



OpenSpan Elective Training

Incorporating HTML Table Data in OpenSpan Projects


- CHAPTER 1: Basic HTML Table Design
- CHAPTER 2: Advanced HTML Table Design

CONTENTS

CHAPTER 1: Basic HTML Table Design	5
Interrogating HTML Tables.....	5
Initiating the Query Function on a Data Cell Property.....	15
Query Function Example	16
Chapter 2: Advanced Table Design.....	17
Using Advanced Table Section Match Capabilities of the HTML Table Designer	17
Refresh Matching	24
Summary	26

CONVENTIONS

You can save time using this training guide by understanding how screen elements, input data, and definitions are shown.

Convention	Meaning
Black bold characters	Names of program elements that require emphasis, such as command buttons, menus, and dialog boxes, are shown in black bold text.
Blue Bold Characters	Text that you are supposed to type or data selections, such as from drop-lists, appear in blue boldface characters.
<u>Remember</u>	<u>Definitions of terms and important concepts that bear remembering.</u>
	Next to the Tip icon, you can find best practices and shortcuts to use OpenSpan Studio more effectively.

OVERVIEW

The OpenSpan Studio HTML Table Designer enables you to incorporate HTML Table Data into a format that can be used by OpenSpan Studio solutions. The table designer allows you to assign the HTML Table cells as either Match Cells or Data Cells. Match Cells make up match rules, which allow OpenSpan Studio to identify specific areas of the table and how they correspond to data cells. The HTML Table Designer allows you to use regex, as well as specific patterns in the creation of match rules related to these match cells.

Prerequisites

This training module assumes that you have successfully completed the **OpenSpan Studio Core** training modules. This training module also assumes you to be familiar with general HTML Table concepts.

The course exercises require your system to be setup with the following:

- OpenSpan Studio 4.5

Solutions Used in this Training Module

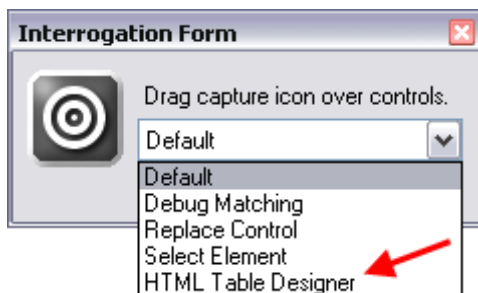
Practice files you will use while working through the step-by-step exercises presented in this training guide can be [downloaded here](#). Finished solutions are available from the same download location as a reference for checking your work after completing the exercises on your own.

CHAPTER 1: Basic HTML Table Design

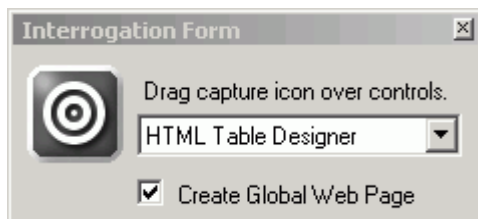
Interrogating HTML Tables

The HTML Table design functions allow you to interrogate an HTML table in order to integrate it into an OpenSpan Solution. Perform the following steps:

1. Create a new blank solution and add a Web application project item.
2. Set the **StartPage** property to the following URL and then click on the **Start Interrogation** button. URL: http://training.openspan.com/product_list1.html
3. Select **HTML Table Designer** from the drop down menu of the Interrogation Form.

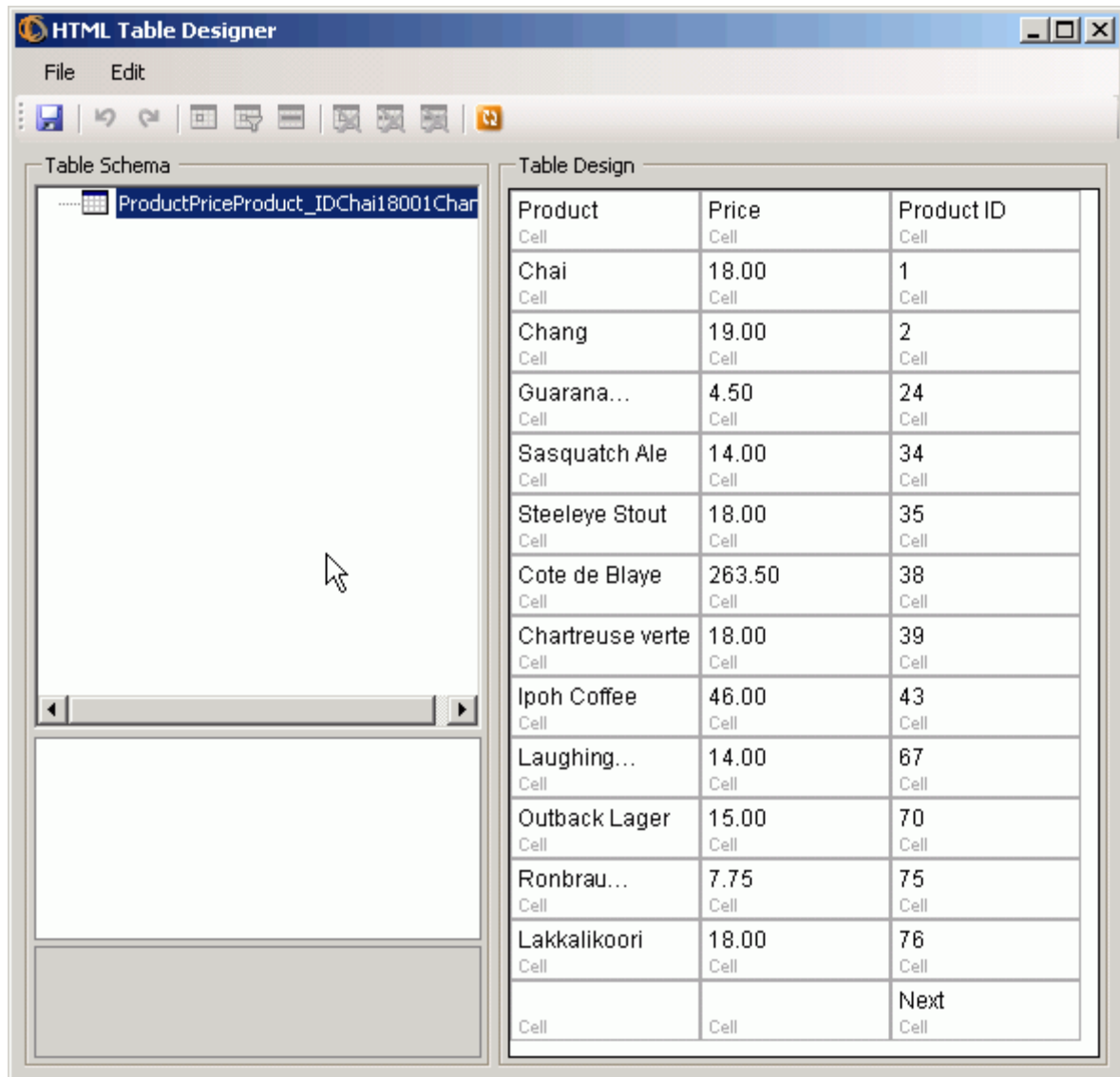


4. Select the **Create Global Web Page** option:



Chapter 1: Basic HTML Table Design

5. Drag and drop the interrogation icon onto the table. The HTML Table Designer displays:



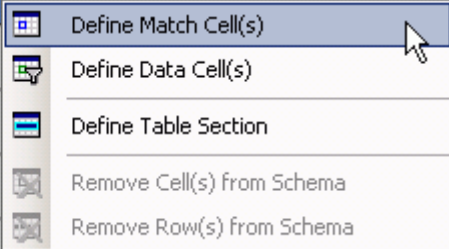
Chapter 1: Basic HTML Table Design

- Looking at the table, the top row does not repeat and always exists for the table. We will use this information to help match the table. Click and drag the cursor over the top three cells in the Table Designer to highlight the row:

Table Design		
Product Cell 1	Price Cell 2	Product ID Cell 3
Chai Cell 1	18.00 Cell 2	1 Cell 3
Chang Cell 1	19.00 Cell 2	2 Cell 3
Guarana... Cell 1	4.50 Cell 2	24 Cell 3
Sasquatch Ale Cell 1	14.00 Cell 2	34 Cell 3
Steeleye Stout Cell 1	18.00 Cell 2	35 Cell 3
Cote de Blaye Cell 1	263.50 Cell 2	38 Cell 3
Chartreuse verte Cell 1	18.00 Cell 2	39 Cell 3
Ipoh Coffee Cell 1	46.00 Cell 2	43 Cell 3
Laughing... Cell 1	14.00 Cell 2	67 Cell 3
Outback Lager Cell 1	15.00 Cell 2	70 Cell 3
Ronbrau... Cell 1	7.75 Cell 2	75 Cell 3
Lakkalikoori Cell 1	18.00 Cell 2	76 Cell 3
		Next Cell 3

- Right click the mouse over one of the cells then select **Define Match Cell(s)**.

Table Design		
Product Cell	Price Cell	Product ID Cell
Chai Cell	18.00 Cell	1 Cell
Chang Cell	19.00 Cell	2 Cell
Guarana... Cell	4.50 Cell	24 Cell
Sasquatch Ale Cell	14.00 Cell	34 Cell



- Define Match Cell(s)
- Define Data Cell(s)
- Define Table Section
- Remove Cell(s) from Schema
- Remove Row(s) from Schema

- The Match Cells appear highlighted in blue in the table. Change the names of the cells as follows using the (Name) property:

Cell	Name
matchCell_1	matchCell_Product
matchCell_2	matchCell_Price
matchCell_3	matchCell_ProductID

Chapter 1: Basic HTML Table Design

- Highlight the cell containing the link “Next”, right-click and select **Define Match Cell(s)**. Rename the cell to matchCell_Next.

The screenshot shows the HTML Table Designer application. The 'Table Design' pane displays a table with three columns: Product, Price, and Product ID. The table contains 12 rows of product data. The last row has a blue cell containing the text 'Next' and 'matchCell_Next', which is highlighted with a red rectangle. The 'Table Schema' pane on the left shows the hierarchical structure of the table, including the 'matchCell_Next' cell. The 'Properties' pane at the bottom shows the settings for the selected cell, including its name, column index, and expected text.

Table Design

Product matchCell_Product	Price matchCell_Price	Product ID matchCell_ProductID
Chai Cell	18.00 Cell	1 Cell
Chang Cell	19.00 Cell	2 Cell
Guarana... Cell	4.50 Cell	24 Cell
Sasquatch Ale Cell	14.00 Cell	34 Cell
Steeleye Stout Cell	18.00 Cell	35 Cell
Cote de Blaye Cell	263.50 Cell	38 Cell
Chartreuse verte Cell	18.00 Cell	39 Cell
Ipoh Coffee Cell	46.00 Cell	43 Cell
Laughing... Cell	14.00 Cell	67 Cell
Outback Lager Cell	15.00 Cell	70 Cell
Ronbrau... Cell	7.75 Cell	75 Cell
Lakkalikoori Cell	18.00 Cell	76 Cell
Cell	Cell	Next matchCell_Next

Table Schema

- ProductPriceProduct_IDChai18001Char
 - Row
 - matchCell_Product
 - matchCell_Price
 - matchCell_ProductID
 - Row
 - matchCell_Next

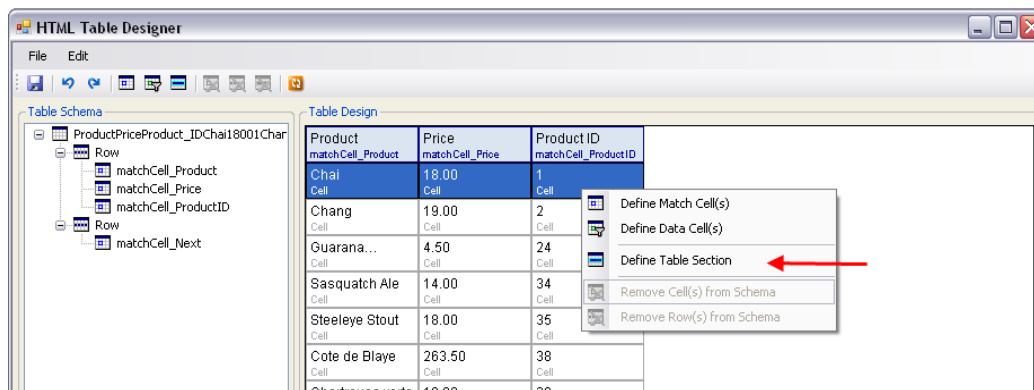
Properties

(Name)	matchCell_Next
Column	3
Expected ColSpan	1
Expected RowSpan	1
Expected Text	Next
Match Settings	

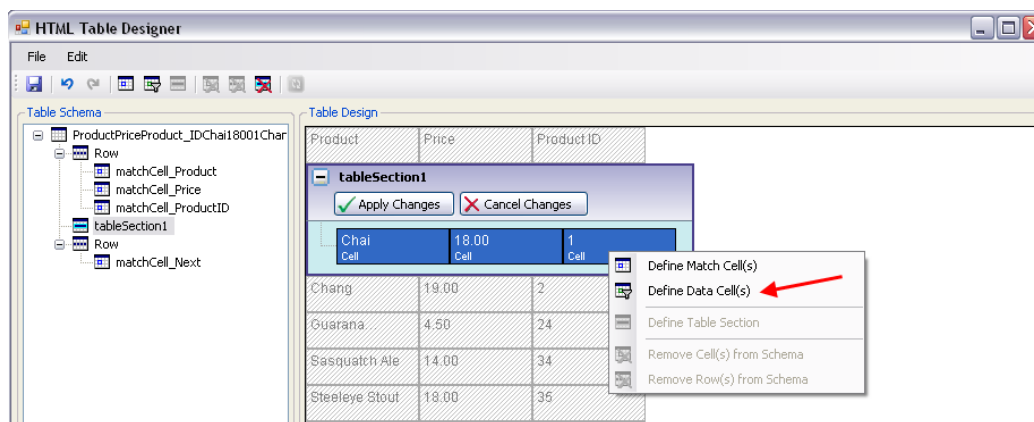
(Name)
Name of this cell.

Chapter 1: Basic HTML Table Design

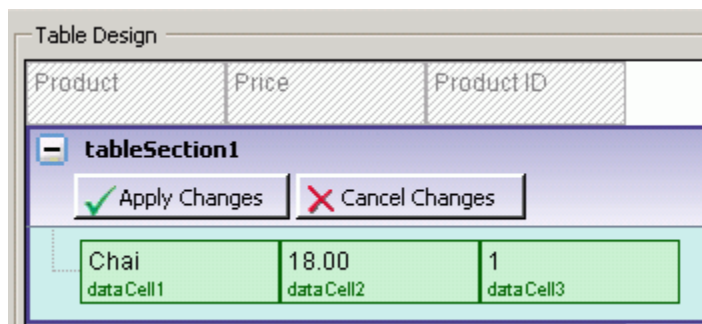
10. From the second row on, the table repeats the Products and pricing information. We handle repeating rows within an HTML table by defining them as Table Sections. Select the cells in the second row, right-click and select **Define Table Section** from the context menu.



11. Knowing that every cell in the row changes showing a different product and its associated information we will define the cells as Data Cells. Select the cells in the row. Right click on a highlighted cell and select **Define Data Cell(s)** from the context menu.



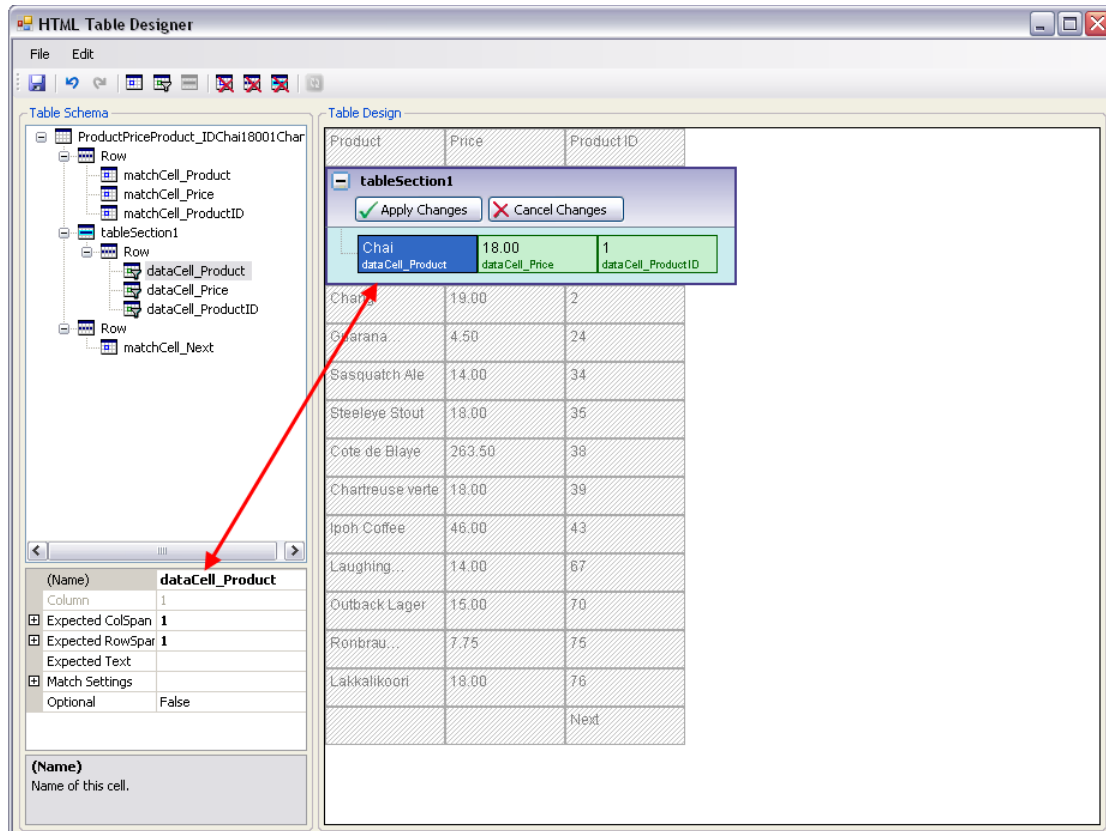
The Table Designer marks the cells as data cells and highlights them in green:



Chapter 1: Basic HTML Table Design

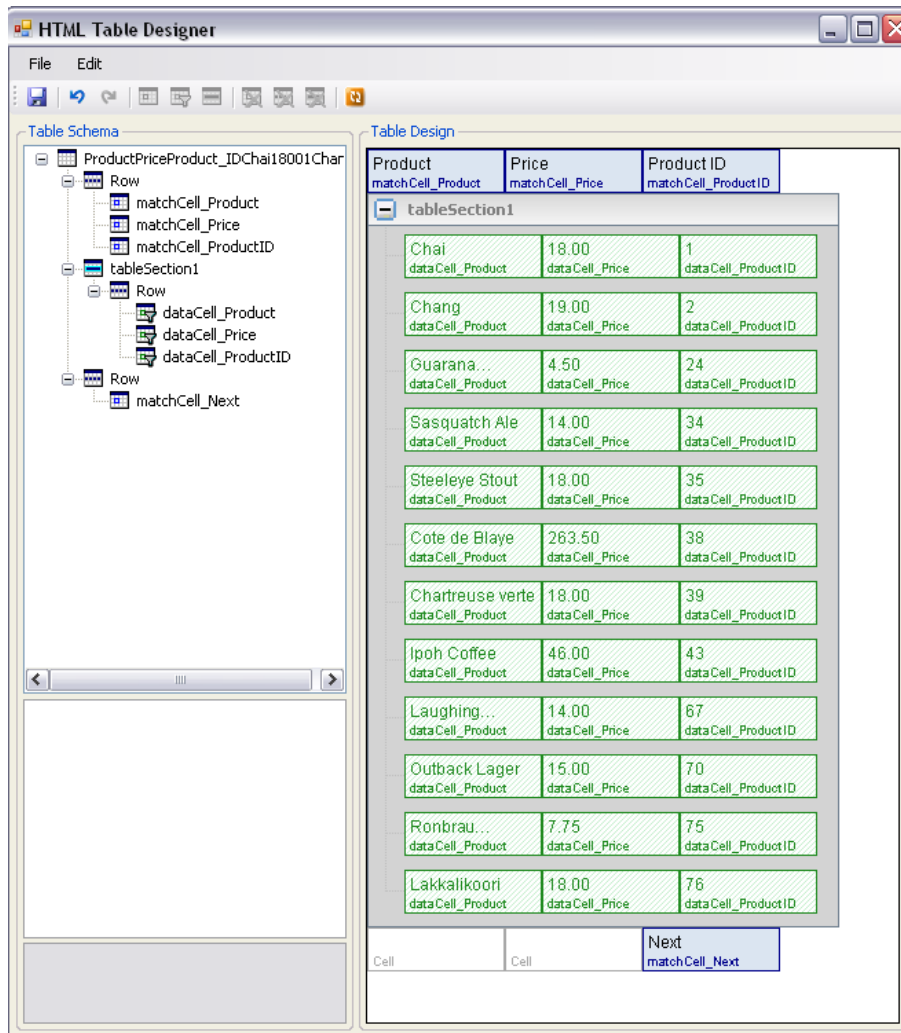
12. Rename the datacells in the table, by selecting the cell, from the designer property window modify the name, as follows;

- dataCell1 → dataCell_Product
- dataCell2 → dataCell_Price
- dataCell3 → dataCell_Product ID



Chapter 1: Basic HTML Table Design

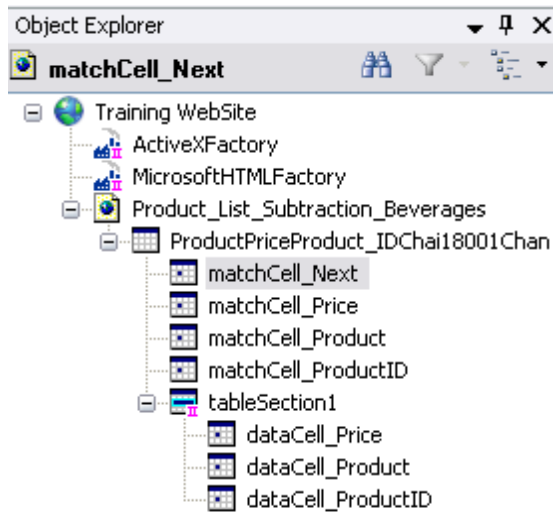
13. Select **Apply Changes** to apply the changes and to be returned to the HTML designer.



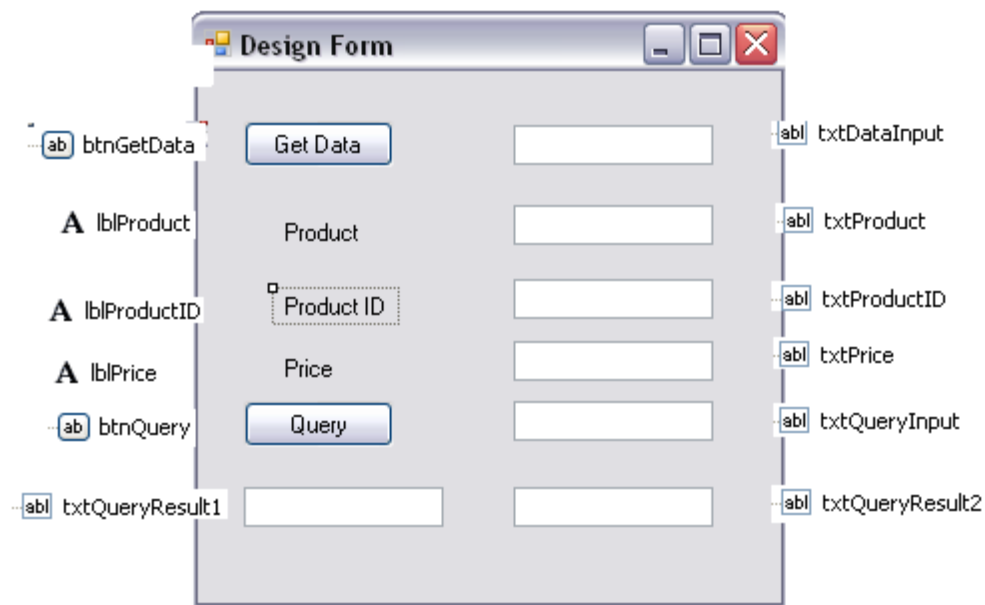
14. Select **File | Save** from the HTML Designer menu.
15. Select **File | Exit** from the HTML Designer menu to close the designer.

Chapter 1: Basic HTML Table Design

16. Stop interrogation. The table objects are created in OpenSpan Studio. An example of the Object Explorer showing the table controls follows:



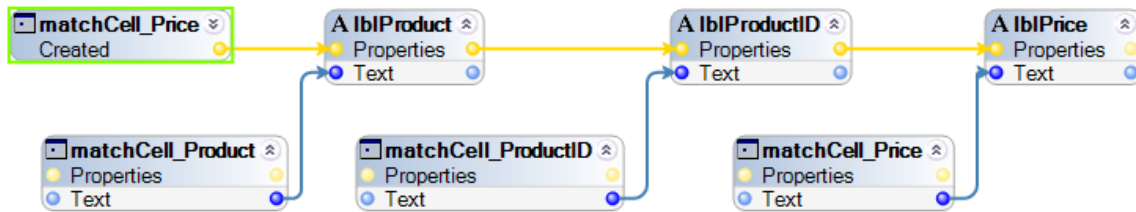
17. Add a windows form to the solution with the following buttons, labels, and textboxes:



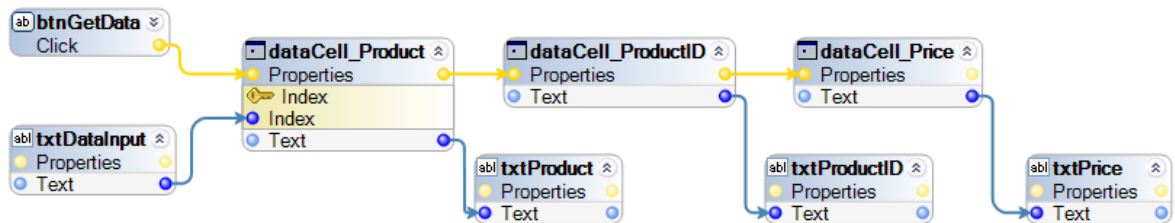
18. Save the solution.

Chapter 1: Basic HTML Table Design

19. Add an automation and name it **Change Headers**. Complete the automation as shown below. This automation automatically changes the labels of the windows form to the Headers from the HTML table.



20. Create a new automation named **Get Data** and add the components and links as shown below. This automation populates the Textboxes in the windows form with the corresponding data from the indexed row of the HTML table.



Note: When you connect the **btnGetData.Click** event to the input event port of the **dataCell_Product.Text** property, a parameter appears for you to specify which instance of the datacell you want to use. Multiple instances of the cell occur since there are multiple data rows in the table. The options are:



Be sure to select **Index** as the solution will provide a row index (from the **txtDataInput** textbox).

Chapter 1: Basic HTML Table Design

21. Save and run the solution. Type **2** in the first text box and click the **Get Data** button, your results should appear as follows:

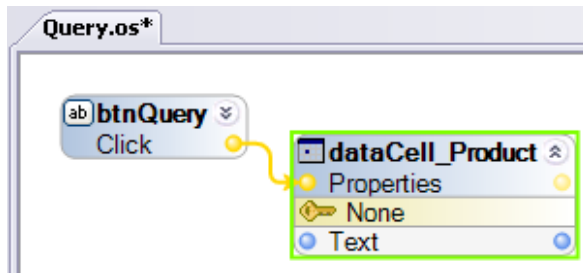


The screenshot shows a window titled "Design Form" with a standard Windows-style title bar (minimize, maximize, close buttons). The form contains the following elements:

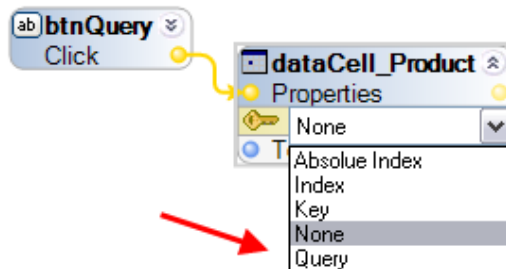
- A "Get Data" button on the left.
- A text input field containing the number "2" on the right.
- A label "Product" on the left.
- A text input field containing "Guarana Fantastica" on the right.
- A label "Product ID" on the left.
- A text input field containing "24" on the right.
- A label "Price" on the left.
- A text input field containing "4.50" on the right.
- A "Query" button on the left.
- An empty text input field on the right.
- At the bottom, there are two empty text input fields, one on the left and one on the right.


Initiating the Query Function on a Data Cell Property

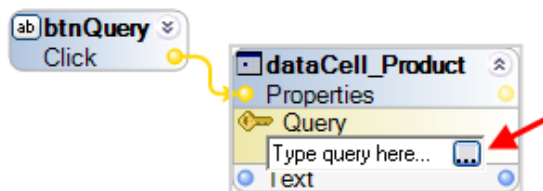
1. Create a new automation named **Query** and add the components and links as shown below. Connect the btnQuery Click event to the dataCell_Product.Text property.



2. Select "Query" as the key on the dataCell_Product object.



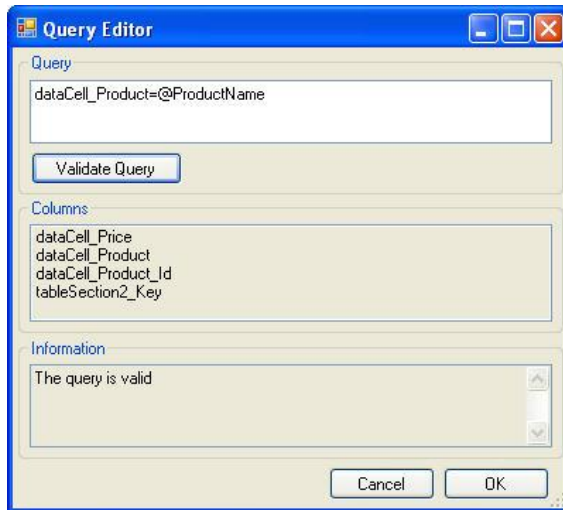
3. Define a query by pressing the  button.



4. The Query Editor window displays. Format for queries is the same as that used by the .Net DataTable component. Refer to the [MSDN website](http://msdn2.microsoft.com/en-us/library/system.data.datacolumn.expression.aspx) for details on the supported syntax (<http://msdn2.microsoft.com/en-us/library/system.data.datacolumn.expression.aspx>).
5. Type in a query (for example: `dataCell_Product=@ProductName`) and click the Validate Query button. Verify in the **Information** groupbox that the query is valid.

Chapter 1: Basic HTML Table Design

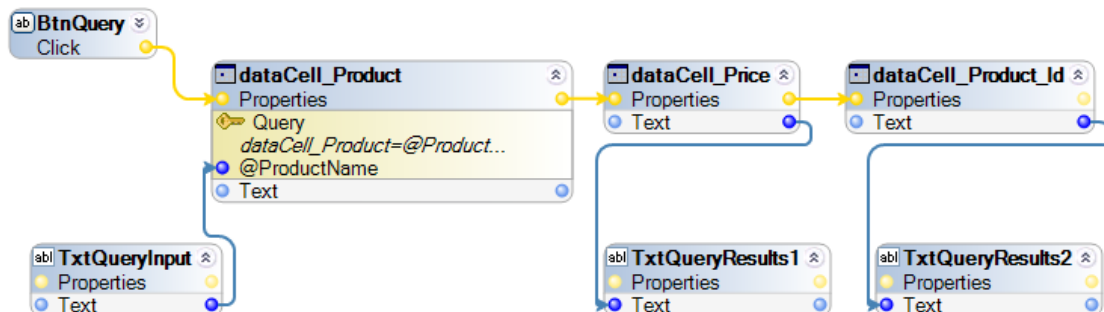
NOTE: The @ character is used in the query string to allow for dynamic input. By using the @ a data property will be created on the connection block you can connect on an automation.



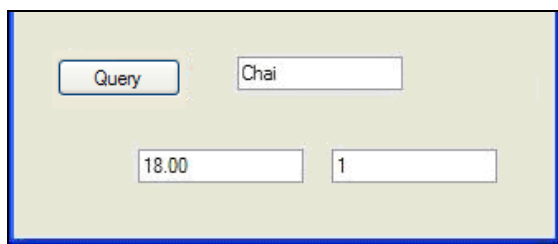
6. If the query is valid, press the OK button.

Query Function Example

7. Complete the Query automation as shown below:



8. Save and Run the solution. Enter 'Chai' in the field by the Query button and the Query button, you get the following results:

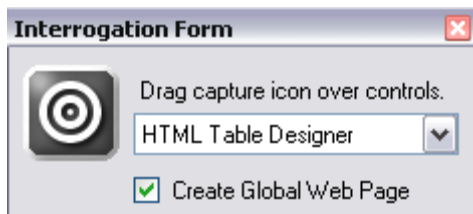


Chapter 2: Advanced Table Design

Using Advanced Table Section Match Capabilities of the HTML Table Designer

1. Create a new solution named **Parts Table**.
2. Add a Web adapter to the solution.
3. Set the StartPage for the Web adapter to:

<http://training.openspan.com/ge%20parts%20grid.htm>
4. Start interrogation.
5. In the Interrogation Form dialog, select the **Create Global Web Page** check box and select **HTML Table Designer** from the drop-list.



Chapter 2: Advanced Table Design

6. Interrogate the table.

Parts Model List - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Refresh Print View Source

Address http://training.openspan.com/ge%20parts%20grid.htm

Test

Filters				
Accessory No.	Description	Price	Availability	
WB2X2052	Range Hood Air / Grease Filter	\$36.25	In Stock	Buy it!
WB2X4267	Over-the-Range Microwave Charcoal Filter	\$29.75	In Stock	Buy it!
WB2X9883	Over the Range Microwave Charcoal Filter Kit	\$41.50	In Stock	Buy it!

Panels/Trim Kits			
Accessory No.	Description	Price	Availability
JX1327MBV	1.3 Cu. Foot Microwave Oven Deluxe Trim Kit - Black	\$124.00	Call 877-959-8688 to order.
JX1527MBW	27" Trim Kit for Countertop Microwave Models JE1660 & JE1640 - Black	\$122.75	Call 877-959-8688 to order.
JX1527MSA	27" Trim Kit for Countertop Microwave Models JE1660 & JE1640 - Stainless	\$169.25	In Stock Buy it!
JX1527MWW	27" Trim Kit for Countertop Microwave Models JE1660 & JE1640 - White	\$169.25	In Stock Buy it!
JX1530MBW	30" Trim Kit for Countertop Microwave Models JE1660 & JE1640 - Black	\$111.50	In Stock Buy it!
JX1530MWW	30" Trim Kit for Countertop Microwave Models JE1660 & JE1640 - White	\$126.50	In Stock Buy it!
JX1530MSA	30" Trim Kit for Countertop Microwave Models JE1660 & JE1640 - Stainless	\$169.25	In Stock Buy it!
JX1827AB	Microwave Oven Trim Kit	\$115.00	Call 877-959-8688 to

Note that the entire table is highlighted. This is because, from an HTML perspective, all three tables on the page, "Filters", "Panels/Trim Kits" and "Cookbooks" are the same table.

7. Select the upper table selection.



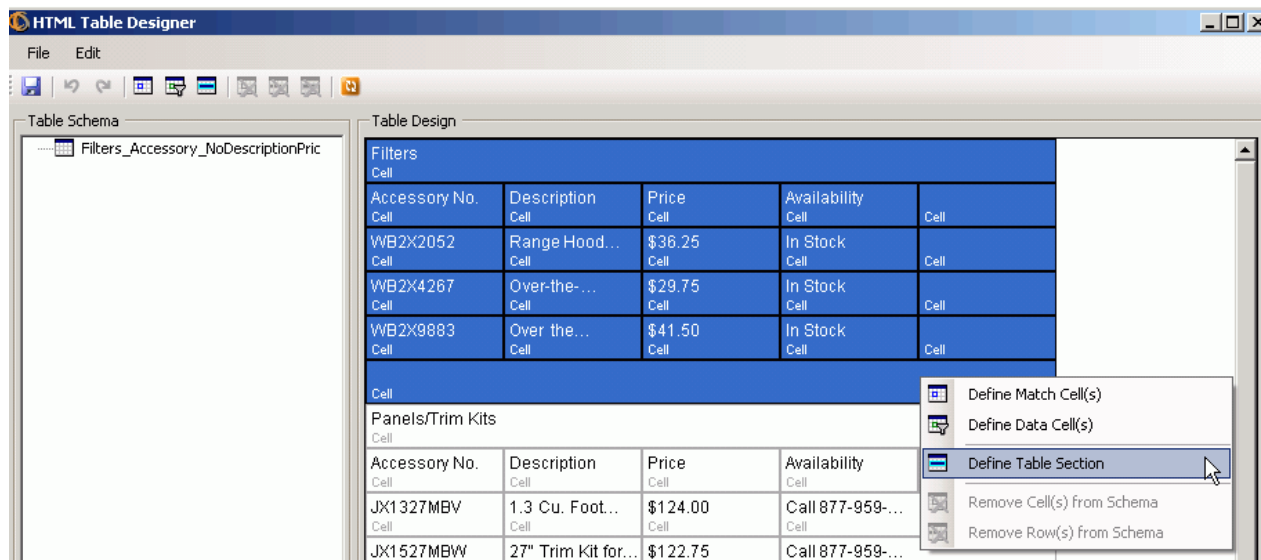
Chapter 2: Advanced Table Design

Examine the design of the table we are working with. Notice that the table layout repeats three times. Also notice that not all three tables have the same number of rows however they do have the same layout. (Table description row → Headers row → Data rows → Blank row)

Filters					
Accessory No.	Description	Description	Price	Availability	
WB2X2052	Range Hood Air / Grease Filter		\$36.25	In Stock	Buy it!
WB2X426	Over-the-Range Microwave Charcoal Filter		\$29.75	In Stock	Buy it!
WB2X9883	Over the Range Microwave Charcoal Filter Kit		\$41.50	In Stock	Buy it!

8. Knowing the repeating format of the table, select the first six rows of the table.

9. Right-click and select **Define Table Section**.



10. **Do not** click **Apply Changes** until all the sections and cells have been defined.

By selecting the first six rows OpenSpan creates a Match Pattern that fits all three table sections.

Chapter 2: Advanced Table Design

11. Define the top row within the tableSection as a **Data Cell**.

The screenshot shows the 'Table Design' window with 'tableSection1' selected. The 'Filters' section is expanded, showing a table with columns: Accessory No., Description, Price, Availability, and an empty cell. A context menu is open over the table, with 'Define Data Cell(s)' highlighted. Below the table, the 'Panels/Trim kits' section is visible.

Accessory No.	Description	Price	Availability	
WB2X2052	Range Hood...	\$36.25	In Stock	
WB2X4267	Over-the-...	\$29.75	In Stock	
WB2X9883	Over the...	\$41.50	In Stock	

12. Change the 'Expected ColSpan' property for this row to "1, 5". This is the amount of columns that this row spans, i.e. as this row is the width of the table the table can now be from 1 to 5 rows wide.

The screenshot shows two windows. The 'Table Schema' window on the left shows the 'dataCell1' property with 'Expected ColSpan' set to '1, 5'. The 'Table Design' window on the right shows the same table as in the previous screenshot.

(Name)	dataCell1
Column	1
Expected ColSpan	1, 5
Expected RowSpan	1
Expected Text	
Match Settings	
Optional	False

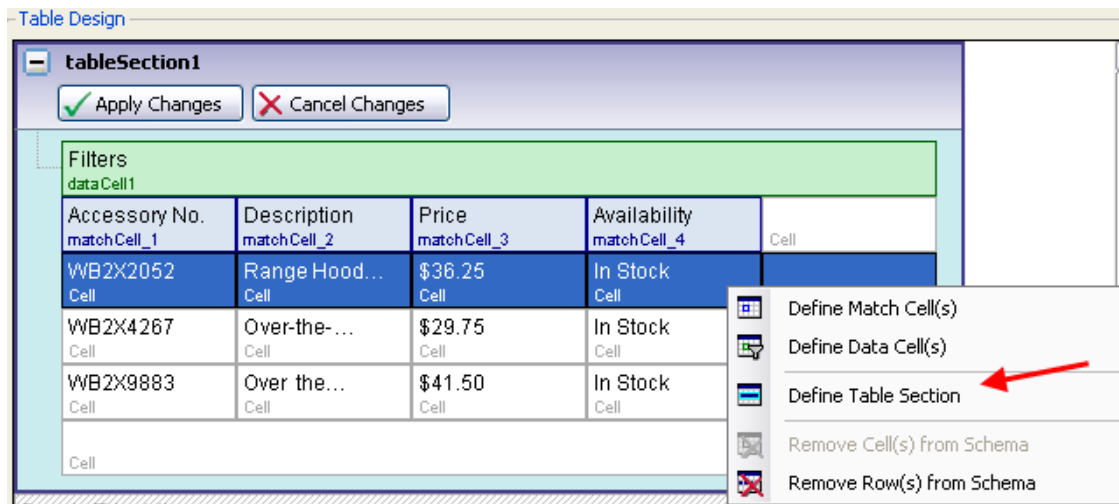
13. Define the top four cells beneath the filter as **Match Cells**.

The screenshot shows the 'Table Design' window with 'tableSection1' selected. The 'Filters' section is expanded, showing a table with columns: Accessory No., Description, Price, Availability, and an empty cell. A context menu is open over the table, with 'Define Match Cell(s)' highlighted. Below the table, the 'Panels/Trim kits' section is visible.

Accessory No.	Description	Price	Availability	
WB2X2052	Range Hood...	\$36.25	In Stock	
WB2X4267	Over-the-...	\$29.75	In Stock	
WB2X9883	Over the...	\$41.50	In Stock	

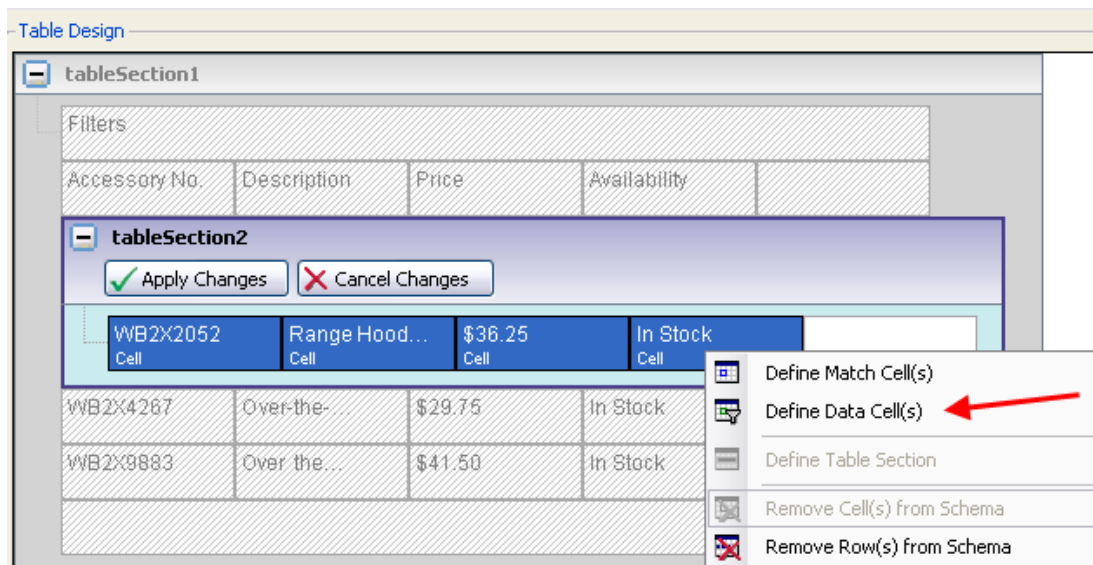
Chapter 2: Advanced Table Design

14. The next set of Row repeats in the table. Select the row and define these cells as a **Table Section** within the **Table Section**.



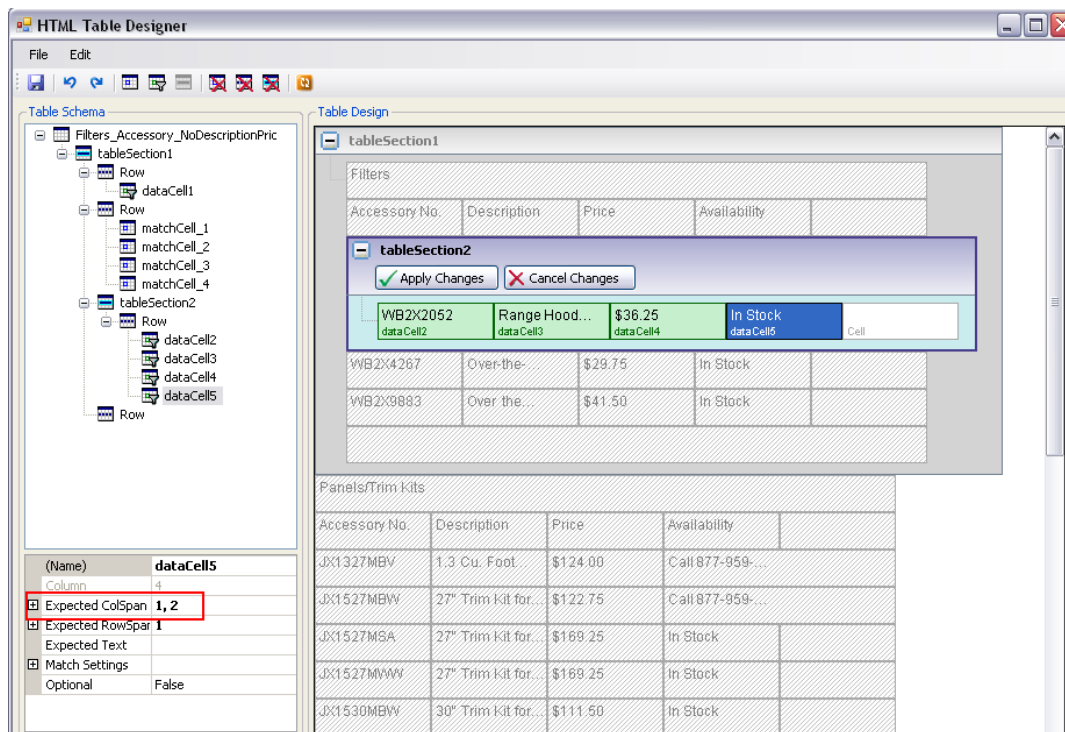
Note: It is very important that you do not define all three rows in this section as data cells. This would cause OpenSpan Studio to only match table data sections in groups of three rows.

15. Define the four cells as **Data Cells**.



Chapter 2: Advanced Table Design

16. Change the 'Expected ColSpan' property for the last selected cell on the right side of this row to "1, 2". This is the amount of columns that this cell spans, i.e. the cells underneath this one can be one or two columns wide.



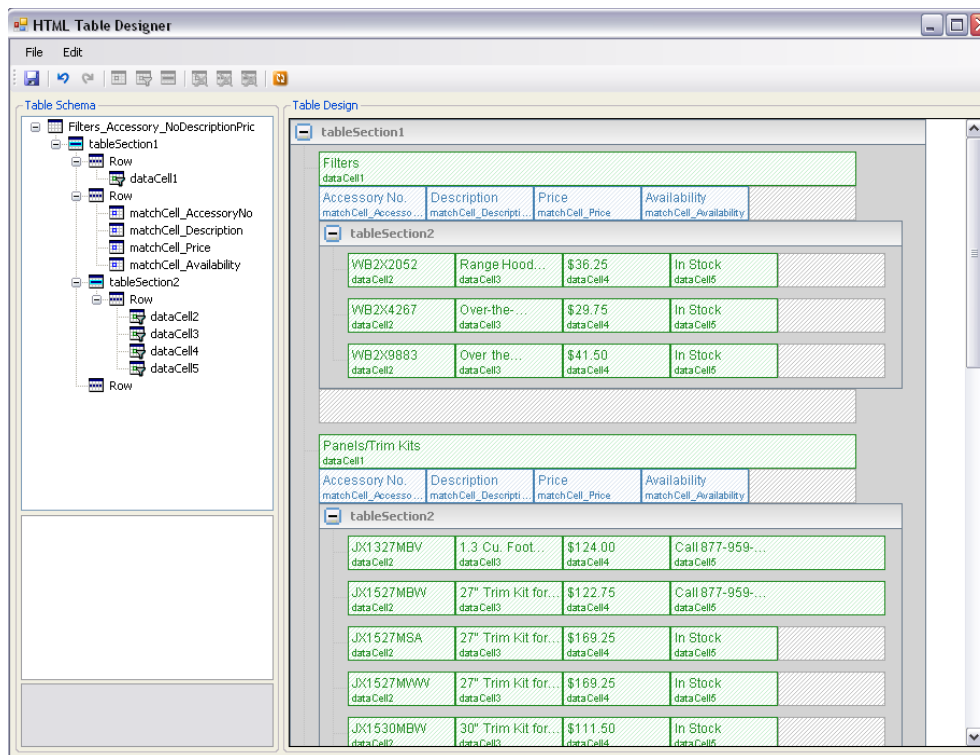
17. Click the **Apply Changes** button for the **inner table** section.

18. Rename the match cells as:

- **matchCell_AccessoryNo**
- **matchCell_Description**
- **matchCell_Price**
- **matchCell_Availability**

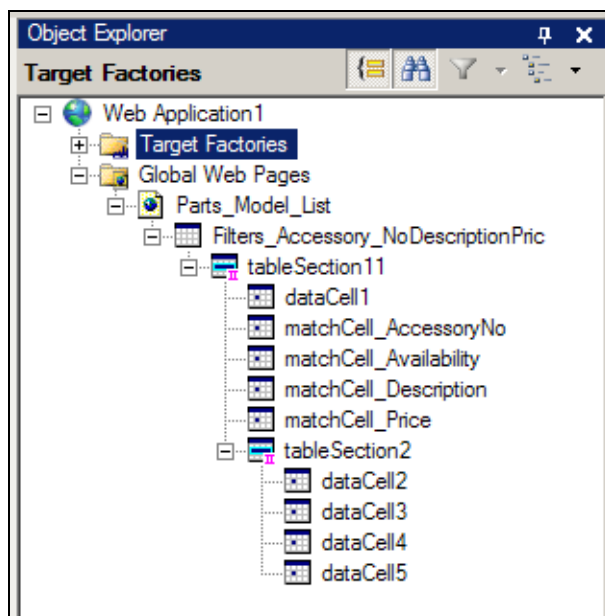
Chapter 2: Advanced Table Design

19. Click the **Apply Changes** button for the **outer table** section. Note that the entire table has now been interrogated and that the Match Pattern has been applied to all three areas of the table.



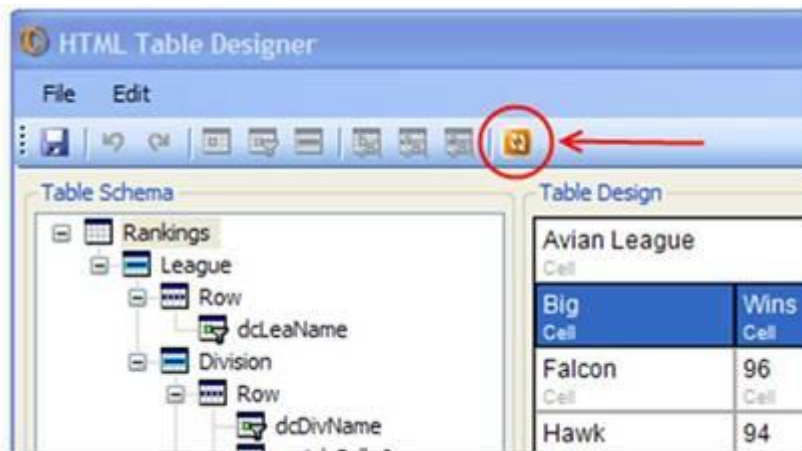
20. Save and close the **HTML Table Designer** window. The table objects are added to the solution. Stop the interrogation.

An example of the Object Explorer after interrogating the Parts table follows:



Refresh Matching

The Refresh Matching button on the HTML Table Designer rematches the entire table against a defined schema. This is normally not useful when defining a schema for a table for the first time, but may be required when debugging matching on a table that doesn't quite match the defined schema. Refresh Matching can be performed at any time, except when you have unapplied changes to be made to a table section. Refresh Matching also has a shortcut of F5.



To use the Refresh Schema button, you must Start Interrogation on the Data Table object in the Object Explorer.



Once you have selected this, you must select Debug Matching in the interrogation object and interrogate the table for which you wish to refresh the Table Matching.

Chapter 2: Advanced Table Design

Alter the data tables as necessary to match the original schema and then press the Refresh Matching button. For example, if one of the headers was misspelled "Steak" instead of "Streak," correcting this mistake will allow you to rematch the schema for the table. Make sure you save any changes once made.

The screenshot shows the HTML Table Designer application. The left pane displays the Table Schema, which includes a hierarchy of tables: Rankings, League, Row, Division, Row, Player, and Row. The right pane shows the Table Design for the League table, which is currently selected. The design shows a table with 6 columns: Big, Wins, Losses, Percent Won, Streak, and Earnings(YTD). The table is divided into sections: Division, Player, and Medium. The Player section contains 5 rows of data for different players: Falcon, Hawk, Eagle, Albatross, and Swan. The Medium section contains 5 rows of data for different birds: Raven, Owl, Duck, and Parrot. The table is currently in a state where the 'Streak' header is misspelled as 'Steak'.

Table Schema

- Rankings
 - League
 - Row
 - dcLeaName
 - Division
 - Row
 - dcDivName
 - matchCell_6
 - matchCell_7
 - matchCell_8
 - matchCell_9
 - matchCell_10
 - matchCell_11
 - Player
 - Row
 - dcPlayer
 - dcWins
 - dcLosses
 - dcPercent
 - dcStreak
 - dcEarnings
 - dcTown

Table Design

League

Avian League

Division

Apply Changes Cancel Changes

Big	Wins	Losses	Percent Won	Streak	Earnings(YTD)
dcDivName	matchCell_6	matchCell_7	matchCell_8	matchCell_9	matchCell_10
Player					
Falcon	96	66	.593	L1	\$37,000
dcPlayer	dcWins	dcLosses	dcPercent	dcStreak	dcEarnings
Hawk	94	68	.580	W2	\$55,000
dcPlayer	dcWins	dcLosses	dcPercent	dcStreak	dcEarnings
Eagle	83	79	.512	L1	\$67,000
dcPlayer	dcWins	dcLosses	dcPercent	dcStreak	dcEarnings
Albatross	69	93	.426	L2	\$21,000
dcPlayer	dcWins	dcLosses	dcPercent	dcStreak	dcEarnings
Swan	66	96	.407	W1	\$8,000
dcPlayer	dcWins	dcLosses	dcPercent	dcStreak	dcEarnings
Medium					
Wins	Losses	Percent Won	Streak	Earnings(YTD)	
Raven	96	66	.593	W1	\$11,000
Owl	88	74	.543	W1	\$197,000
Duck	79	83	.488	W1	\$98,000
Parrot	72	90	.444	L1	\$350,000

(Name) matchCell_9
Column 5
Expected ColSpan 1
Expected RowSpan 1
Expected Text **Streak**
Match Settings
Optional False

Expected Text
Inner text that is expected in this cell.

Summary

The HTML Table Designer enables you to interrogate an HTML Table. The HTML Table Designer allows you to select which cells are fixed or data cells. Once you have interrogated these cells, you can query them for additional information. By using table sections, you can apply the same set of rules to multiple areas of a table. For more information on the HTML Table and its components, refer to the OpenSpan Studio online Help