Pager Control Properties - 7 Pager Samples

Pager Sample	Example
Pager Sample 1	 <p>Previous 1   2 Next</p>
Pager Sample 2	 <p>Previous   1   2   Next</p>
Pager Sample 3	 <p>Page 1 Page 1   2 Previous   Next</p>
Pager Sample 4	 <p>⏪ ◀ Page 1   2 ▶ ⏩</p>
Pager Sample 5	 <p>Page 1 ⏪ ◀ Page 1   2 ▶ ⏩</p>
Pager Sample 6	 <p>&lt;&lt; First   &lt; Previous   Page 1   2   Next &gt;   Last &gt;&gt;</p>
Pager Sample 7	 <p>Page 1 &lt;&lt; First   &lt; Previous   Page 1   2   Next &gt;   Last &gt;&gt;</p>

## Custom Pager Styles

The screenshot shows the Domino Designer Properties window for a Pager control. The window has three tabs: Properties, Events, and Problems (0 items). The Properties tab is active, showing the following settings:

- Name:** pager1
- Attach to:** (empty dropdown)
- Alignment:** Default
- Partial refresh
- Visible

The **Pager** section is expanded, showing a list of properties: Style, Font, Background, Margins, Facet, and All Properties. The **Style** property is selected, and the **Custom** style is chosen from the **Pager style:** dropdown.

The **Custom: Pager Controls** section is highlighted with an orange border. It contains two lists of controls:

- Available controls:** First, Previous, Next, Last, Page Selector, Current Page, Separator.
- Display in pager:** First, Previous, Separator, Page Selector, Separator, Next, Last.

Arrows indicate the movement of controls between the two lists. The **Page Selector** control is highlighted in both lists. Below the lists is a field for **Number of pages to show in Page Selector:** (empty).

# View Column Header Properties

The screenshot illustrates the configuration of a View Column Header in an XPage. The main workspace shows a table with the following structure:

Customer	City	Contact	Phone
Customer	City	Contact	Phone

The Properties window for the selected 'View Column Header' shows the following configuration:

- Name:** viewColumnHeader1
- Label:** Customer
- Width:** [ ] Units: [ ]
- Content type:** Text
- Sort column
- Check box
- Icon
- Header image: [ ]
- Alternate text: [ ]

# View Column Properties

The screenshot shows the IBM Domino Designer interface. On the left, the 'XPages Intro' tree view shows a project structure with 'DemoXPage233View' selected. Below it, the 'Outline' view shows the 'View column: viewColumn1' element selected under a 'View: viewPanel1' container. The main workspace displays a table titled 'Customer By Name' with the following structure:

Customer	City	Contact	Phone
Customer	City	Contact	Phone

The Properties window at the bottom right is open to the 'View Column' section, showing the following configuration for 'viewColumn1':

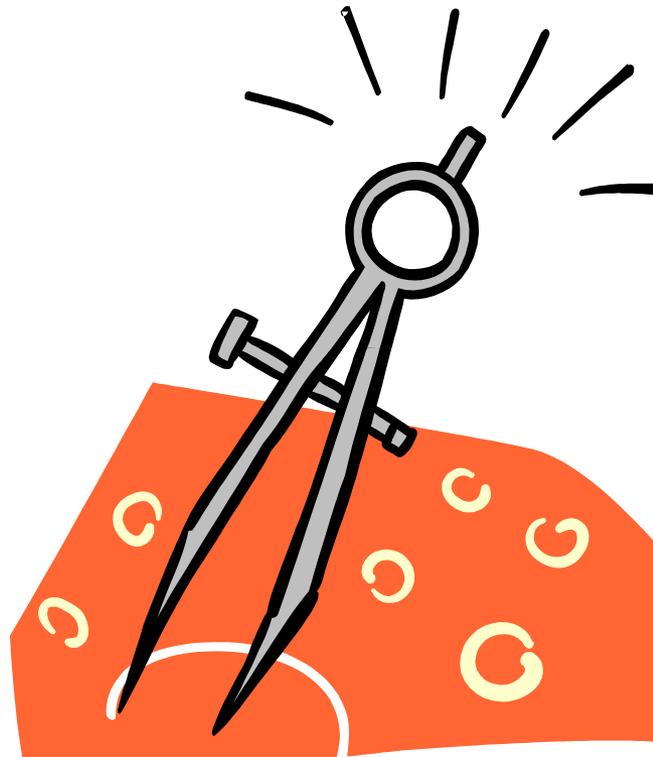
- Name: viewColumn1
- Width: [ ] Units: [ ]
- Visible:

## Procedure: Pager Control Properties

---

## Procedure: View Column Properties

---

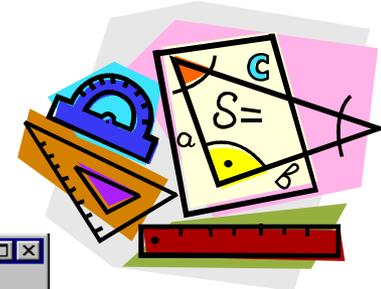


## ... Activity - Display a Domino View on an XPage

---



# Lesson: Display a Domino Document on an XPage



Outline - IBM Lotus Domino Designer

File Edit Create Table Design Navigate Tools Search Project Window Help

Applications

- Folders
  - XPages
    - ActivityDemo4-1-NewXpage
    - Customer
    - CustomerDocument
    - CustomersByRegionView
    - CustomerView
    - Demo-1-3
    - Demo-4-3-1-Form

Outline

- Root Document
  - XML Processing Instruction
  - XPage
    - xp.this.data
    - Div
      - Line break
      - Table
        - Table row
        - Table row

Customer Information

Customer:	Customer
Address:	Address
City:	City
State:	State
Zip:	Zip
Contact:	Contact
Phone:	Phone
Region:	Region
Sales Representative:	{SalesRep1}
Sales potential:	SalesPotential
Status:	Status
Type:	Type
Other products:	OtherProducts
(Error Messages)	
<input type="button" value="Submit"/> <input type="button" value="Return to View"/>	

Data source: document1

- Customer\ [Text]
- Address\ [Text]
- City\ [Text]
- State\ [Text]
- Zip\ [Text]
- Phone\ [Text]
- Contact\ [Text]
- Type\ [Text]
- SalesPotential\ [Number]
- Region\ [Text]
- SalesRep\ [Text]
- Status\ [Text]
- OtherProducts\ [Text]

Design Source

Properties Events Problems (0 items)

Online

## Procedure: Add a Document Data Source to a NEW XPage ...

1. **New XPage** dialog - check 'Add data source to the page':
  - i. Choose "Domino Document" from drop-down 'Data source' field .
  - ii. Set appropriate values for 'Application' and 'Form' fields.
  - iii. **Accept** or change data source name.
2. Click the [OK] button.

**New XPage**

Create a new XPage.

Name: MyFirstXPage

Comment:

Application: ND85XPGS.852 Workbench : tlcc\85DEV\ND85XPGS\XPG85wrk.nsf

Add data source to the page

Data source: Domino Document

Data Source: Domino Document

Application:

Current tlcc\85DEV\ND85XPGS\XPG85wrk.nsf

Other

Form: CustomerForm

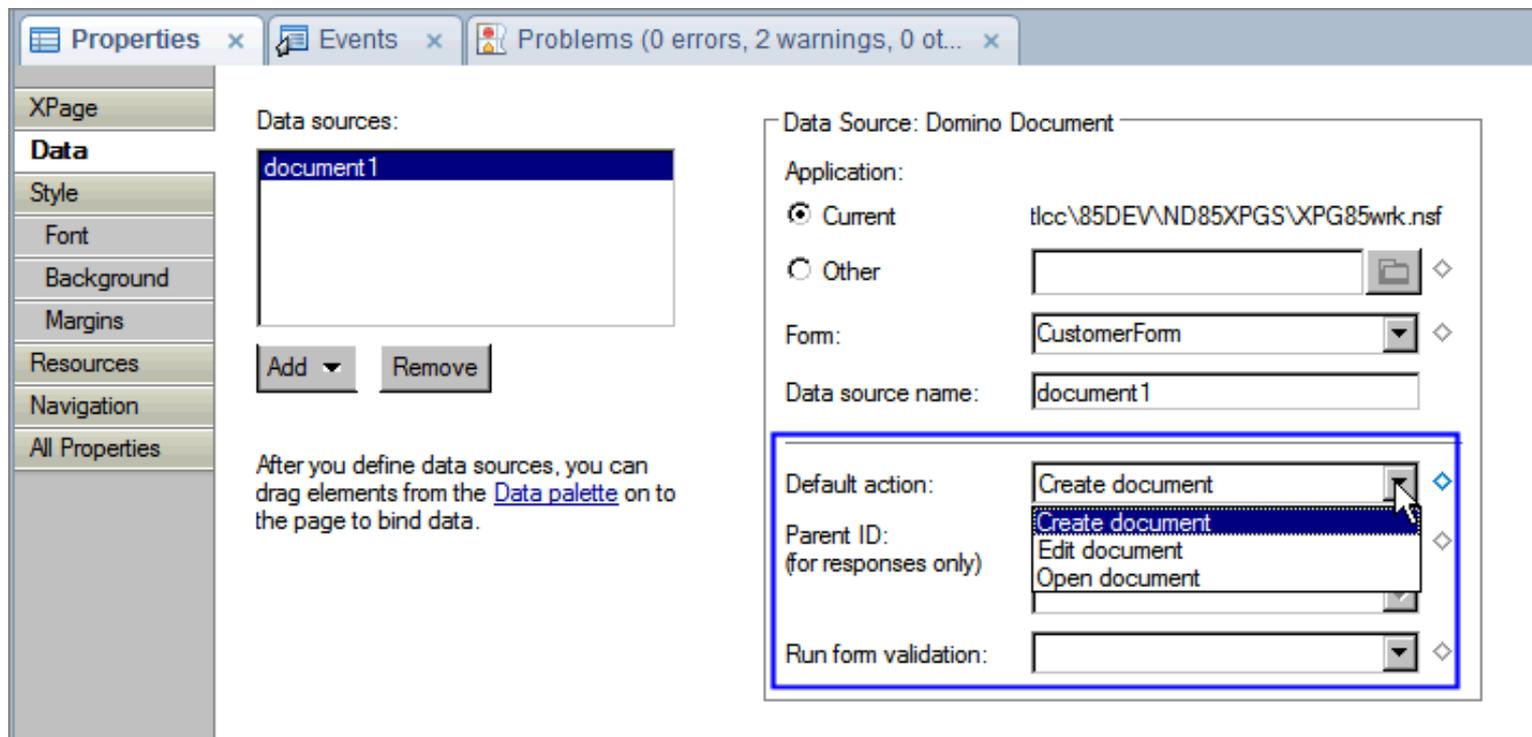
Data source name: document1

Run form validation:

OK Cancel

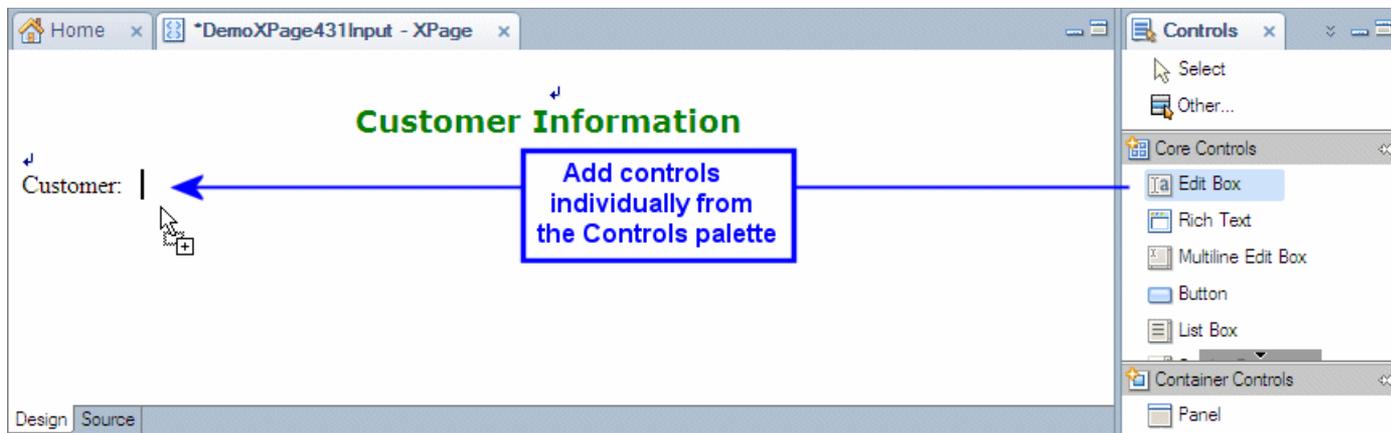
## Procedure: Add a Document Data Source to an EXISTING XPage

1. XPage Properties Data tab.
2. Click [Add] button and choose **Domino Document** from menu.
3. Select application for the document data source.
4. Select Domino form for the data source.
5. The 'Default action' is to create a new document.

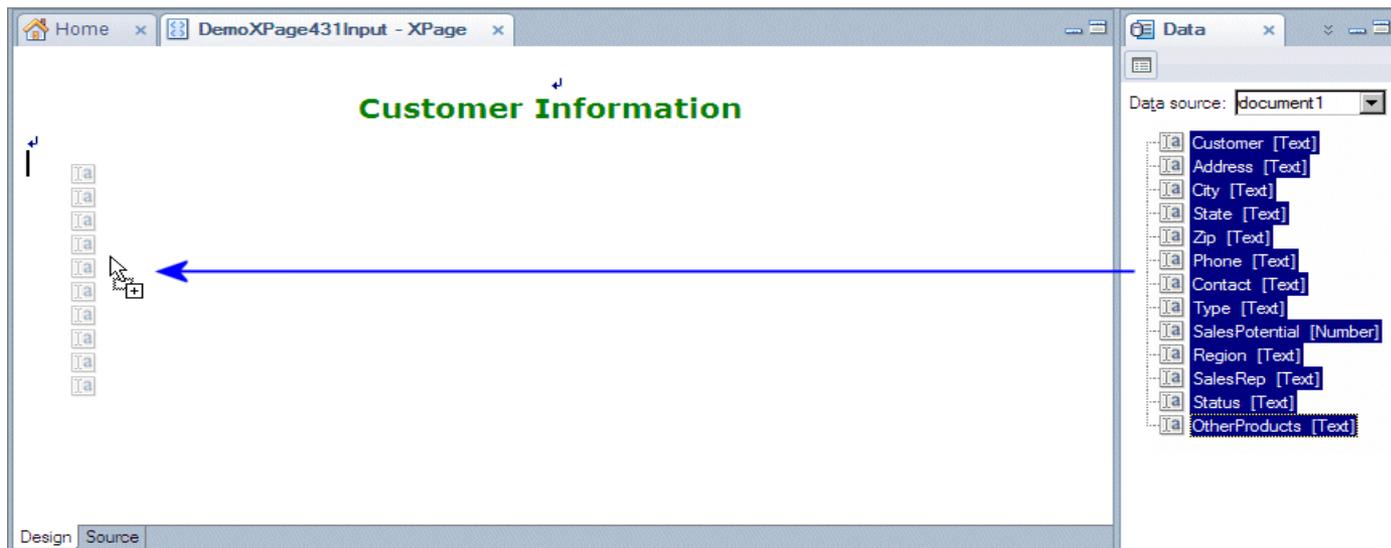


## Adding Controls for Fields in a Domino Document ...

- Approach 1: Add controls individually and then bind each control



- Approach 2: Use the Data palette to add the selected fields and automatically bind them



## Approach 1: Add Controls individually and then Bind each Control ...

... using the Controls Palette (2 step process)

The screenshot displays the IBM Domino Designer interface. At the top, there are three browser tabs: 'Home', 'DemoXPage291 - XPage', and '\*DemoXPage281 - XPage'. The main workspace shows a design view of an XPage with a large dashed box labeled 'Customer Information'. Below this, there are two input fields: 'Customer: Customer' and 'Phone: Phone'. A blue box with the number '1' is positioned between the 'Customer Information' box and the 'Phone' field, with a blue arrow pointing from the 'Edit Box' control in the 'Controls' palette to the 'Phone' field.

The 'Controls' palette on the right side of the interface is open, showing a list of controls. Under 'Core Controls', 'Edit Box' is selected. Under 'Container Controls', 'Panel', 'Repeat', 'Include Page', 'Table', 'View', and 'Data Table' are listed. Under 'Custom Controls', 'CustomerFooter', 'CustomerHeader', and 'MultiFileUpload' are listed.

At the bottom, the 'Properties' window is open, showing the configuration for the selected 'Edit Box' control. The 'Data' section is expanded, and the 'Bind data using:' section is highlighted with a blue box and the number '2'. The 'Simple data binding' radio button is selected. The 'Data source:' dropdown is set to 'document', the 'Bind to:' dropdown is set to 'Phone', and the 'Default value:' field is empty. The 'Display type:' dropdown is set to 'String'.

## Approach 2: Use the Data Palette to add Multiple Controls

The image illustrates the process of adding data controls to an XPage in IBM Domino Designer. It is divided into two main parts:

**Top Part: Data Palette and XPage Design**

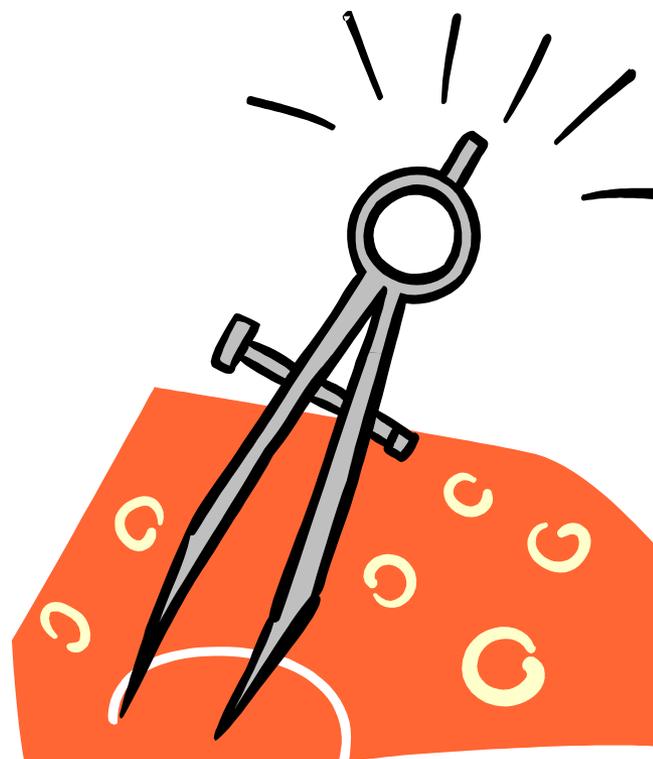
- The **Data** palette is open, showing a list of data sources from a document, including: Customer [Text], Contact [Text], Address [Text], City [Text], State [Text], Zip [Text], **Phone [Text]**, Type [Text], SalesPotential [Number], Region [Text], SalesRep [Text], Status [Text], OtherProducts [Text], and Referral [Text].
- The **Customer Information** XPage is shown in Design mode. A blue arrow labeled '1' points from the **Phone [Text]** field in the Data palette to a **Phone** control on the page.

**Bottom Part: Properties Window Configuration**

- The **Properties** window for the **Phone** control is open, showing the **Data Binding** section.
- The **Data Source** is set to **document**.
- The **Bind to** field is set to **Phone**.
- The **Display type** is set to **String**.
- The **Simple data binding** radio button is selected.

## Procedure: Use the Data Palette to add Multiple Controls

---



## Procedure: Enabling Links in a View Control to Open Documents ...

The screenshot displays the IBM Domino Designer interface. On the left, the 'XPages Intro' project tree shows the 'DemoXPage431View' selected. Below it, the 'Outline' pane shows the view structure, with 'View column: viewColumn1' highlighted. The main design area shows a table titled 'Customers By Name' with columns for 'Customer', 'City', 'Contact', and 'Phone'. The 'City' column is currently empty. Below the design area, the 'Properties' pane is open for the selected 'View Column'. The 'Display' section is active, and the 'Show values in this column as links' checkbox is checked and highlighted with a blue box. The 'Content type' is set to 'Text', and the 'Document open mode' is set to 'Read-only'.

**Customers By Name** <<First | <Previous | Page1|2|3|4|5 | Next> | Last>>

Customer	City	Contact	Phone
Customer	City	Contact	Phone

**Properties** x **Events** x **Problems (0 items)** x

**View Column**

**Data**

**Display**

**Style**

**Font**

**Background**

**Margins**

**All Properties**

Display column values

Content type: Text

Show values in this column as links

Document open mode:  Edit  Read-only

Check box

Icon

Column image:

Alternate text:

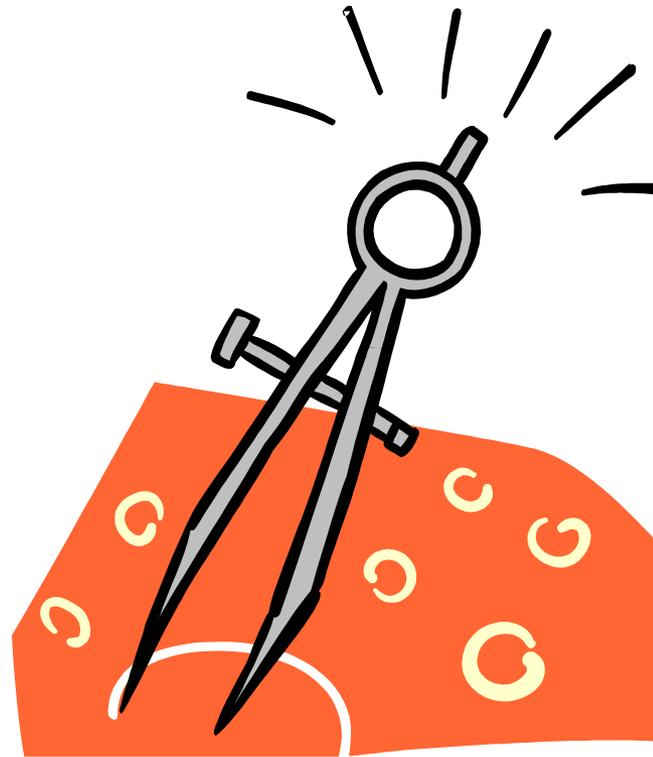
## Procedure: Enabling Links in a View Control to Open Documents

The screenshot displays the IBM Domino Designer interface with three main panes:

- Project Explorer (top-left):** Shows a tree view of the project structure. The 'DemoXPage431View' is selected under the 'XPages' folder.
- Design View (top-right):** Shows a preview of the XPage. At the top, there is a navigation bar: '<<First | <Previous | Page1|2|3|4|5 | Next> | Last>>'. Below it is a table with four columns: 'Customer', 'City', 'Contact', and 'Phone'. The first row contains the column headers, and the second row contains the word 'Customer' under the 'Customer' column.
- Properties Panel (bottom-right):** Shows the configuration for the selected 'View' control. The 'Name' is 'viewPanel1'. The 'Visible' checkbox is checked. A blue box highlights the 'At runtime, open selected document using:' dropdown menu, which is currently set to 'DemoXPage431Input'. Below this, the 'Window behavior for navigation and links (Notes client only):' dropdown is set to 'Use page default'.
- Outline (bottom-left):** Shows the hierarchical structure of the XPage, including 'xp.this.data', 'xp.this.resources', several 'Panel' controls, and the 'View: viewPanel1' which contains 'xp.this.facets' and four 'View column' controls.

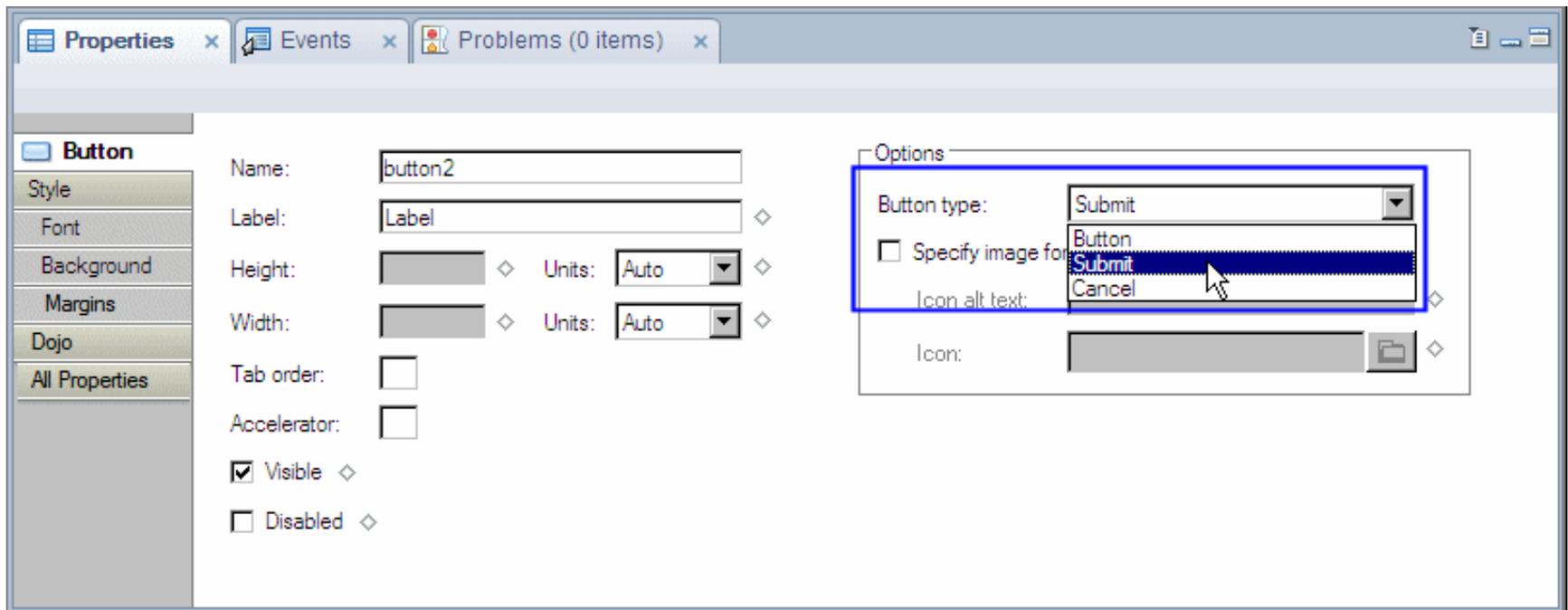
## Procedure: Create Links to Display Documents from View Control

---

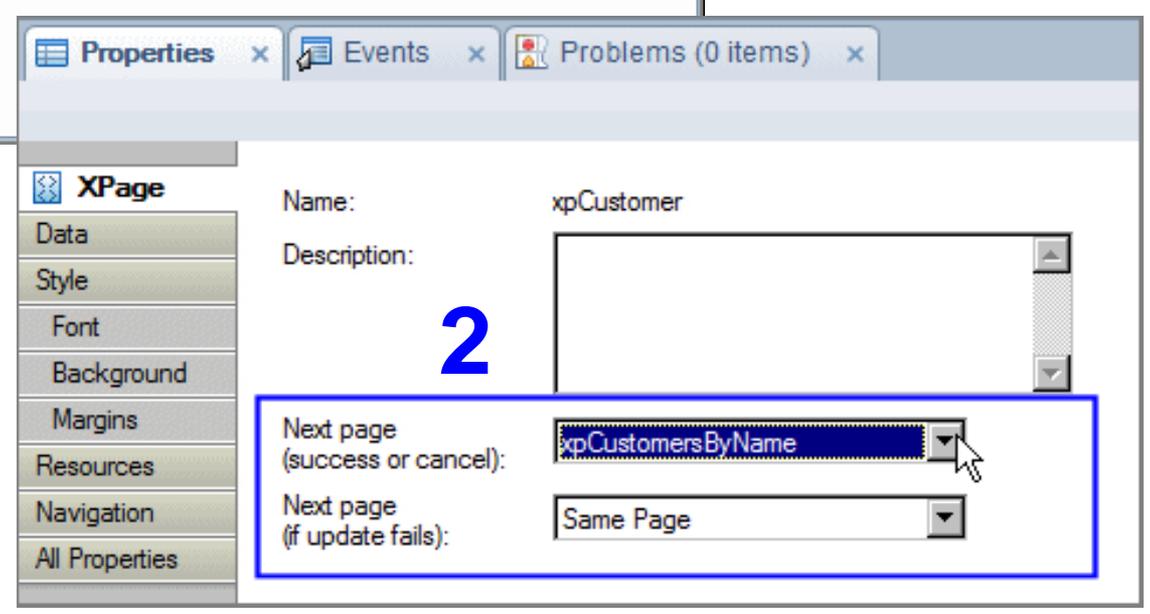
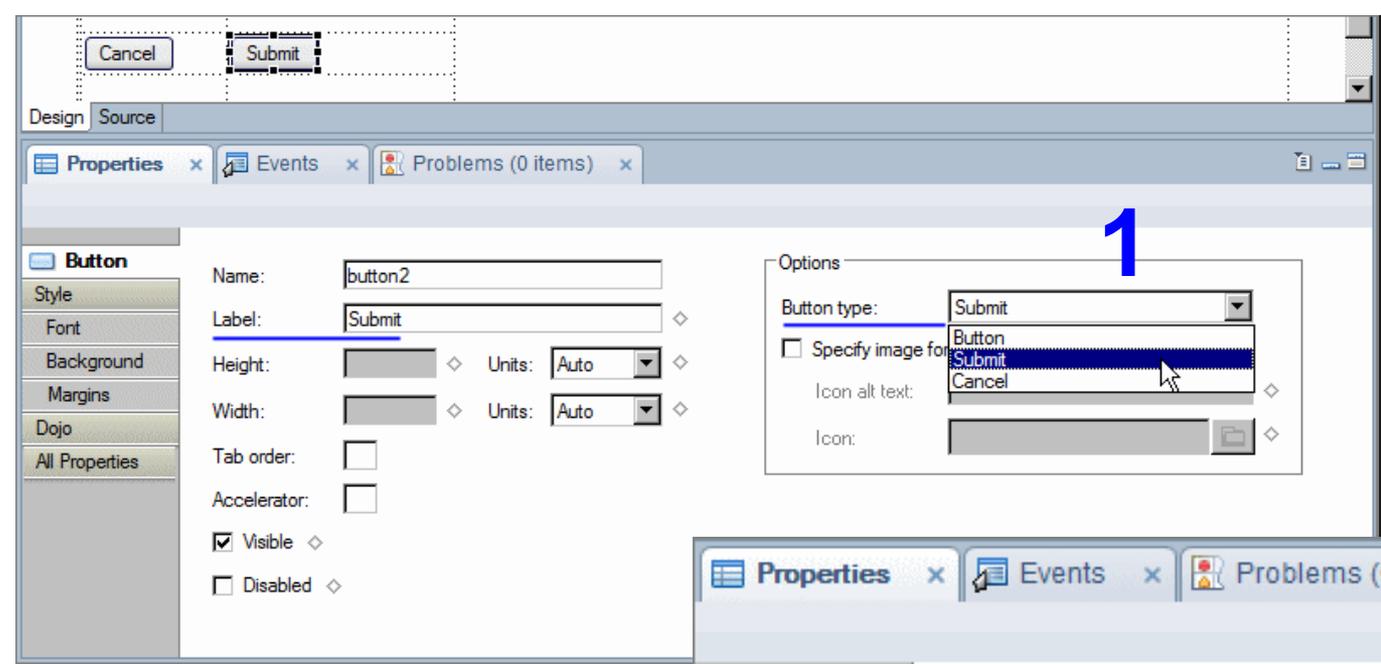


## Adding Navigation Controls to an XPage ...

- **Submit button** - validates input, submits and saves data sources
- **Cancel button** - submits XPages, no validation, no saving
- **Other buttons** - code onclick event to specify what happens



# Procedure: Create a Submit and Cancel Button



## Procedure: Adding Other Buttons to an XPage

**1**

Design Source

Properties Events Problems (0 items)

**Button**

Name: button2

Label: Return to View

Options

Button type: Button

Specify image for button

Icon alt text:

Icon:

Events

Client Server

Simple Actions Script Editor

Add Action... Add Group... Edit... Remove

Action	Arguments
Simple Actions Root	
Open Page	Page Name: /DemoXPage432Vie...

Server Options

Full Update

Partial Update

No Update

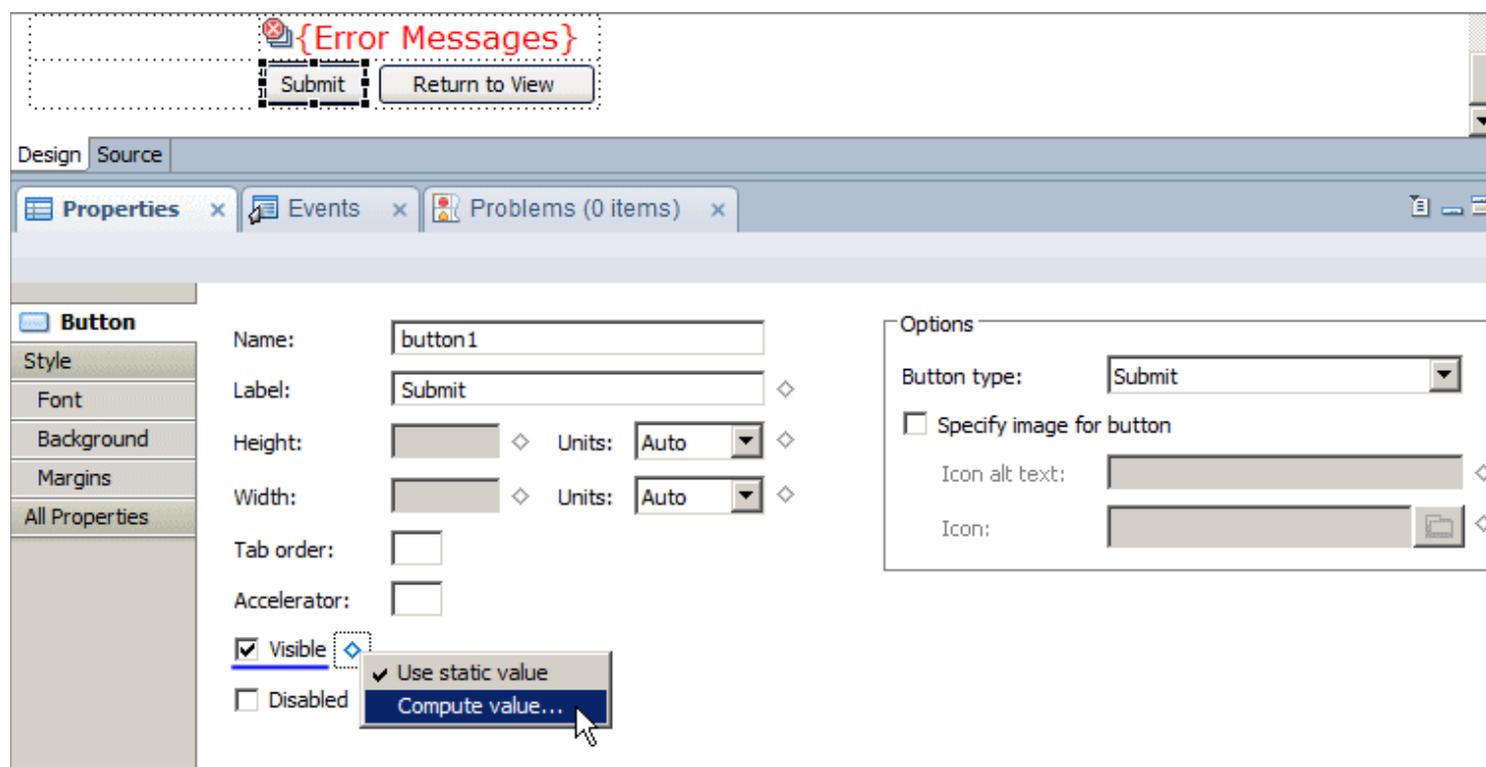
No Submission

No data validation

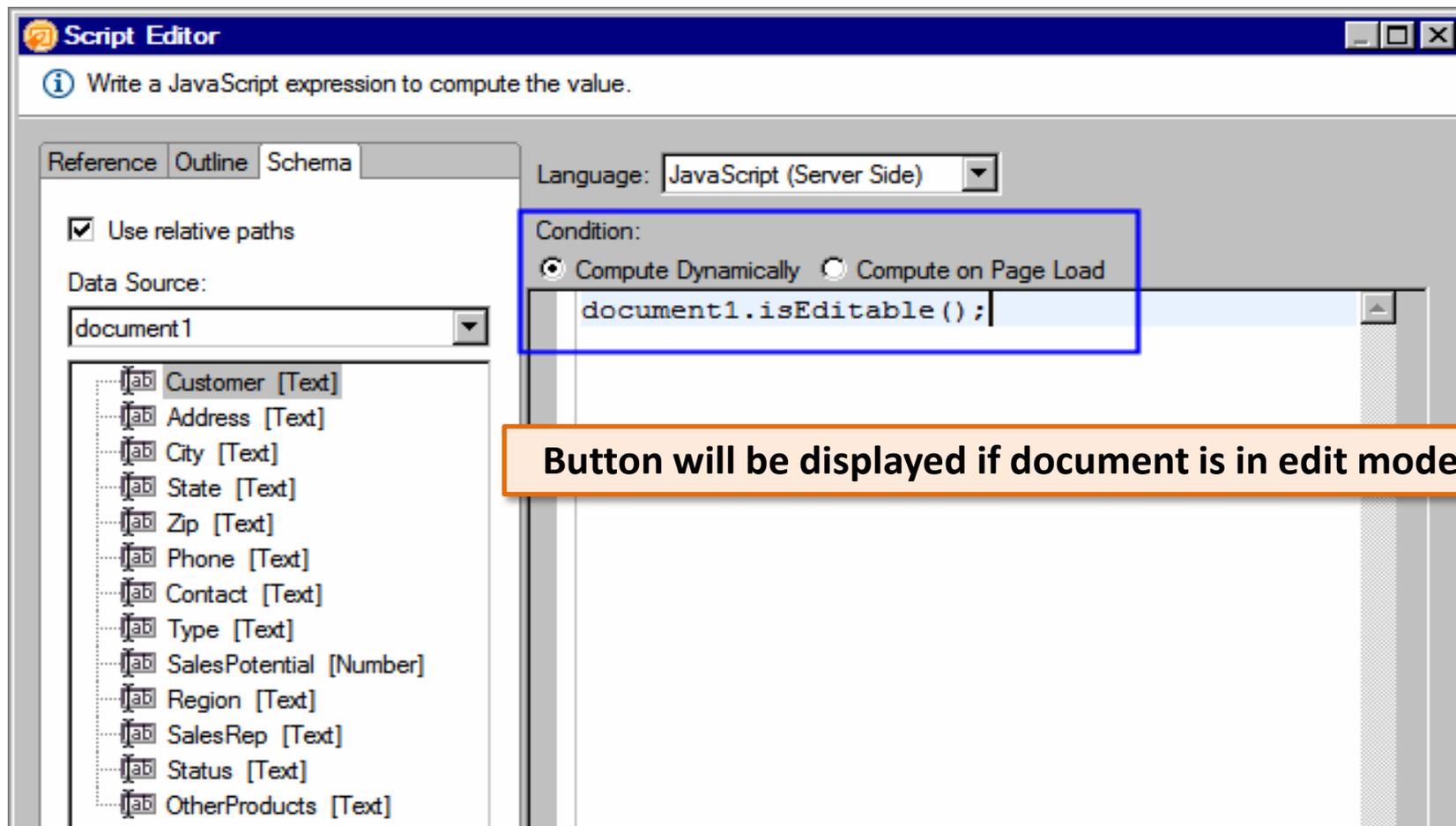
**2**

## Hiding Controls on an XPage ...

- Script to compute the value of the 'Visible' property
  - Return boolean True to show (not like the Hide When on a form!!!)



## Hiding Controls on an XPage



The screenshot shows the Script Editor window with the following configuration:

- Language: JavaScript (Server Side)
- Condition:  Compute Dynamically  Compute on Page Load
- Code: `document1.isEditable();`

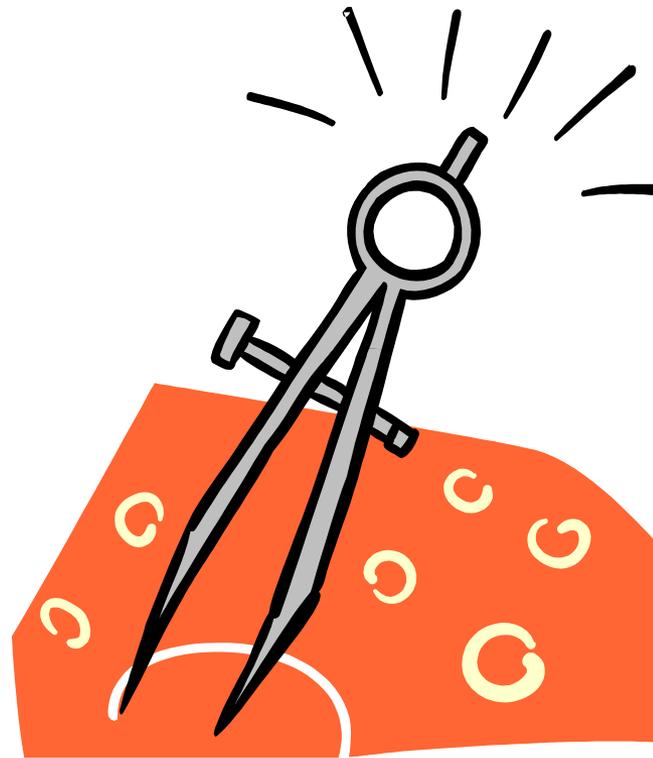
The left pane shows the Schema tab with the following data source fields:

- Customer [Text]
- Address [Text]
- City [Text]
- State [Text]
- Zip [Text]
- Phone [Text]
- Contact [Text]
- Type [Text]
- SalesPotential [Number]
- Region [Text]
- SalesRep [Text]
- Status [Text]
- OtherProducts [Text]

An orange callout box highlights the code and condition, stating: **Button will be displayed if document is in edit mode**

## Procedure: Adding Submit and Cancel Buttons

---

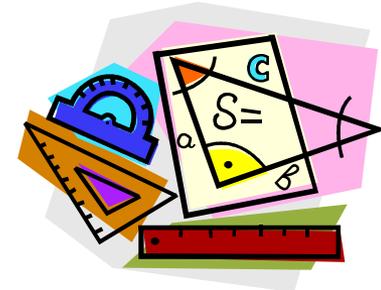


## ... Activity - Display a Domino Document on an XPage

---



# Lesson: Creating and Editing Domino Documents



Home x CustomerDocument - XPage x Controls x Data x

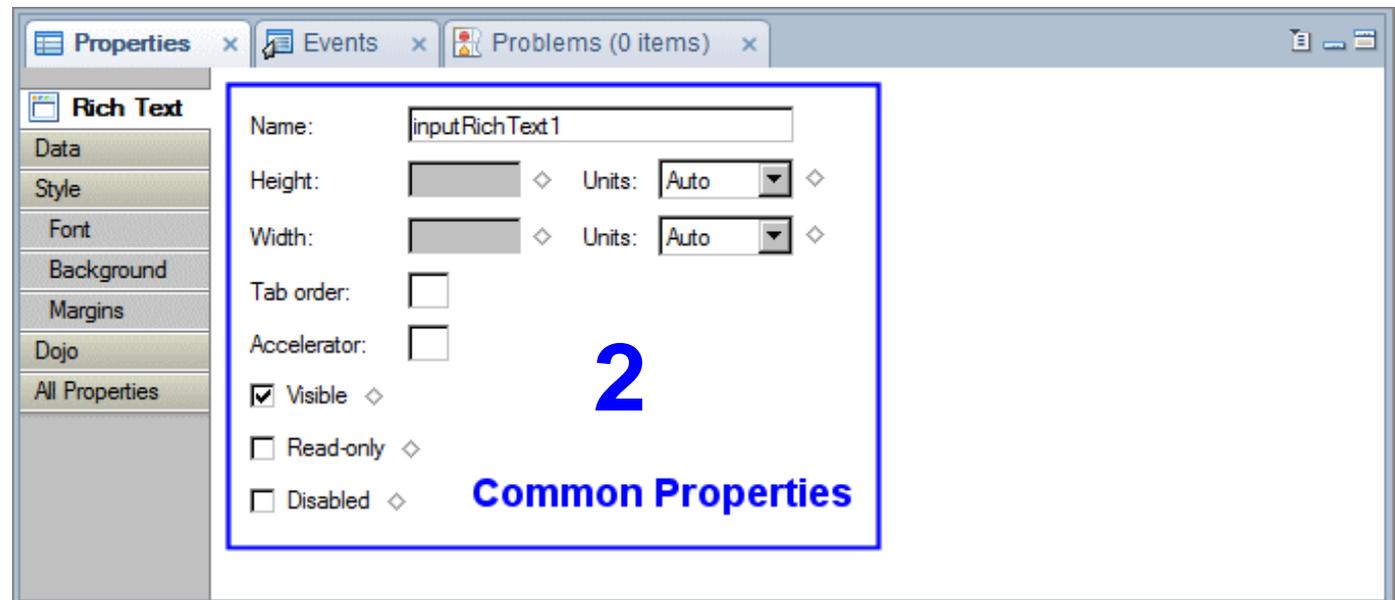
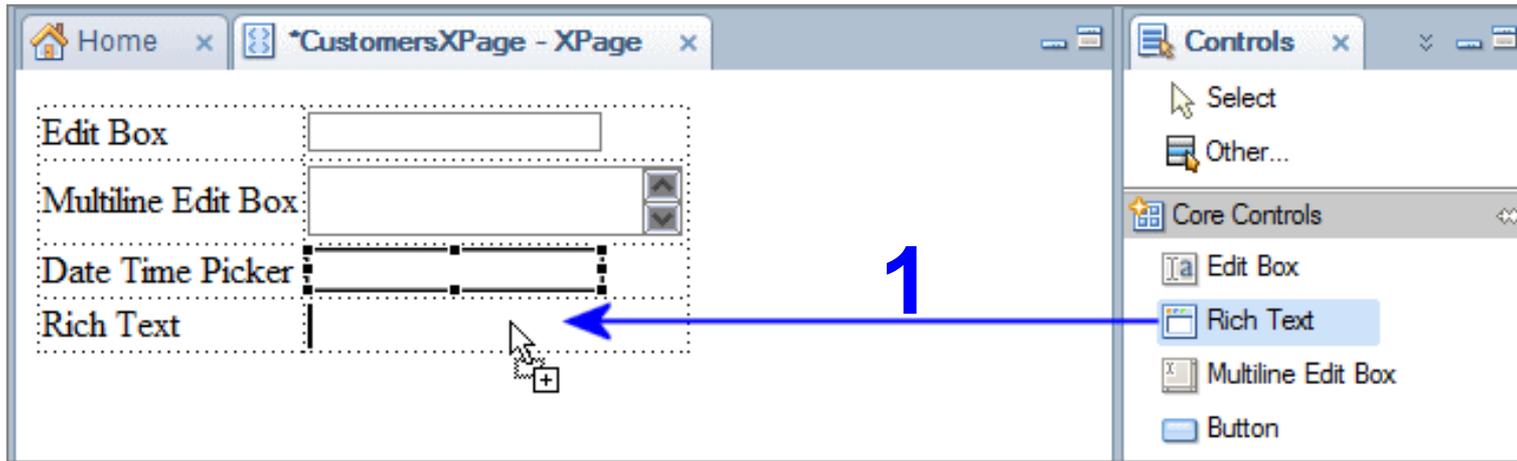
## Customer Information

Customer:	Customer	T
Address:	Address	T
City:	City	T
State:	State	T
Zip:	Zip	T
Contact:	Contact	T
Phone:	Phone	T
Region:	Region	▾
Sales Representative:	{SalesRep1}	
Sales potential:	SalesPotential	#
Status:	Status	
Type:	Type	
Other products:	OtherProducts	
	{Error Messages}	
	<input type="button" value="Submit"/> <input type="button" value="Return to View"/>	

Core Controls

- Edit Box
- Rich Text
- Multiline Edit Box
- Button
- List Box
- Combo Box
- Check Box
- Radio Button
- Check Box Group
- Radio Button Group
- Link
- Label
- Computed Field
- Image
- File Upload
- File Download
- Date Time Picker
- Display Error
- Display Errors
- Pager

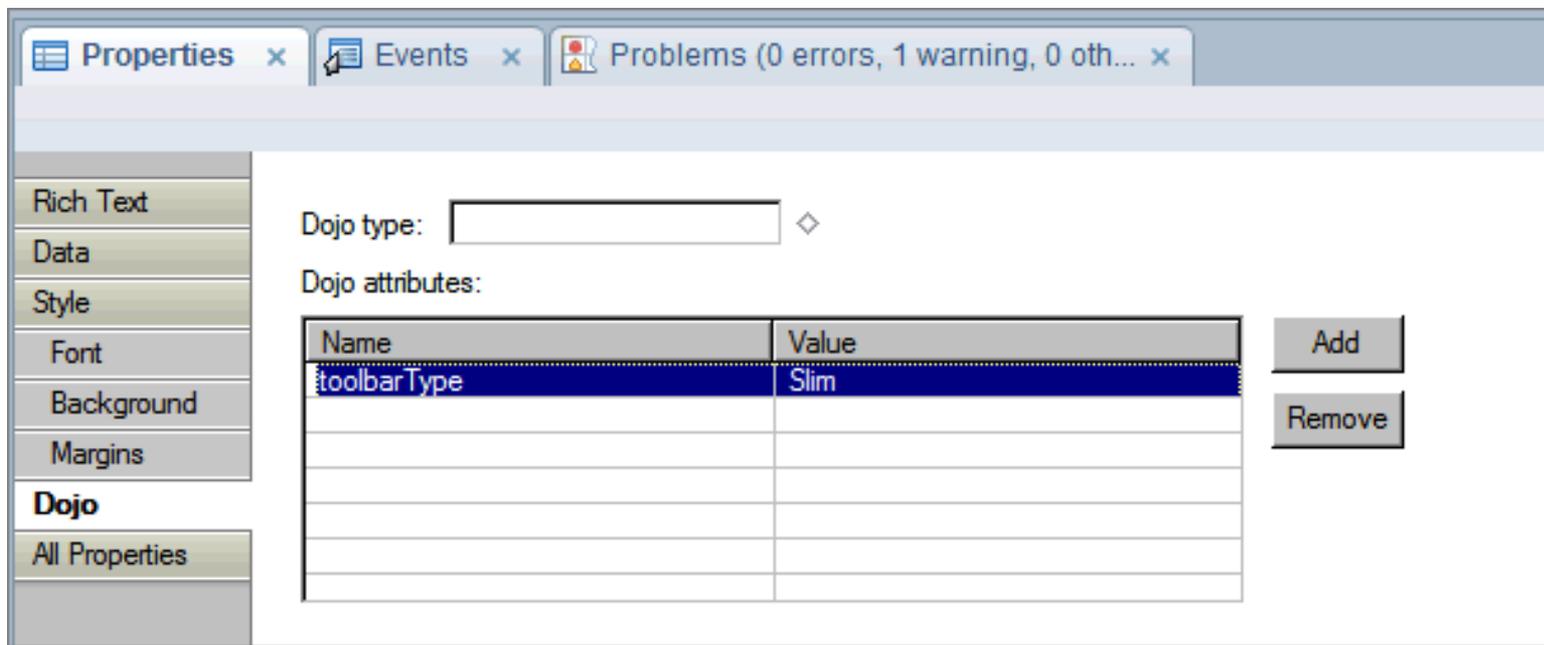
# Rich Text



## Rich Text Editor Control Type ...

Three tool bar types available for the new (in 8.5.2) CKEditor widget

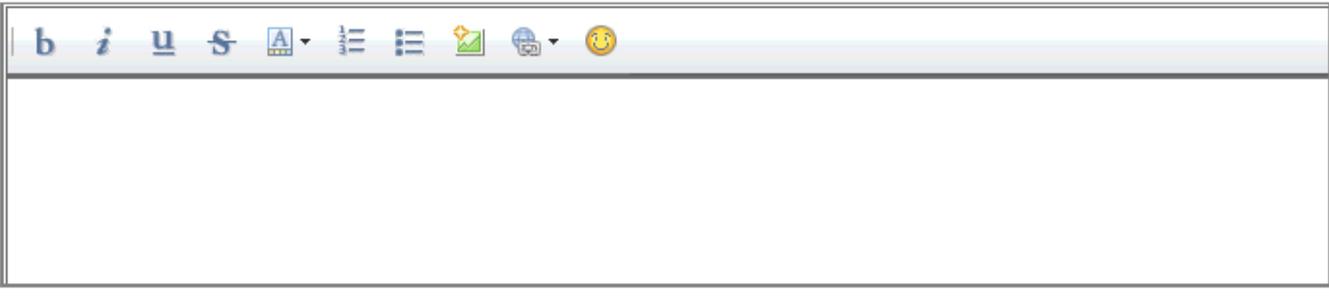
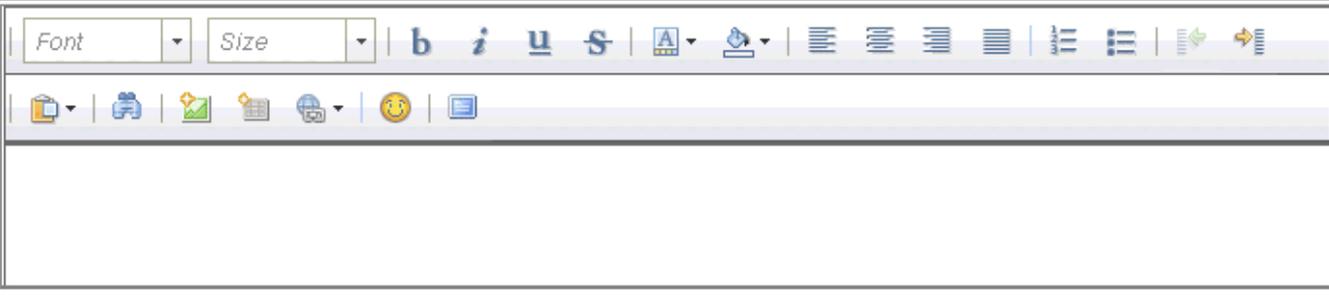
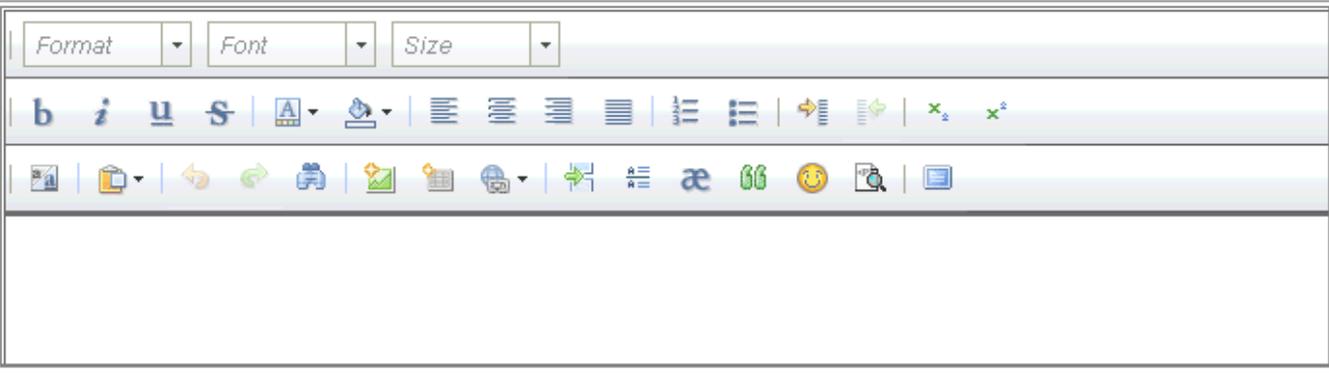
- Slim
- Medium (default if no toolbar attribute is specified)
- Large



The screenshot shows the Domino Properties dialog box with the 'Dojo' section selected. The 'Dojo type' field is empty. The 'Dojo attributes' table has one row with 'toolbarType' and 'Slim'. The 'Add' and 'Remove' buttons are visible to the right of the table.

Name	Value
toolbarType	Slim

## Rich Text Editor Control Type

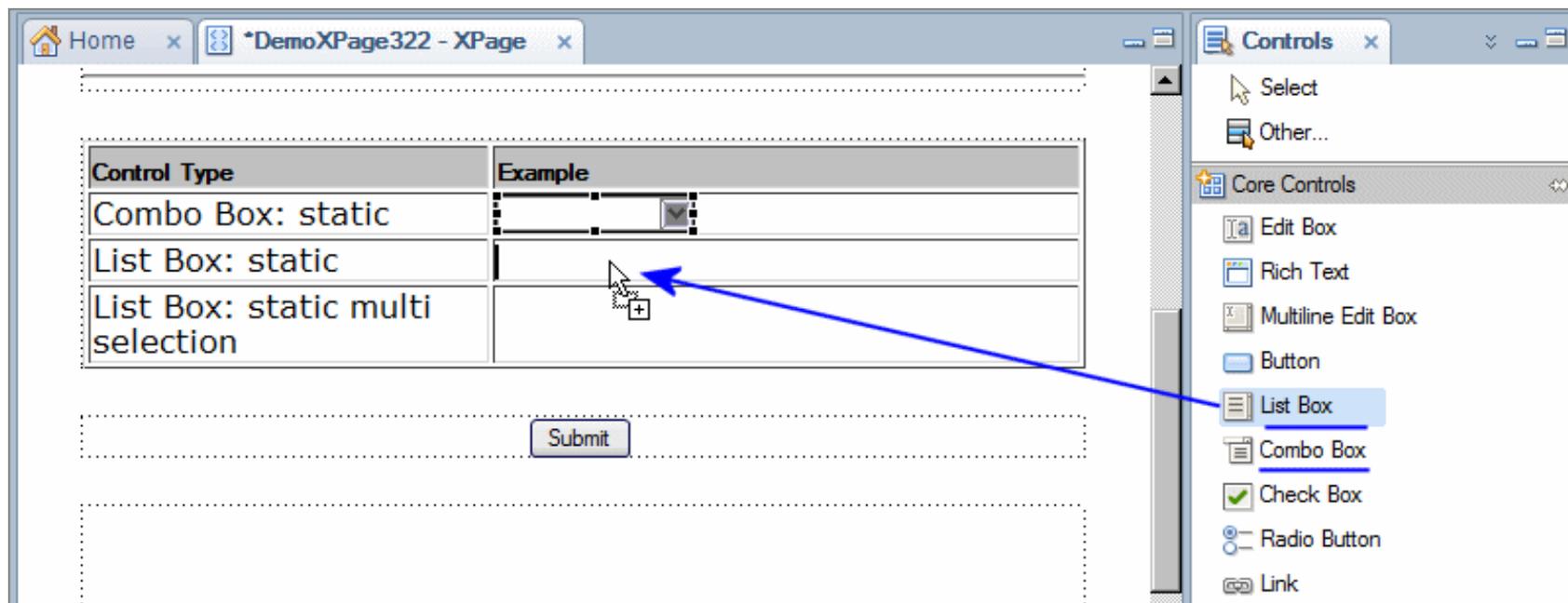
Rich Text Editor - toolbarType	Rendered in a Browser
Slim	 <p>The Slim toolbar is a single horizontal row of icons. From left to right, it includes: bold (b), italic (i), underline (u), strikethrough (ABC), font color (A), bulleted list, numbered list, link, unlink, and smiley face.</p>
Medium	 <p>The Medium toolbar consists of two rows of icons. The top row includes: Font dropdown, Size dropdown, bold (b), italic (i), underline (u), strikethrough (ABC), font color (A), background color (brush), bulleted list, numbered list, decrease indent, increase indent, bulleted list, numbered list, undo, and redo. The bottom row includes: insert image, insert link, insert table, insert table, link, unlink, smiley face, and insert video.</p>
Large	 <p>The Large toolbar consists of three rows of icons. The top row includes: Format dropdown, Font dropdown, and Size dropdown. The middle row includes: bold (b), italic (i), underline (u), strikethrough (ABC), font color (A), background color (brush), bulleted list, numbered list, decrease indent, increase indent, bulleted list, numbered list, undo, redo, link, unlink, and insert video. The bottom row includes: insert image, insert link, undo, redo, link, unlink, insert table, insert table, link, unlink, smiley face, insert video, and insert video.</p>

# Demo: Edit Controls

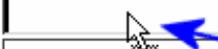
---



## Creating List Box and Combo Box Controls



The screenshot displays the Domino Designer interface. The main workspace shows a table with control types and examples, a 'Submit' button, and a 'Controls' palette on the right. A blue arrow points from the 'List Box' control in the palette to the 'List Box: static' example in the table.

Control Type	Example
Combo Box: static	
List Box: static	
List Box: static multi selection	

Submit

**Controls**

- Select
- Other...
- Core Controls
  - Edit Box
  - Rich Text
  - Multiline Edit Box
  - Button
  - List Box
  - Combo Box
  - Check Box
  - Radio Button
  - Link

## List Box and Combo Box Properties

The screenshot shows the Domino Properties window for a Combo Box control. The window title is "Properties" and it has tabs for "Events" and "Problems (1 error, 0 warnings, 0 ot...)". The left sidebar shows a tree view with "Combo Box" selected, and sub-items: "Data", "Values", "Style", "Font", "Background", "Margins", "Dojo", and "All Properties". The main area displays the following properties:

- Name:
- Height:  Units:
- Width:  Units:
- Tab order:
- Accelerator:
- Visible
- Read-only
- Disabled

The text "Common Properties" is written in blue in the bottom right of the main area.

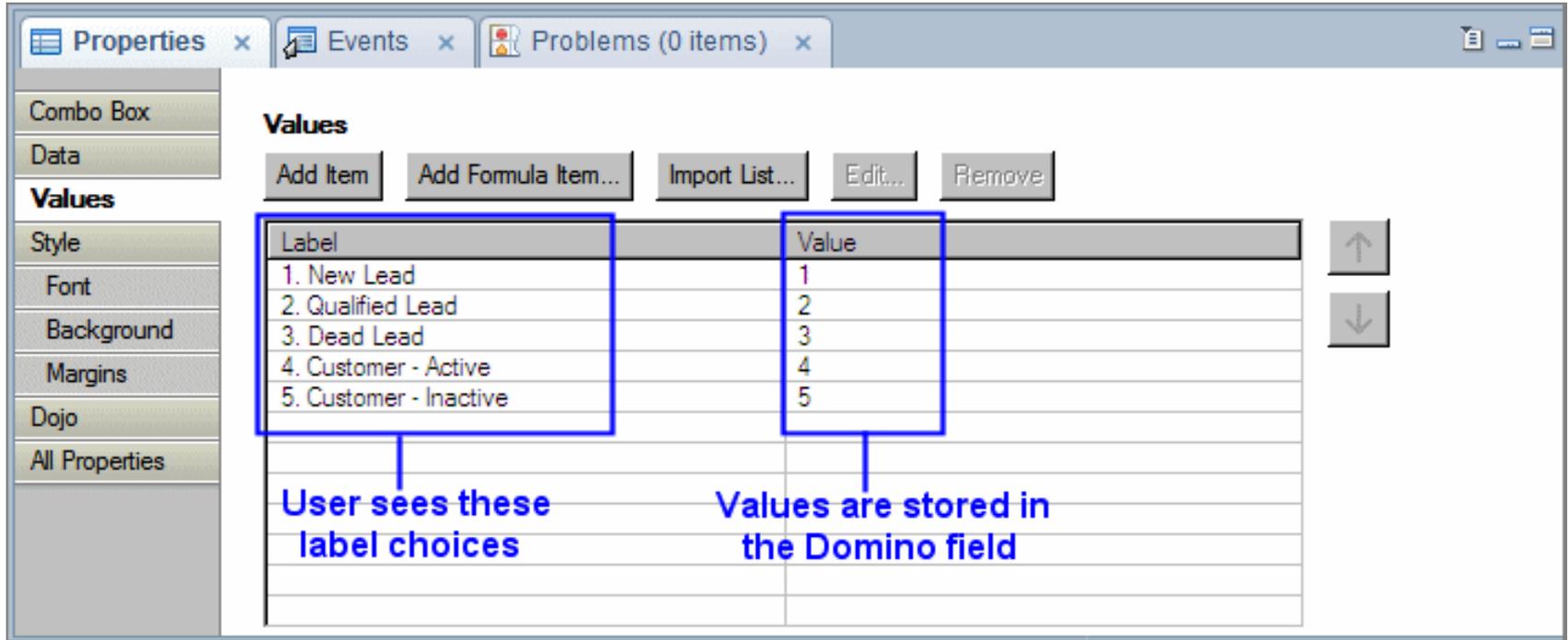
The screenshot shows the Domino Properties window for a List Box control. The window title is "Properties" and it has tabs for "Events" and "Problems (1 error, 0 warnings, 0 ot...)". The left sidebar shows a tree view with "List Box" selected, and sub-items: "Data", "Values", "Style", "Font", "Background", "Margins", "Dojo", and "All Properties". The main area displays the following properties:

- Name:
- Height:  Units:
- Width:  Units:
- Tab order:
- Accelerator:
- Visible
- Read-only
- Disabled

An "Options" section is highlighted with a blue box, containing the property:

- Allow multiple selections

## Combo Box and List Box Controls



The screenshot shows the Domino Properties window for a Combo Box control. The left sidebar contains the following categories: Combo Box, Data, Values, Style, Font, Background, Margins, Dojo, and All Properties. The main area is titled "Values" and contains a table with two columns: "Label" and "Value". The table lists five items:

Label	Value
1. New Lead	1
2. Qualified Lead	2
3. Dead Lead	3
4. Customer - Active	4
5. Customer - Inactive	5

Annotations in blue text with arrows point to the "Label" and "Value" columns:

- User sees these label choices** (points to the Label column)
- Values are stored in the Domino field** (points to the Value column)

Buttons above the table include: Add Item, Add Formula Item..., Import List..., Edit..., and Remove. Up and down arrow buttons are on the right side of the table.



## Tip: Copy Domino Form Field Choices for the List Import

**Properties for the Field in a Domino form**

The screenshot shows the 'Field' properties dialog box. The 'Choices' section is highlighted with a blue box. It contains a text area with the following text: '1. New Lead | 1', '2. Qualified Lead | 2', '3. Dead Lead | 3', '4. Customer - Active | 4', and '5. Customer - Inactive | 5'. A blue arrow points from this text area to the 'Import List' dialog box on the right.

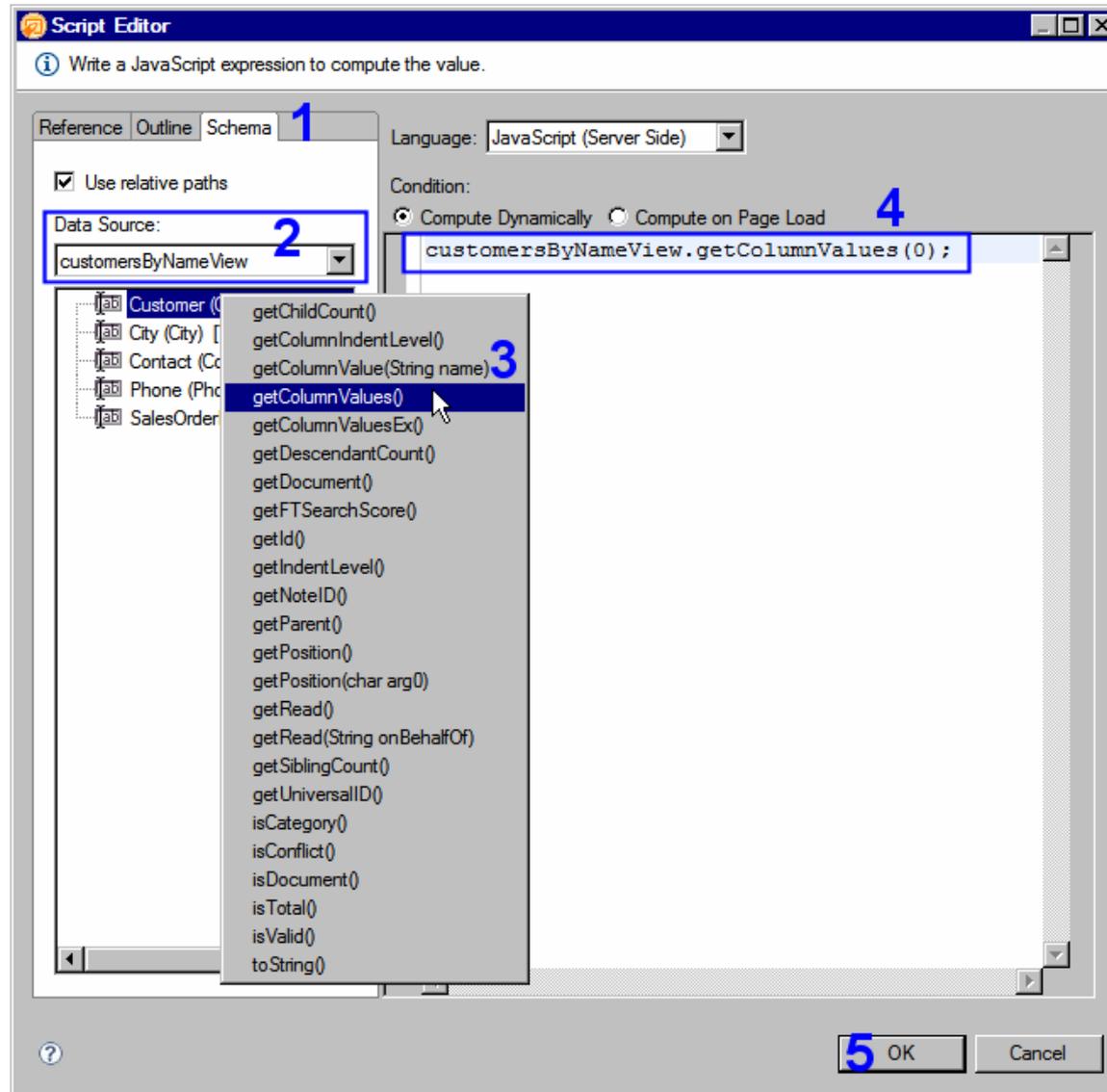
**Import List Dialog from the Combo Box Control**

The screenshot shows the 'Import List' dialog box. It has a text area containing the same list as the previous dialog: '1. New Lead|1', '2. Qualified Lead|2', '3. Dead Lead|3', '4. Customer - Active|4', and '5. Customer - Inactive|5'. A blue arrow points from the text area to the 'Separator Character' section. In the 'Separator Character' section, the 'Enter a custom separator' option is selected, and the text field contains a vertical bar '|'. A blue arrow points from this text field to the text 'Add the vertical bar as the custom separator.'

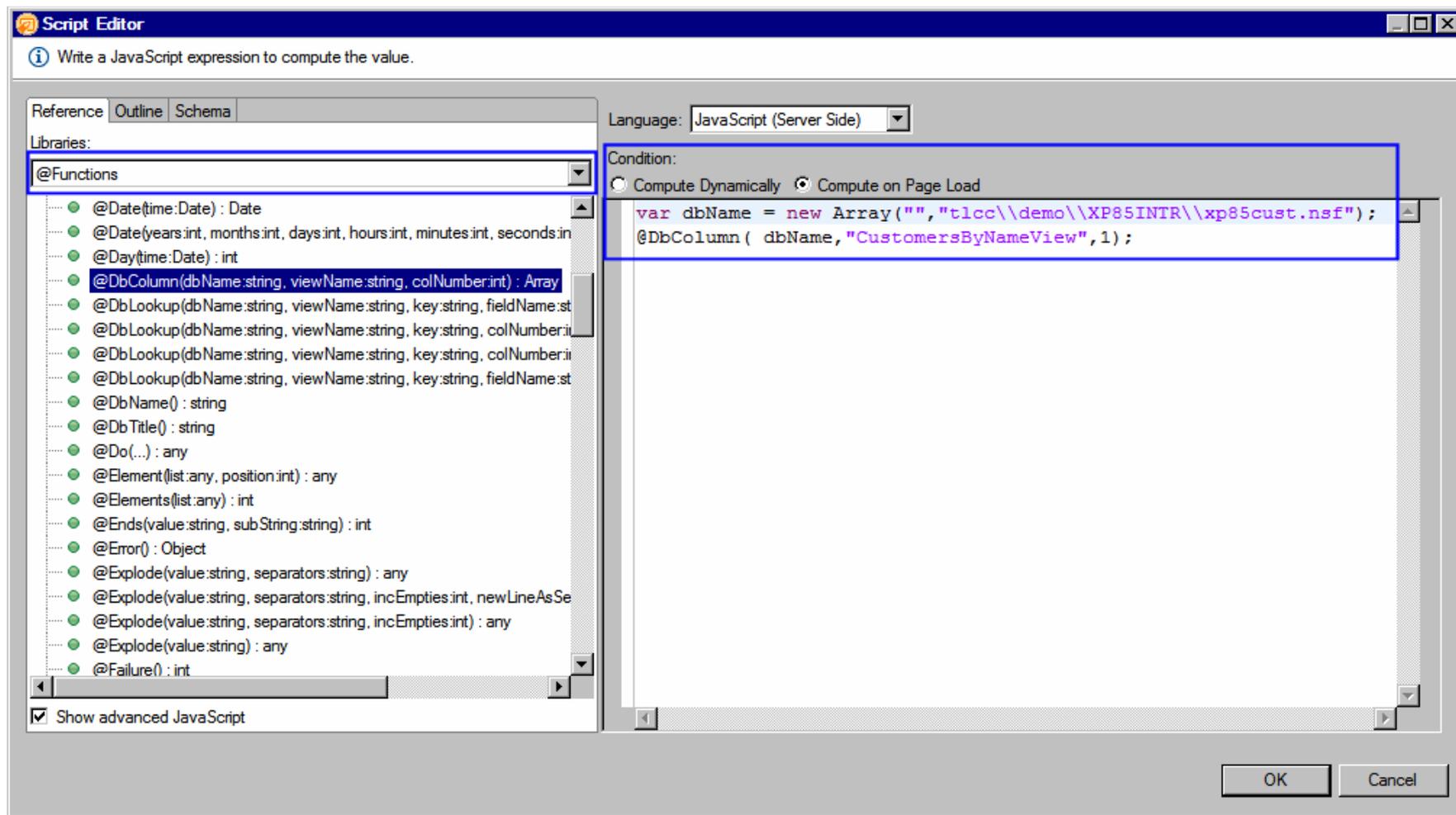
# Computing Values for Combo Box and List Box Controls ...

The screenshot shows the Domino Designer interface. On the left, the 'Properties' pane is open to the 'Values' tab. A blue box highlights the 'Add Formula Item...' button, with a large blue number '1' above it. Below this, a table with a 'Label' column is visible. On the right, the 'Script Editor' is open, showing a list of JavaScript functions under the '@Functions' category. A blue number '2' is placed next to the '@DbColumn(dbName:string, viewName:string)' function, which is highlighted by the mouse. The script editor also shows the 'Language' set to 'JavaScript (Server Side)' and the 'Condition' set to 'Compute Dynamically'.

## Return the List of Values using the Schema Tab ...



## Return the List of Values using @Functions





## Note: Considerations when using @Functions in XPages

---

XPage @Functions are JavaScript... so JavaScript rules and syntax apply:

- Use commas for parameters, not semi-colons
- Case-sensitive (@dbcolumN will not work)
- Database parameter: use empty string for local db (or @DbName)
  - or, use a two element array with server and db path

```
1 var dbName = new Array("", "tlcc\\demo\\XP85INTR\\xp85cust.nsf");  
2 @DbColumn( dbName, "CustomersByNameView", 1);
```

## Radio Button Group Control

---

- Horizontal or vertical layout.

<b>Region:</b>	<input type="radio"/> North <input type="radio"/> South <input checked="" type="radio"/> East <input type="radio"/> West
<b>Type:</b>	<input checked="" type="radio"/> Retail Golf Store <input type="radio"/> Retail Sports Store <input type="radio"/> Pro Shop

## Check Box Group Control

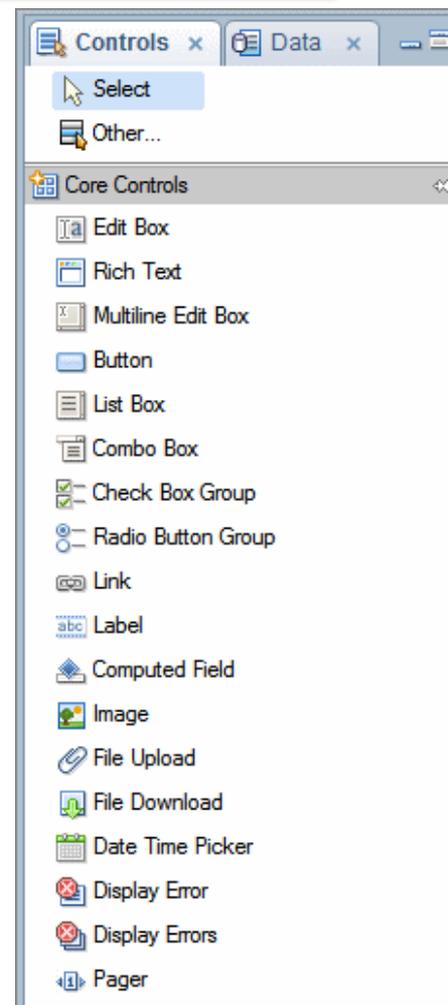
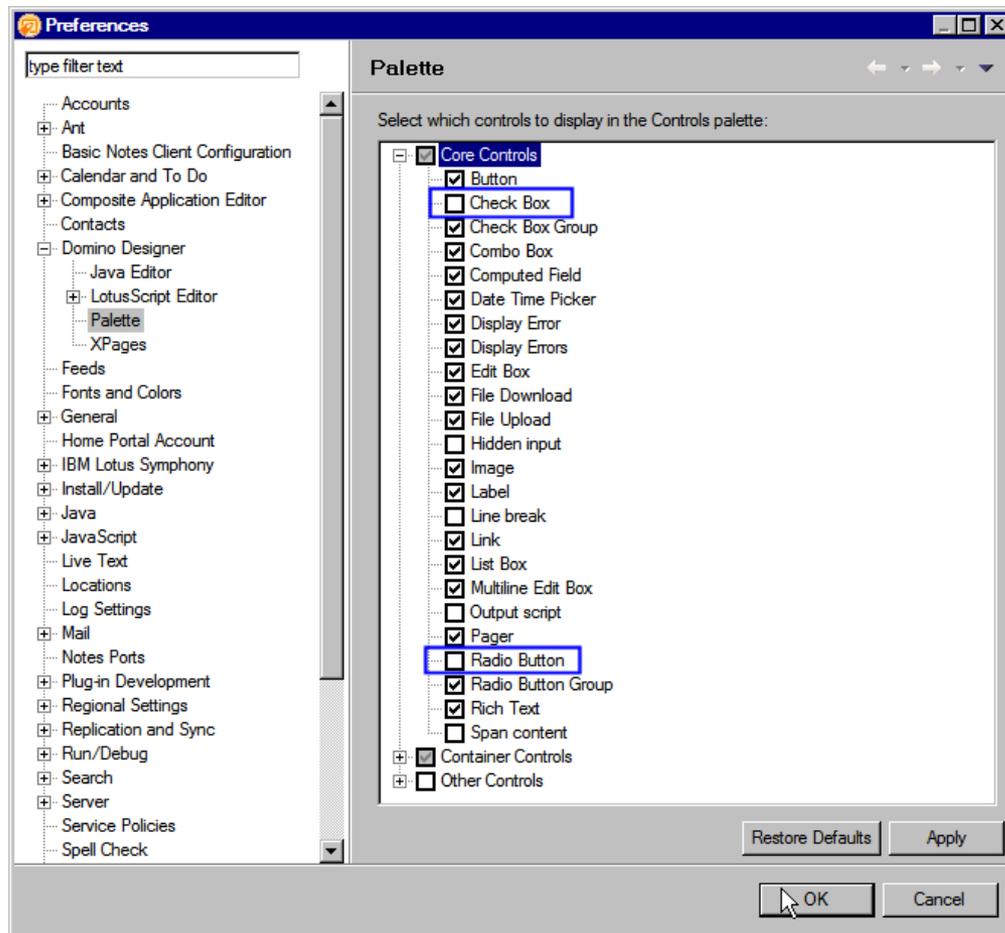
---

- Each box can be individually selected or cleared
- Multiple or single choice

<b>Other Products:</b>	<input checked="" type="checkbox"/> Golf Balls <input type="checkbox"/> Golf Bags <input checked="" type="checkbox"/> Tees <input checked="" type="checkbox"/> Gloves
<b>Agree to license agreement?</b>	<input checked="" type="checkbox"/> Yes



## Tip: Remove Legacy Check Box and Radio Button Controls from Controls Palette



## Demo: Computing Values for Combo Box and List Box Controls

---

## Demo : Using Check Box and Radio Button Group Controls

---



## ... Activity - Editing Domino Documents on an XPage

---

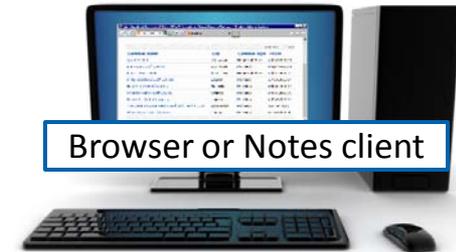


## Demonstrations of Advanced Techniques using XPages

---



# XPages Architecture



--- XPages Presentation Layer ---

HTTP Request and Response

XPage Design Element

- XPages, Custom Controls
- CSS, Themes

Compiled in  
Domino Designer

XSP Engine  
(Domino Server)

Data Sources

Domino

Domino Documents

Domino Views

# XPages Architecture



-- XPages Presentation Layer

HTTP Request and Response

## XPage Design Element

- XPages, Custom Controls
- CSS, Themes

Compiled in  
Domino Designer

XSP Engine  
(Domino Server)

Data Sources

Domino

Domino Documents

Domino Views

# XPages Architecture



-- XPages Presentation Layer

HTTP Request and Response

## XPage Design Element

- XPages, Custom Controls
- CSS, Themes

Compiled in  
Domino Designer

XSP Engine  
(Domino Server)

Data Sources

Domino

Domino Documents

Domino Views

JDBC

IBM DB2

Oracle

MS SQL

# Skills Needed for XPages Development

---

## ✓ Notes and Domino Application Development

- Domino Object Model (LotusScript or Java)

## ✓ JavaScript

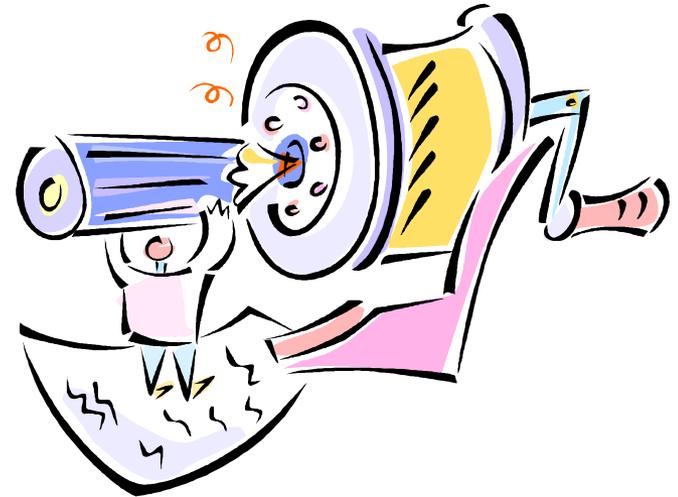
## ✓ Web Development

- HTML and CSS

## ✓ XPages skills

## ❖ You don't need to be a Java Guru!

- Create great XPages apps with no Java code ... but ...
- There is clearly a place for Java in XPages
  - ✓ Java Libraries
  - ✓ Java beans



# The XPages Developer Progression

---

## 1 - Drag and Dropper:

- Simple Actions
- Some JavaScript
- Convert existing Notes apps to XPages

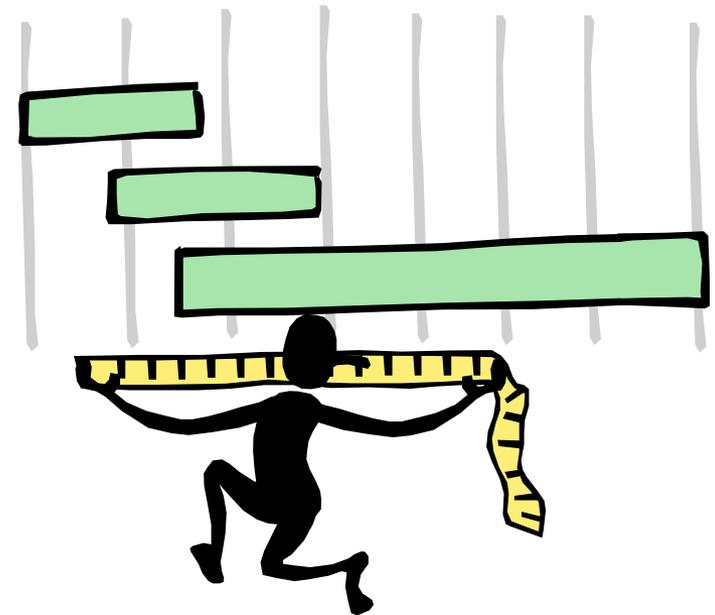
## 2 - SSJS Coder:

- Rewrite LotusScript business logic in Server-side JavaScript
- Use Source Pane more
- Use CSS to style XPage
- New apps

## 3 - The Java Guru:

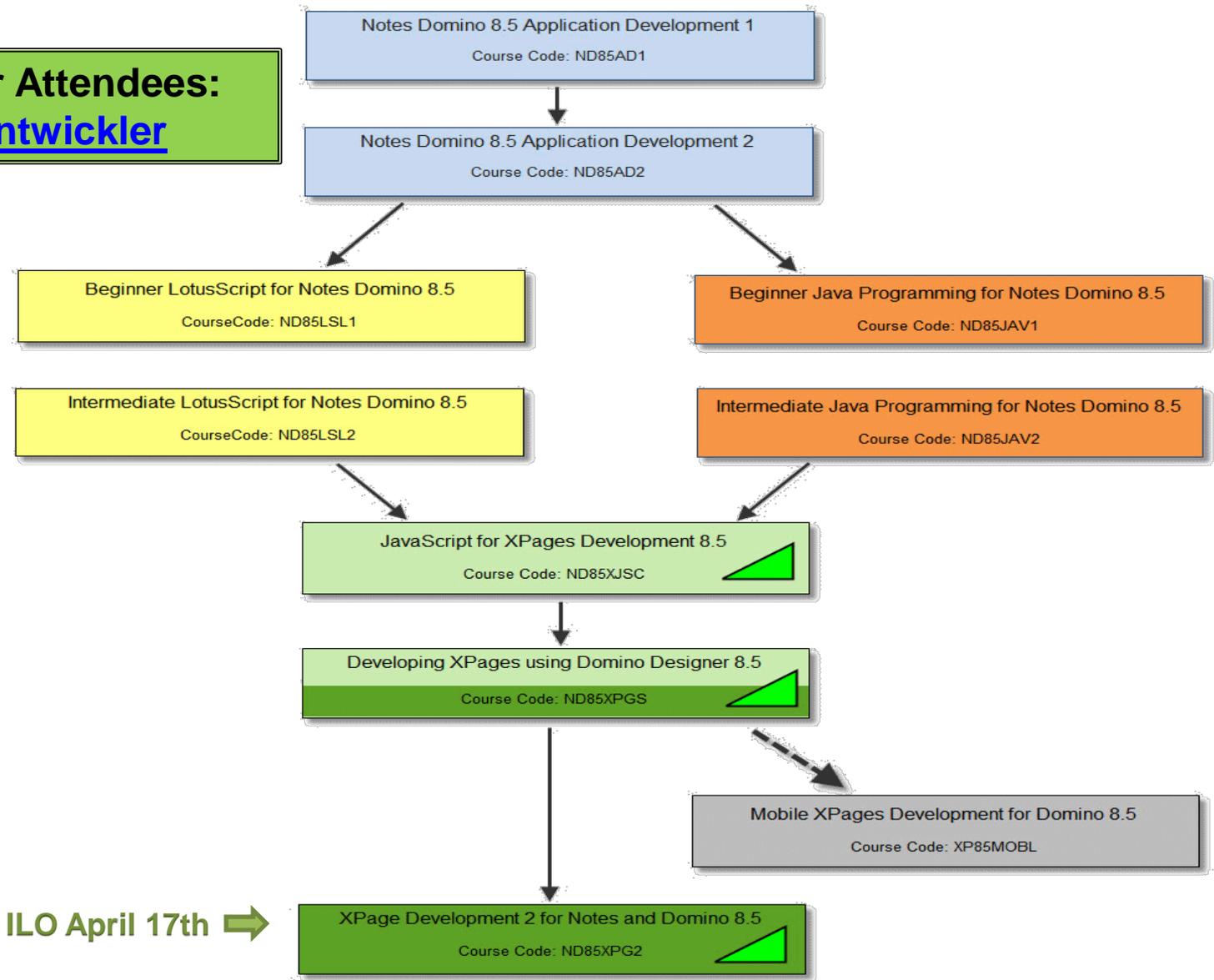
- Java - beans
- MVC architecture – separate code vs. presentation
- Pull in data from other sources
  - Relational databases
  - Web Services

❖ Not everyone will get to The Java Guru level – and that is OK!!!



# GOAL: Develop XPages Applications

**Special Offer for Attendees:**  
[www.tlcc.com/entwickler](http://www.tlcc.com/entwickler)



ILO April 17th →

# Drawing for a Course!!!



## Q and A

---

