Designing a Data Model in Power BI Desktop

Lab Time: 60 minutes

Lab Folder: C:\Student\Modules\04_DataModeling\Lab

Lab Overview: In this set of lab exercises, you will continue to work on the Power BI Desktop project you started in the previous lab. At this point, you have already imported Wingtip sales data from a SQL Azure database and transformed it using the Power Query features of Power BI Desktop to create a starting point for the data model associated with your project. In this lab you will begin to leverage the Power Pivot features of Power BI Desktop to enhance this data model by adding calculated columns and calculated fields. Along the way, you will add pages and visuals to a Power BI Desktop report so you can see the effects of your modeling efforts.

Lab Dependencies: This lab assumes you have completed the previous lab titled **Designing Queries to Generate a Data Model in Power BI Desktop** in which you created a Power BI Desktop project named **Wingtip Sales Analysis.pbix**. In the previous lab you should have imported data into the data model using data from the **WingtipSalesDB** database in SQL Azure and transformed the data into a schema that is better suited for data modeling and analysis. If you would like to begin work on this lab without first completing the previous lab, use the Windows Explorer to copy the lab solution file named **Wingtip Sales Analysis.pbix** which is located in the student folder at **C:\Student\Modules\03_Queries\Lab** into the folder at **C:\Student\Projects**.

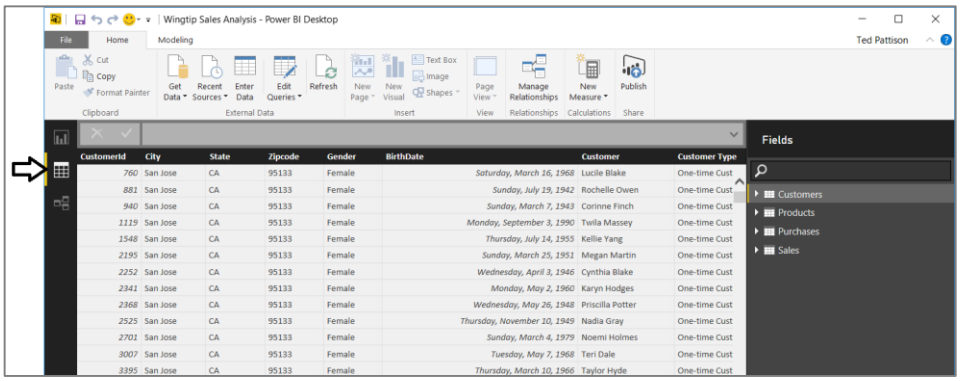
Exercise 1: Hiding and Formatting Columns in the Data Model

In this exercise you will start with the data model you created in the previous lab and begin by hiding and formatting table columns inside the data model. After that, you will create a simple report to see the effects of your formatting changes.

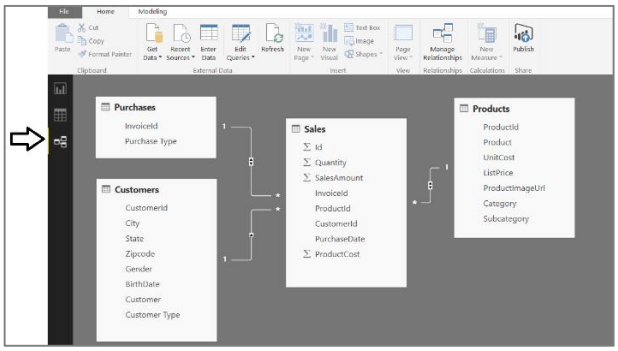
1. Launch Power BI Desktop to start a new project.
2. Open the Power BI Desktop project named **Wingtip Sales Analytics.pbix** from the previous lab located at the following path.

```
C:\Student\Projects\Wingtip Sales Analysis.pbix
```

3. When the project opens, click the table icon in the middle of the sidebar to enter data view mode.

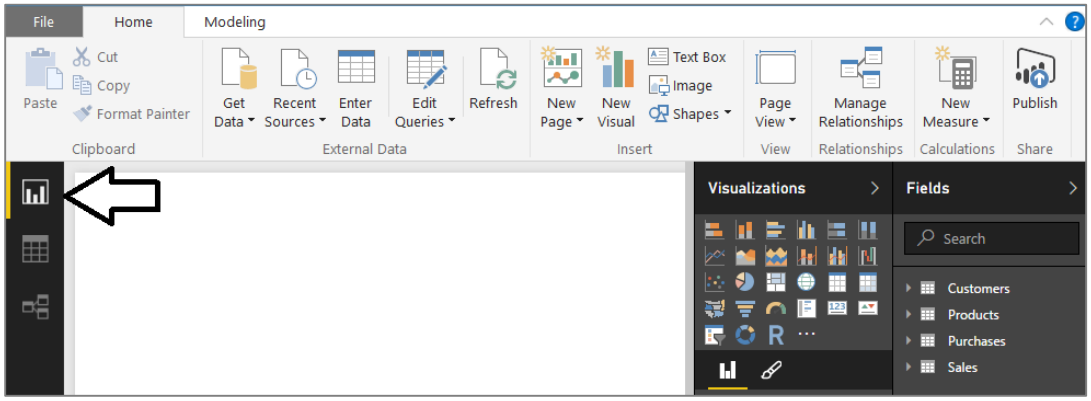


4. Take a moment to review the data in each of the four tables in the data model by clicking on the tables inside the **Fields** list.
5. Click the bottom button in the sidebar to navigate to relationship view. You should see that the four tables are arraigned in a star schema where the **Sales** table has a relationship established with each of the three other tables in the data model

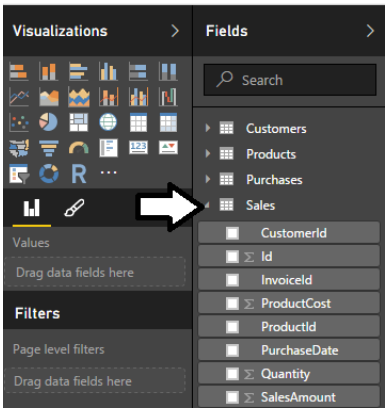


© Critical Path Training. 2016. All Rights Reserved

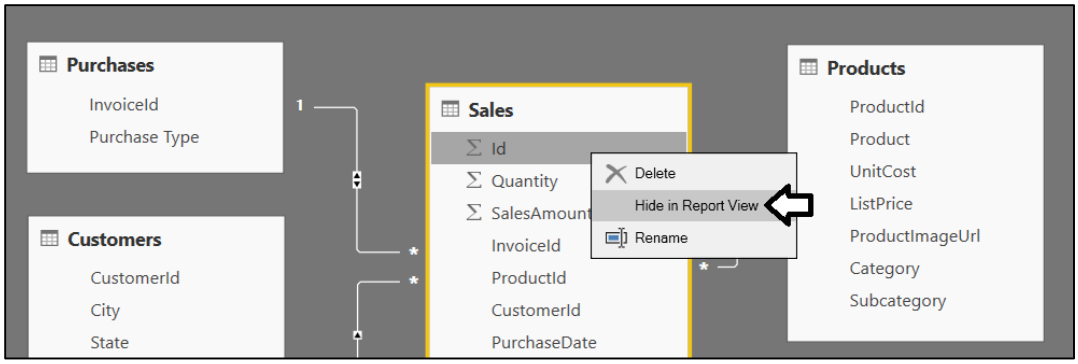
6. Now it's time to inspect the data model from a different perspective. More specifically, you will examine the data model from the perspective of a consumer who is designing reports and creating visuals using the Power BI Desktop report designer.
   a) Click the top button in the sidebar to navigate to report view.



   b) Inside the **Fields** list, use the mouse to expand the fields inside the **Sales** table. You can see that there are several fields in the **Sales** table that will never be used when designing reports such as the four identifier columns. The data model will be easier for consumers such as report designers to understand if you hide these types of fields which add unnecessary clutter.



7. Use relationship view to update the data model by hiding fields in the **Sales** table that are unnecessary to display in report view.
   a) Navigate to relationship view in the Power BI Desktop window.
   b) Using the mouse, right-click on the **Id** column in the **Sales** table and select the **Hide in Report View** command.

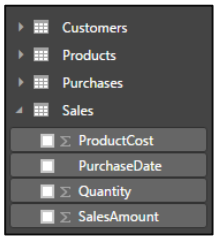


   c) Right-click on the **InvoiceId** column in the **Sales** table and select the **Hide in Report View** command.
   d) Right-click on the **CustomerId** column in the **Sales** table and select the **Hide in Report View** command.

© Critical Path Training. 2016. All Rights Reserved

e) Right-click on the **ProductId** column in the **Sales** table and select the **Hide in Report View** command.

f) Now, navigate back to report view and examine the set of fields displayed for the **Sales** table.



You should be able to see that hiding unnecessary columns from report view makes your data model easier to use. This is especially true in the scenario where you are creating a data model that other less technical people will be using to create reports & dashboards.

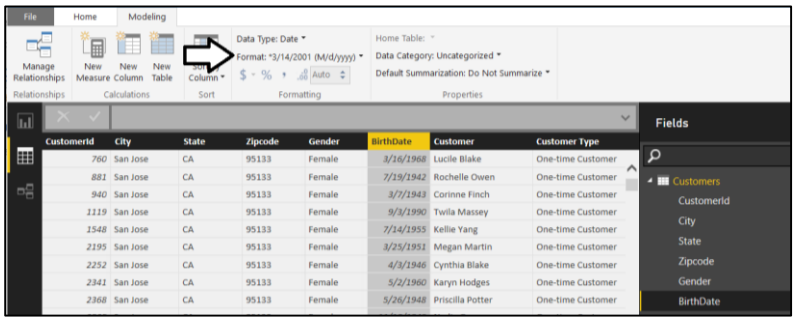
8. Modify the formatting of the **BirthDate** column in the **Customers** table.

a) In the Power BI Desktop windows, navigate back to data view.

b) In the **Fields** list on the right, select the **Customers** table to display its rows and columns.

c) Select the **BirthDate** column.

d) Modify the the **BirthDate** column formatting using the **Format** menu to select a format of **Date Time > 3/14/2001 (M/d/yyyy)**.



e) The **BirthDate** column should now reflect the change in formatting.

Gender	BirthDate	Customer
Female	3/16/1968	Lucile Blake
Female	7/19/1942	Rochelle Owen
Female	3/7/1943	Corinne Finch
Female	9/3/1990	Twila Massey

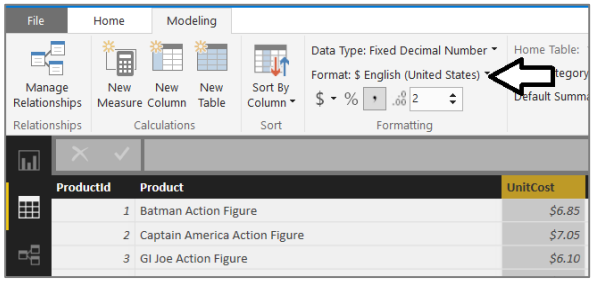
9. Modify the formatting of columns in the **Products** table.

a) In the **Fields** list on the right, select the **Products** table to display its rows and columns.

b) Select the **UnitCost** column by clicking on its column header.

c) Use the **Format** menu button in the ribbon to update the format setting to **Currency > $ English (United States)** and set the number of decimal places to **2** in the spin control located underneath the **Format** dropdown menu.

© Critical Path Training. 2016. All Rights Reserved



d) Changing the format setting of the **ListPrice** column to **$ English (United States)** and set the number of decimal places to **2** so it matches the **UnitCost** column.

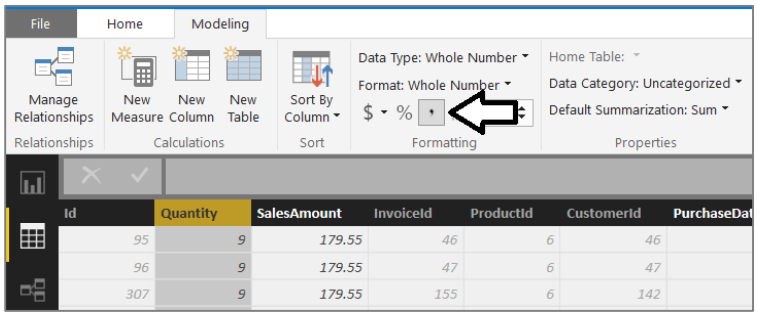
UnitCost	ListPrice
$6.85	$14.95
$7.05	$12.95
$6.1	$14.95
$2.85	$9.95

10. Modify the formatting of columns in the **Sales** table.

a) In the **Fields** list on the right, select the **Sales** table to display its rows and columns.

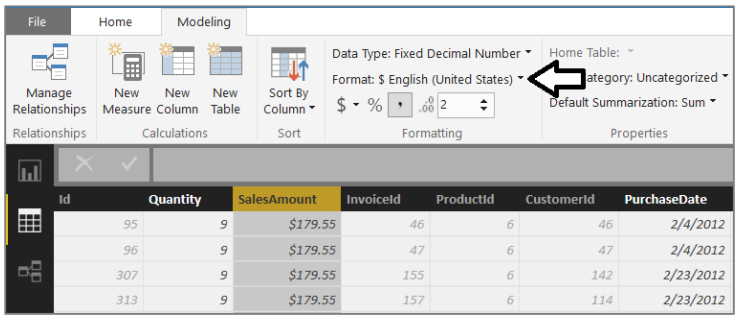
b) Select the **Quantity** column by clicking on its column header.

c) Modify the **Quantity** column by clicking to select the comma button on the ribbon to add a comma separator.



d) Select the **SalesAmount** column by clicking on its column header.

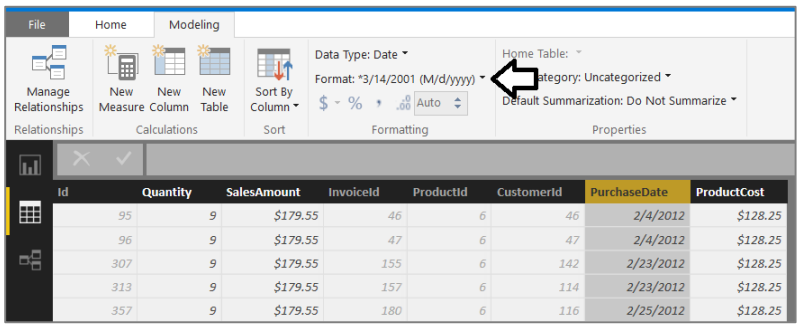
e) Modify the formatting of the **SalesAmount** column to **Currency > $ English (United States)** and set the number of decimal places to **2** in the spin control located underneath the **Format** dropdown menu



f) Select the **PurchaseDate** column by clicking on its column header.

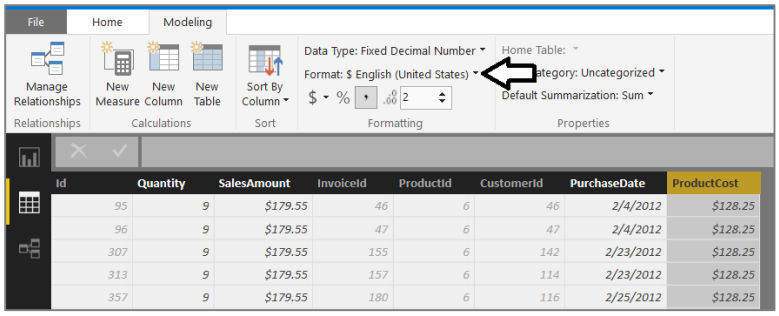
g) Modify the formatting of the **PurchaseDate** to of **Date Time > 3/14/2001 (M/d/yyyy)**.

© Critical Path Training. 2016. All Rights Reserved



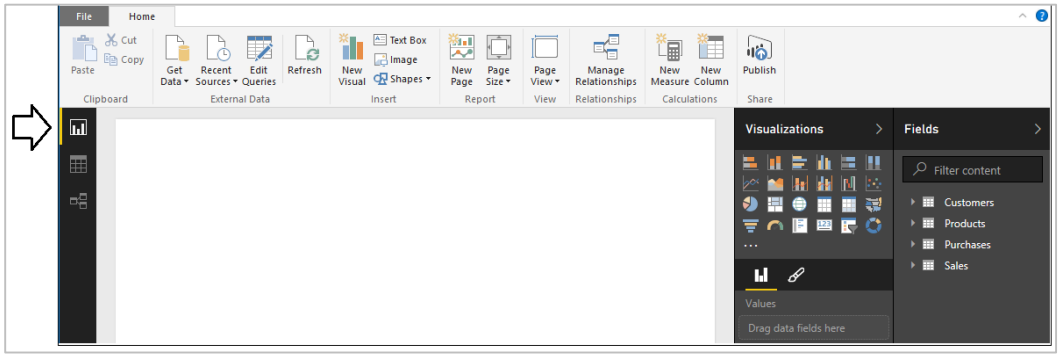
h) Select the **ProductCost** column by clicking on its column header.

i) Modify the formatting of the **ProductCost** column to **Currency> $ English (United States)** and set the number of decimal places to **2** in the spin control located underneath the **Format** dropdown menu.

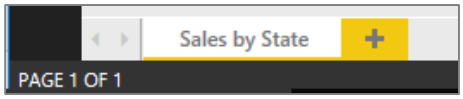


11. See the effect of your formatting by adding a visual to a report.

a) Navigate to report view. There should be an empty report for the project with a single page named **Page 1**.

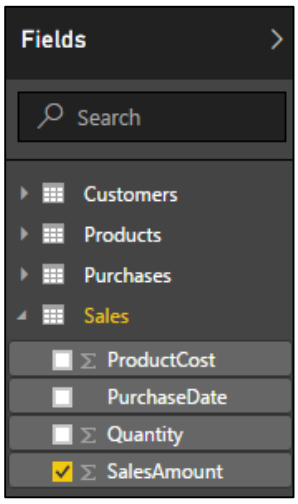


b) Change the name of the page in the report from **Page 1** to **Sales by State**.



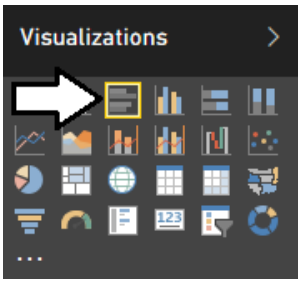
c) Create a new visual in the report by selecting the checkbox for the **SalesAmount** column in the **Fields** list.

© Critical Path Training. 2016. All Rights Reserved



d) When you select the **SalesAmount** column, Power BI Desktop will automatically add a new visual to the report based on the visualization type of **Clustered Column Chart**.

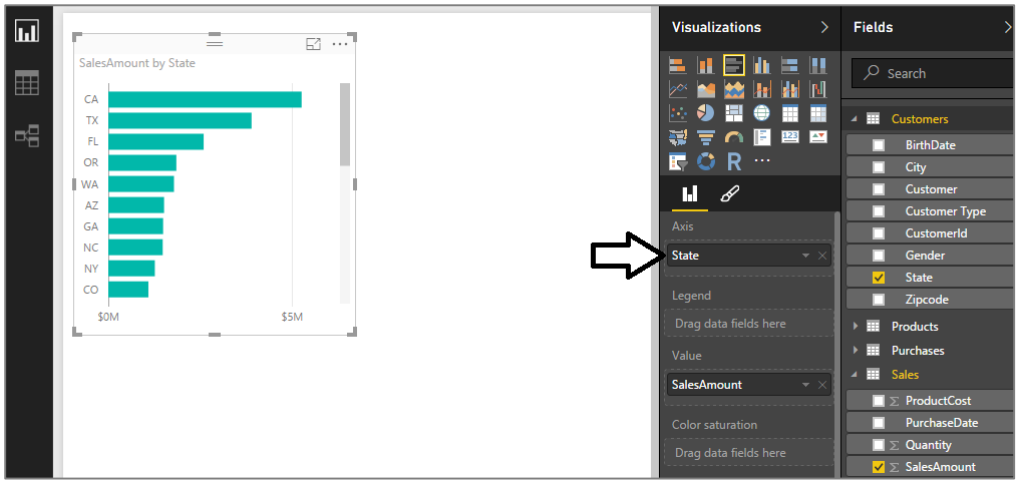
e) Click the **Clustered Bar Chart** button in the **Visualizations** list to change the visualization type to a clustered bar chart.



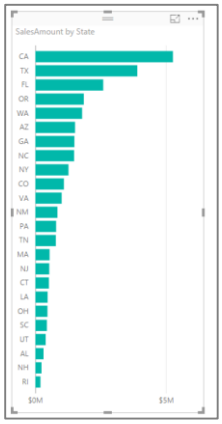
f) Since you haven't added any row labels yet, the clustered bar chart currently displays a single bar showing total sales.

© Critical Path Training. 2016. All Rights Reserved

g) Drag and drop the **States** field from the **Fields** list into the **Axis** well of the **Visualizations** pane.



h) Use your mouse to resize the visual so that it can display all the stats without a scrollbar.



i) Reposition the visual to the bottom left corner of the page as shown in the following screenshot.

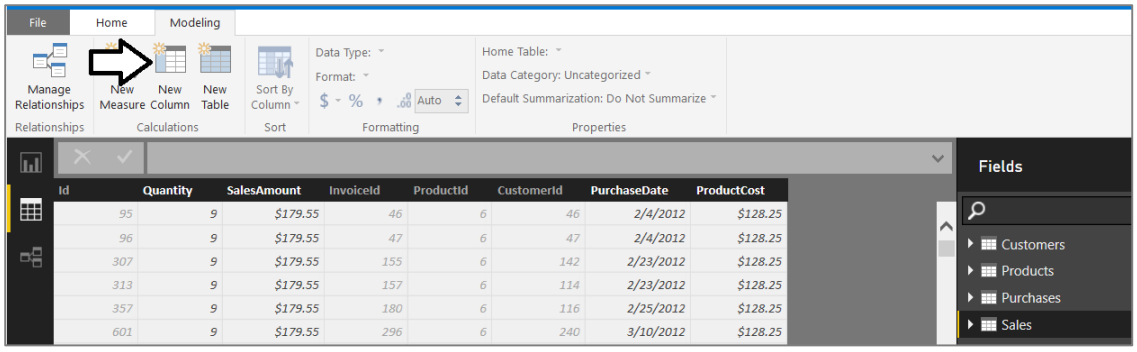
12. Save the work you have done by clicking the Save button in the upper left corner of the Power BI Desktop window.

Exercise 2: Extending the Data Model by Creating Calculated Columns

In this exercise you will create several calculated columns which will require you to write and test DAX expressions. After creating calculated columns, you will use them to enhance reports in the current project.

© Critical Path Training. 2016. All Rights Reserved

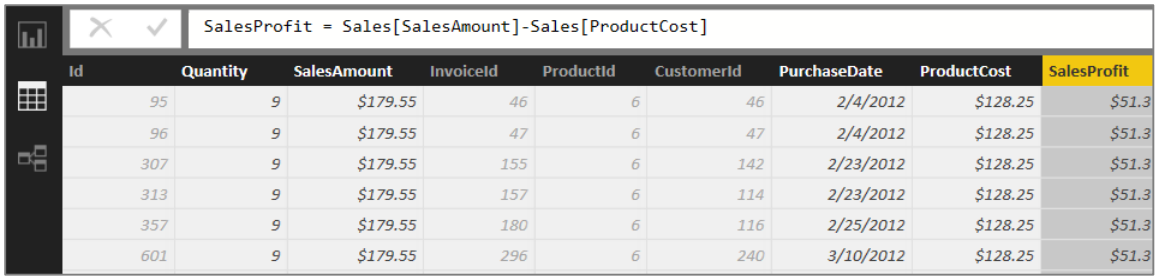
1. Add a calculated column to the **Sales** table named **SalesProfit** to determine profit by calculating the difference between **SalesAmount** and **ProductCost**.
   a) Navigate to data view.
   b) Select the **Sales** table in the **Fields** list.
   c) Create a new calculated column by clicking the **New Column** button in the ribbon.



   d) Enter to following DAX expression into the formula bar to create the calculated column named **SalesProfit**.

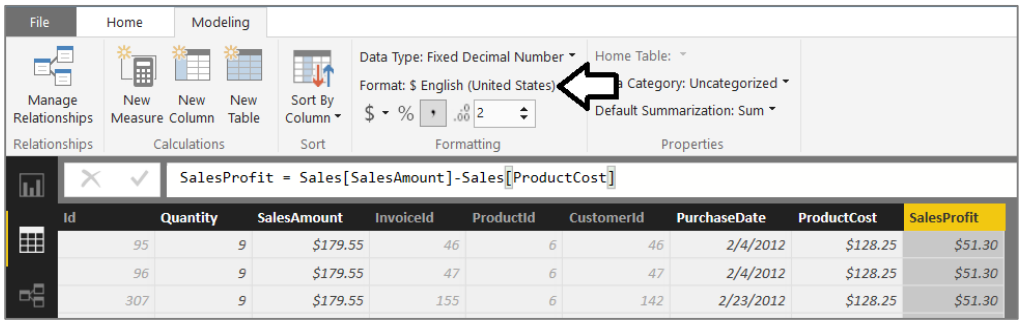
```
SalesProfit = Sales[SalesAmount]-Sales[ProductCost]
```

   e) Press the **ENTER** key to add the calculated column to the table. You should be able to see a **SalesProfit** value for each row in the **Sales** table.



Id	Quantity	SalesAmount	InvoiceId	ProductId	CustomerId	PurchaseDate	ProductCost	SalesProfit
95	9	$179.55	46	6	46	2/4/2012	$128.25	$51.3
96	9	$179.55	47	6	47	2/4/2012	$128.25	$51.3
307	9	$179.55	155	6	142	2/23/2012	$128.25	$51.3
313	9	$179.55	157	6	114	2/23/2012	$128.25	$51.3
357	9	$179.55	180	6	116	2/25/2012	$128.25	$51.3
601	9	$179.55	296	6	240	3/10/2012	$128.25	$51.3

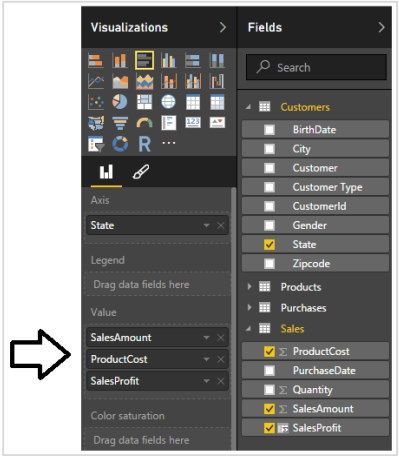
   f) Configure the column's formatting by using the **Format** menu on the ribbon to select **Currency > English (United States)**.



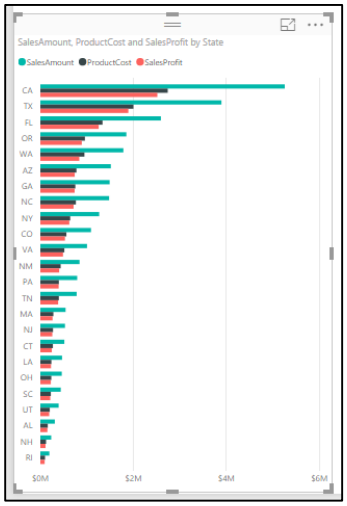
Id	Quantity	SalesAmount	InvoiceId	ProductId	CustomerId	PurchaseDate	ProductCost	SalesProfit
95	9	$179.55	46	6	46	2/4/2012	$128.25	$51.30
96	9	$179.55	47	6	47	2/4/2012	$128.25	$51.30
307	9	$179.55	155	6	142	2/23/2012	$128.25	$51.30

2. Update the clustered bar chart visual you created in the previous exercise.
   a) Navigate to report view.
   b) Select the visual you created in the previous exercise.
   c) Using your mouse, drag and drop the **ProductCost** column from the **Fields** list into the **Value** well in the **Visualizations** pane.
   d) Using your mouse, drag and drop the **SalesProfit** column from the **Fields** list into the **Value** well in the **Visualizations** pane.

© Critical Path Training. 2016. All Rights Reserved



e) You should now see that the cluster bar chart is showing additional bars for product cost and profit. Note you might have to resize the visual and make it a little taller to get rid of the scrollbars.

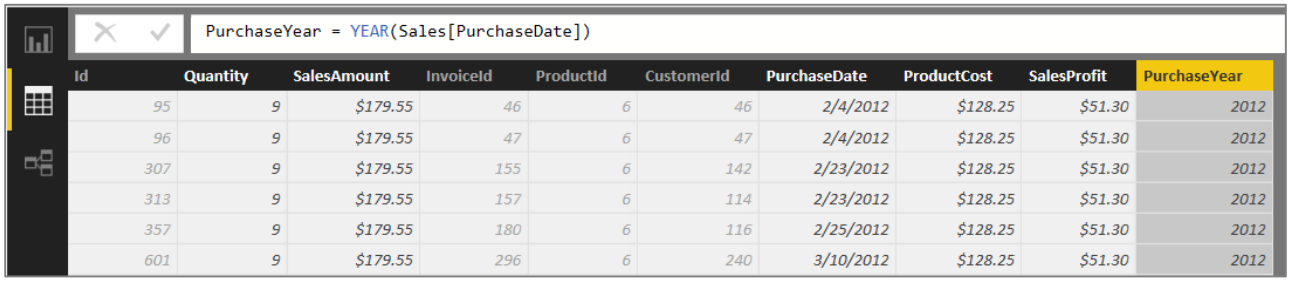


3. Add a calculated column to the **Sales** table named **PurchaseYear** to indicate the calendar year of each purchase.
   a) Navigate to data view.
   b) Select the **Sales** table in the **Fields** list.
   c) Create a new calculated column by clicking the **New Column** button in the ribbon.
   d) Enter to following DAX expression into the formula bar to create the calculated column named **PurchaseYear**.

```
PurchaseYear = YEAR(Sales[PurchaseDate])
```

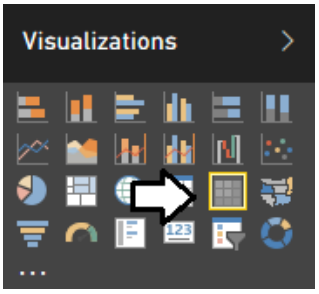
   e) Press the **ENTER** key to add the calculated column to the table. You should be able to see a calendar year value (e.g. 2012) for each row in the **Sales** table

© Critical Path Training. 2016. All Rights Reserved

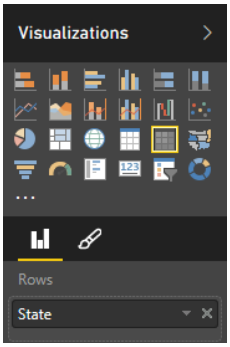


Id	Quantity	SalesAmount	InvoiceId	ProductId	CustomerId	PurchaseDate	ProductCost	SalesProfit	PurchaseYear
95	9	$179.55	46	6	46	2/4/2012	$128.25	$51.30	2012
96	9	$179.55	47	6	47	2/4/2012	$128.25	$51.30	2012
307	9	$179.55	155	6	142	2/23/2012	$128.25	$51.30	2012
313	9	$179.55	157	6	114	2/23/2012	$128.25	$51.30	2012
357	9	$179.55	180	6	116	2/25/2012	$128.25	$51.30	2012
601	9	$179.55	296	6	240	3/10/2012	$128.25	$51.30	2012

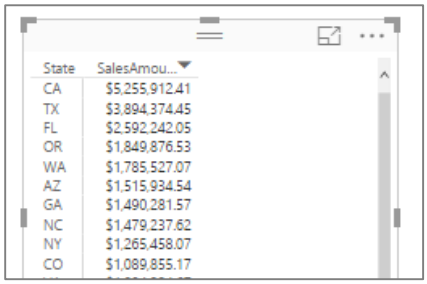
4. Use the new calculated column named **PurchaseYear** in a report
   a) Navigate to report view.
   b) Make sure that no visuals are selected on the page so that you can create a new visual.
   c) Select the checkbox for the **SalesAmount** column in the **Fields** list to create a new visual.
   d) Click the **Matrix** button in the **Visualizations** list to change the visualization type to a matrix.



   e) Add a second field to the visual by dragging the **State** column from the **Customers** table in the **Fields** list into the **Rows** well to displays a separate row for each state.



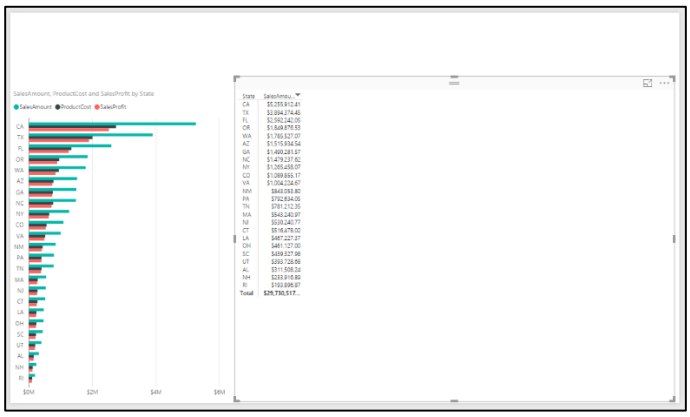
   f) Now the matrix should display a row for each state that shows the aggregated sum of the **SalesAmount** column.



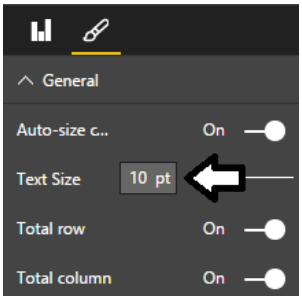
State	SalesAmou...
CA	$5,255,912.41
TX	$3,894,374.45
FL	$2,592,242.05
OR	$1,849,876.53
WA	$1,785,527.07
AZ	$1,515,934.54
GA	$1,490,281.57
NC	$1,479,237.62
NY	$1,265,458.07
CO	$1,089,855.17

   g) Use the mouse to reposition the visual so it takes up the bottom right corner of the page as shown in the following screenshot.

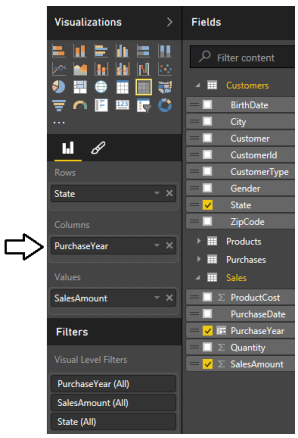
© Critical Path Training. 2016. All Rights Reserved



h) Increase the font size of the Matrix visual by changing the **Text Size** property to **10pt**. The Text Size property can be found in the Format properties pane inside the **General** section.



i) Extend the matrix visual to pivot on the **PurchaseYear** column. Accomplish this by using the mouse to drag the **PurchaseYear** column from the **Fields** list and drop it into the **Columns** well in the **Visualization** pane



j) The visual should now display a column for each year from 2012 through 2015.

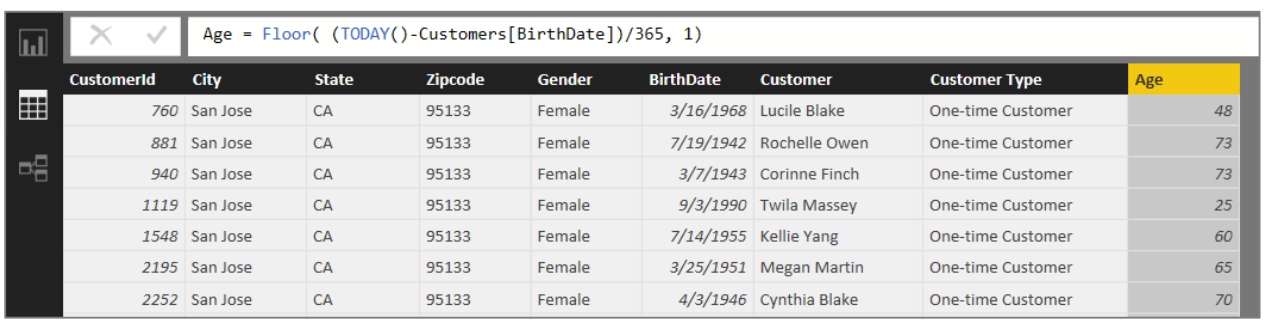
© Critical Path Training. 2016. All Rights Reserved

State	2012	2013	2014	2015	Total
CA	$955,304.01	$1,439,664.26	$1,500,626.10	$1,360,318.04	**$5,255,912.41**
TX		$838,439.67	$1,421,216.63	$1,634,718.15	**$3,894,374.45**
FL		$252,368.21	$951,663.56	$1,388,210.28	**$2,592,242.05**
OR	$387,033.51	$563,852.60	$512,936.37	$386,054.05	**$1,849,876.53**
WA	$364,797.81	$467,791.13	$477,825.82	$475,112.31	**$1,785,527.07**
AZ	$103,345.96	$437,648.46	$490,128.54	$484,811.58	**$1,515,934.54**
GA		$190,768.06	$617,963.81	$681,549.70	**$1,490,281.57**
NC		$124,681.62	$567,058.87	$787,497.13	**$1,479,237.62**
NY		$50,748.68	$461,902.86	$752,806.53	**$1,265,458.07**
CO	$8,494.19	$268,693.86	$379,118.18	$433,548.94	**$1,089,855.17**
VA		$62,695.55	$399,074.09	$542,455.03	**$1,004,224.67**
NM	$106,009.88	$228,703.65	$293,862.07	$214,478.20	**$843,053.80**
PA		$45,168.53	$315,564.82	$431,900.70	**$792,634.05**
TN		$86,663.58	$319,379.52	$375,169.25	**$781,212.35**
MA		$11,075.41	$196,888.80	$335,276.76	**$543,240.97**
NJ		$25,923.36	$228,423.14	$275,894.27	**$530,240.77**
CT		$21,904.29	$180,613.64	$313,960.09	**$516,478.02**
LA		$48,411.68	$182,881.19	$235,934.50	**$467,227.37**
OH		$26,620.80	$205,217.83	$229,288.37	**$461,127.00**
SC		$34,574.49	$163,703.11	$241,050.38	**$439,327.98**
UT	$19,000.85	$89,363.12	$138,379.17	$146,985.54	**$393,728.68**
AL		$28,025.76	$108,814.70	$174,667.78	**$311,508.24**
NH		$3,831.90	$87,681.30	$142,403.69	**$233,916.89**
RI		$8,558.40	$73,326.51	$112,011.96	**$193,896.87**
Total	**$1,943,986.21**	**$5,356,177.07**	**$10,274,250.63**	**$12,156,103.23**	**$29,730,517.14**

5. Add a calculated column to the **Customers** table named **Age** to indicate the age of the customer.
   a) Navigate to data view.
   b) Select the **Customers** table in the **Fields** list.
   c) Create a new calculated column by clicking the **New Column** button in the ribbon.
   d) Enter the following DAX expression into the formula bar to create the calculated column named **Age**.

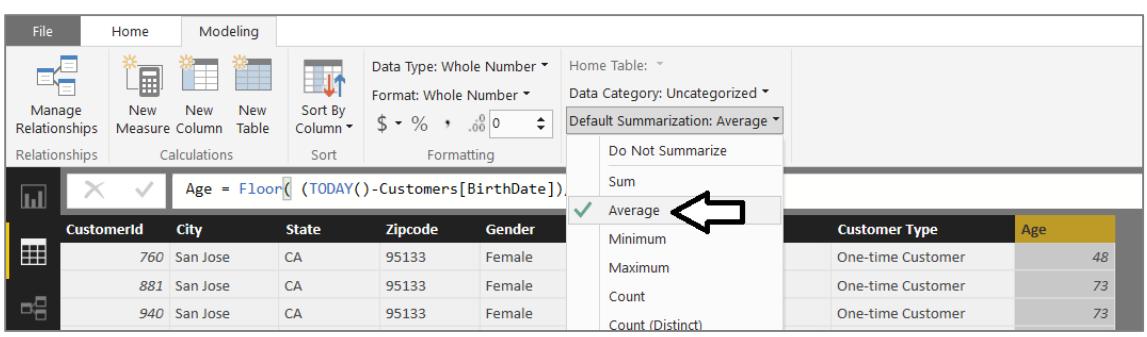
```
Age = Floor( (TODAY()-Customers[BirthDate])/365, 1)
```

   e) Press the **ENTER** key to add the calculated column. You should be able to see a whole number for the age of each customer.



CustomerId	City	State	Zipcode	Gender	BirthDate	Customer	Customer Type	Age
760	San Jose	CA	95133	Female	3/16/1968	Lucile Blake	One-time Customer	48
881	San Jose	CA	95133	Female	7/19/1942	Rochelle Owen	One-time Customer	73
940	San Jose	CA	95133	Female	3/7/1943	Corinne Finch	One-time Customer	73
1119	San Jose	CA	95133	Female	9/3/1990	Twila Massey	One-time Customer	25
1548	San Jose	CA	95133	Female	7/14/1955	Kellie Yang	One-time Customer	60
2195	San Jose	CA	95133	Female	3/25/1951	Megan Martin	One-time Customer	65
2252	San Jose	CA	95133	Female	4/3/1946	Cynthia Blake	One-time Customer	70

   f) Use the **Default Summarization** dropdown menu in the ribbon to change the default summarization setting for **Age** column from **Sum** to **Average**.



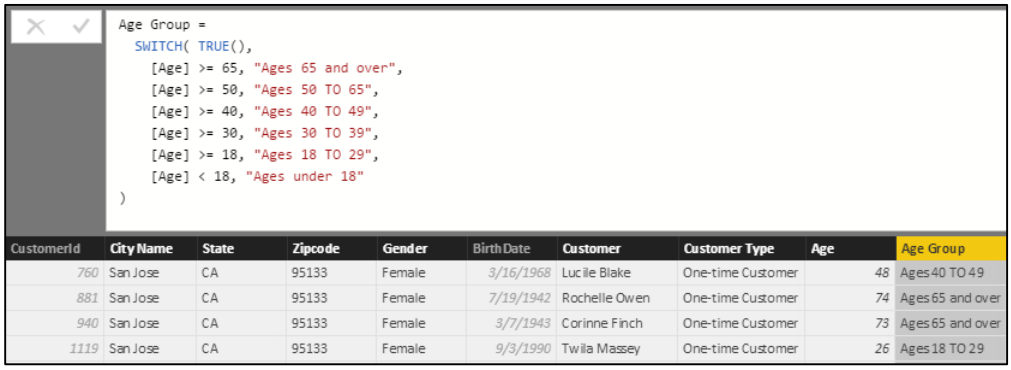
6. Add a calculated column to the **Customers** table named **Age Group** to break customers up into age-based sets.
   a) Create a new calculated column in the **Sales** table by clicking the **New Column** button in the ribbon.

© Critical Path Training. 2016. All Rights Reserved

b) Enter to following DAX expression into the formula bar to create the calculated column named **Age Group**.

```
Age Group =
  SWITCH( TRUE(),
    [Age] >= 65, "Ages 65 and over",
    [Age] >= 50, "Ages 50 TO 65",
    [Age] >= 40, "Ages 40 TO 49",
    [Age] >= 30, "Ages 30 TO 39",
    [Age] >= 18, "Ages 18 TO 29",
    [Age] < 18, "Ages under 18"
)
```

c) After creating the calculated column, you should be able to verify that the **Age Group** column calculates a value for each customer row which places that customer in a bucket for a particular age group.



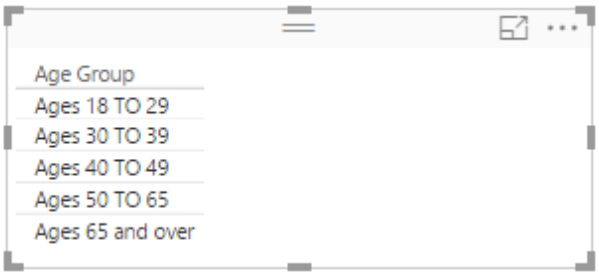
CustomerId	City Name	State	Zipcode	Gender	Birth Date	Customer	Customer Type	Age	Age Group
760	San Jose	CA	95133	Female	3/16/1968	Lucile Blake	One-time Customer	48	Ages 40 TO 49
881	San Jose	CA	95133	Female	7/19/1942	Rochelle Owen	One-time Customer	74	Ages 65 and over
940	San Jose	CA	95133	Female	3/7/1943	Corinne Finch	One-time Customer	73	Ages 65 and over
1119	San Jose	CA	95133	Female	9/3/1990	Twila Massey	One-time Customer	26	Ages 18 TO 29

7. Use the **Age Group** calculated column as a row label in a matrix visual.

a) Navigate to report view.

b) Make sure that no visuals as selected on the page.

c) Create a new visual in the report by selecting the checkbox for the **Age Group** column in the **Fields** list.

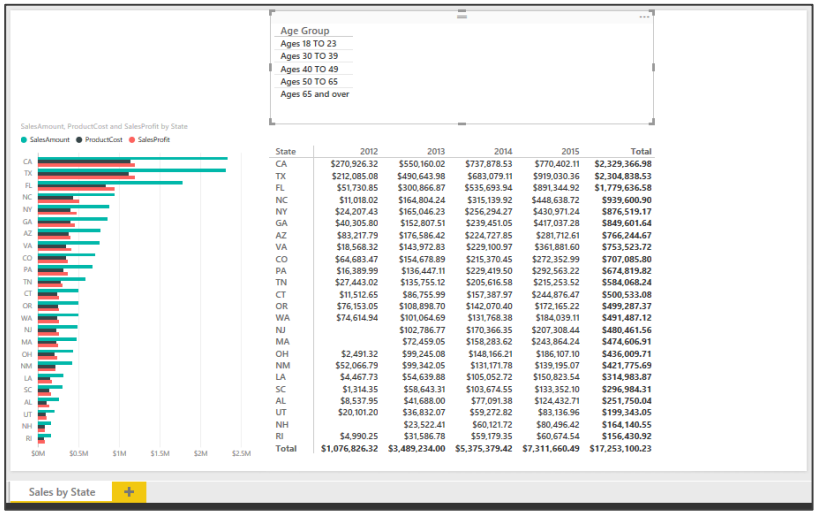


Note that this visual does not display a row for the age demographic value of **Age under 18** even though you added that value to the DAX formula. The reason for this is that the Wingtip Sales database you are working with does not contain any customers under 18 years of age so the age demographic value of **Age under 18** is automatically filtered out of the visual.

d) Change the visualization type of the new visual by clicking the **Matrix** button in the **Visualizations** list.

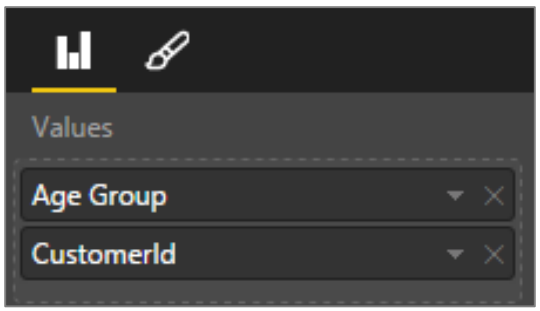
e) Use the mouse to resize and reposition the visual so it appears on top of the matrix you created earlier.

© Critical Path Training. 2016. All Rights Reserved

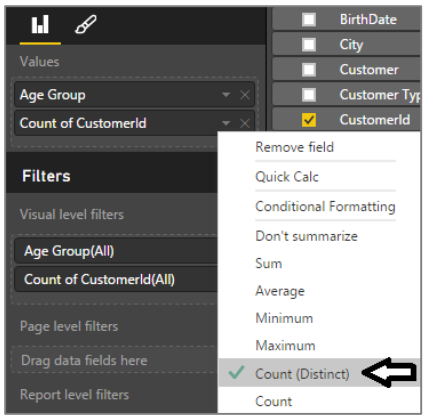


State	2012	2013	2014	2015	Total
CA	$270,926.32	$550,160.02	$737,878.53	$770,402.11	$2,329,366.98
TX	$212,085.08	$490,643.98	$683,079.11	$919,030.36	$2,304,838.53
FL	$51,730.85	$300,866.87	$535,693.94	$891,344.92	$1,779,636.58
NC	$11,018.02	$164,804.24	$315,139.92	$448,638.72	$939,600.90
NY	$24,207.43	$165,046.23	$256,294.27	$430,971.24	$876,519.17
GA	$40,305.80	$152,807.51	$239,451.05	$417,037.28	$849,601.64
AZ	$83,217.79	$176,586.42	$224,727.85	$281,712.61	$766,244.67
VA	$18,568.32	$143,972.83	$229,100.97	$361,881.60	$753,523.72
CO	$64,683.47	$154,678.89	$215,370.45	$272,352.99	$707,085.80
PA	$16,389.99	$136,447.11	$229,419.50	$292,563.22	$674,819.82
TN	$27,443.02	$135,755.12	$205,616.58	$215,253.52	$584,068.24
CT	$11,512.65	$86,755.99	$157,387.97	$244,876.47	$500,533.08
OR	$76,153.05	$108,898.70	$142,070.40	$172,165.22	$499,287.37
WA	$74,614.94	$101,064.69	$131,768.38	$184,039.11	$491,487.12
NJ		$102,786.77	$170,366.35	$207,308.44	$480,461.56
MA		$72,459.05	$158,283.62	$243,864.24	$474,606.91
OH	$2,491.32	$99,245.08	$148,166.21	$186,107.10	$436,009.71
NM	$52,066.79	$99,342.05	$131,171.78	$139,195.07	$421,775.69
LA	$4,467.73	$54,639.88	$105,052.72	$150,823.54	$314,983.87
SC	$1,314.35	$58,643.31	$103,674.55	$133,352.10	$296,984.31
AL	$8,537.95	$41,688.00	$77,091.38	$124,432.71	$251,750.04
UT	$20,101.20	$36,832.07	$59,272.82	$83,136.96	$199,343.05
NH		$23,522.41	$60,121.72	$80,496.42	$164,140.55
RI	$4,990.25	$31,586.78	$59,179.35	$60,674.54	$156,430.92
Total	$1,076,826.32	$3,489,234.00	$5,375,379.42	$7,311,660.49	$17,253,100.23

f) Now it's time to modify the table visual by adding a new column to show the count of customers in each age group. Accomplish this by dragging the **CustomerId** column from the **Customers** table in the **Fields** list and dropping it into the **Values** well in the **Visualizations** pane. When you drop the **CustomerId** column into the **Values** well, make sure to drop it so it appears underneath the **Age Group** column.

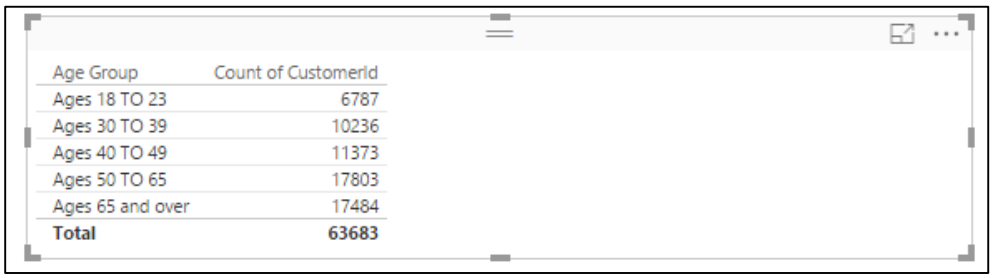


g) Click on the dropdown menu of the **CustomerId** field inside the **Values** well so you can see and change the aggregation that will be performed. Select the aggregation type that is **Count (Distinct)**.



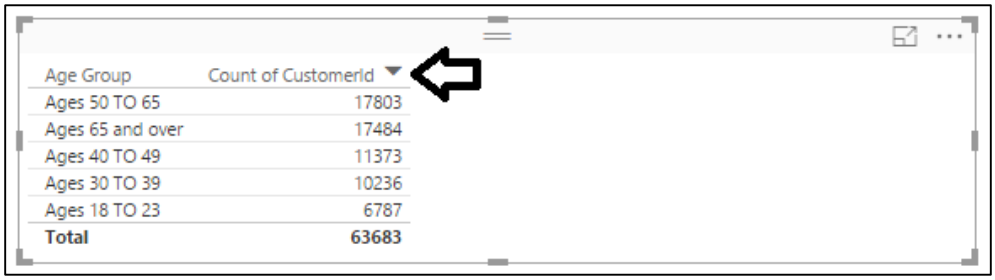
h) Now the new visual should display one row per age group which includes a total count of customers.

© Critical Path Training. 2016. All Rights Reserved



Age Group	Count of CustomerId
Ages 18 TO 23	6787
Ages 30 TO 39	10236
Ages 40 TO 49	11373
Ages 50 TO 65	17803
Ages 65 and over	17484
Total	**63683**

i) Click on the column header for the **Count of CustomerId** column to sort the age groups with most customers to the top.

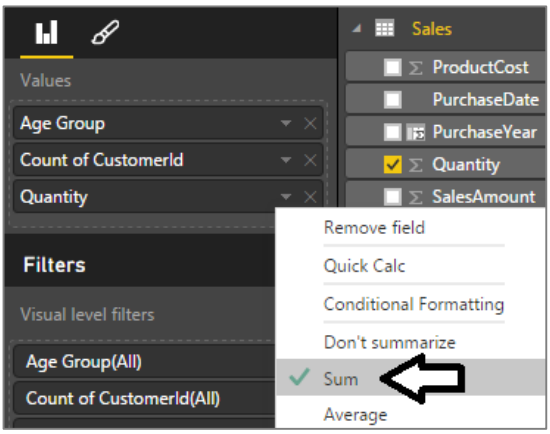


Age Group	Count of CustomerId
Ages 50 TO 65	17803
Ages 65 and over	17484
Ages 40 TO 49	11373
Ages 30 TO 39	10236
Ages 18 TO 23	6787
Total	**63683**

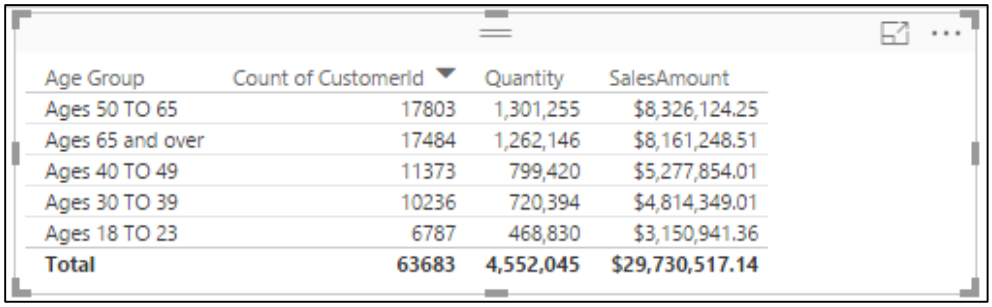
j) Drag the **Quantity** column from the **Sales** table and drop it into the **Values** well in the **Visualizations** pane.

k) Drag the **SalesAmount** column from the **Sales** table and drop it into the **Values** well in the **Visualizations** pane.

l) Configure the summarization type for the **Quantity** field and the **SalesAmount** field to perform a **Sum**.



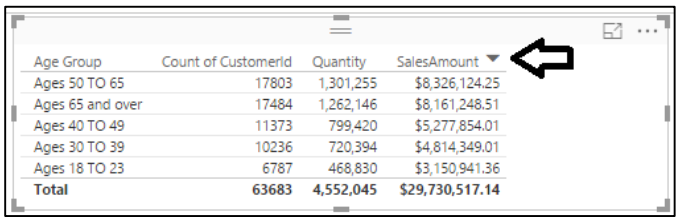
m) The visual should now display a new **Quantity** column as well as a **SalesAmount** column displaying aggregated totals.



Age Group	Count of CustomerId	Quantity	SalesAmount
Ages 50 TO 65	17803	1,301,255	$8,326,124.25
Ages 65 and over	17484	1,262,146	$8,161,248.51
Ages 40 TO 49	11373	799,420	$5,277,854.01
Ages 30 TO 39	10236	720,394	$4,814,349.01
Ages 18 TO 23	6787	468,830	$3,150,941.36
Total	**63683**	**4,552,045**	**$29,730,517.14**

n) Sort the rows in the table visual by clicking in the column header for the **SalesAmount** column so that the **Age Group** with the greatest amount of sales revenue is sorted to the top.

© Critical Path Training. 2016. All Rights Reserved



Age Group	Count of CustomerId	Quantity	SalesAmount
Ages 50 TO 65	17803	1,301,255	$8,326,124.25
Ages 65 and over	17484	1,262,146	$8,161,248.51
Ages 40 TO 49	11373	799,420	$5,277,854.01
Ages 30 TO 39	10236	720,394	$4,814,349.01
Ages 18 TO 23	6787	468,830	$3,150,941.36
Total	**63683**	**4,552,045**	**$29,730,517.14**

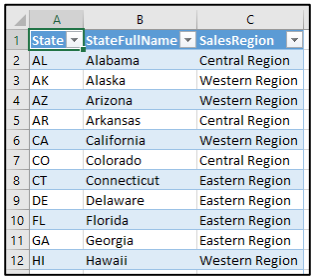
At this point, you might be bothered by the fact that some of the column header names in this visual are not a pretty as they could be. For example, it would be less confusing to business users if the column displaying a count of customers had a column heading of **Customer Count** instead of **Count of CustomerId**. You will address this issue later in the lab exercise where you begin to extend the data model by creating measures.

8. Save the work you have done by clicking the Save button in the upper left corner of the Power BI Desktop window.

Exercise 3: Extending the Data Model with a Lookup Table

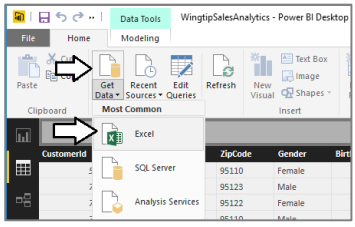
In this exercise, you will extend the data model by adding a new lookup table named **SalesRegions** which assigns a geographic sales region to each state. The work to accomplish this will involve importing data from an Excel workbook to create the **SalesRegions** table and also creating a relationship between the **SalesRegions** table and the **Customers** table. You will also create calculated columns in the **Customers** table to pull in related data from the **SalesRegions** table.

1. Locate the Excel workbook file which contains the **SalesRegions** table.
   a) Using Windows Explorer, open the folder at **c:\Student\Data** and locate the Excel workbook file named **SalesRegions.xlsx**.
   b) Double-click on named **SalesRegions.xlsx**. to open it inside Microsoft Excel 2016.
   c) Once the workbook file opens in Excel, take a moment to examine what's inside. You should be able to verify that there is a **SalesRegions** table which contains a row for each of the 50 states along with the state full name and an assigned sales region. As you can see from the following screenshot, the 50 states have been divided into three sales regions which are **Western Region**, **Central Region** and **Eastern Region**.

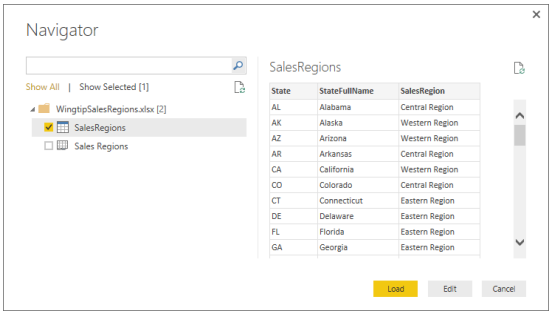


	A	B	C
1	State	StateFullName	SalesRegion
2	AL	Alabama	Central Region
3	AK	Alaska	Western Region
4	AZ	Arizona	Western Region
5	AR	Arkansas	Central Region
6	CA	California	Western Region
7	CO	Colorado	Central Region
8	CT	Connecticut	Eastern Region
9	DE	Delaware	Eastern Region
10	FL	Florida	Eastern Region
11	GA	Georgia	Eastern Region
12	HI	Hawaii	Western Region

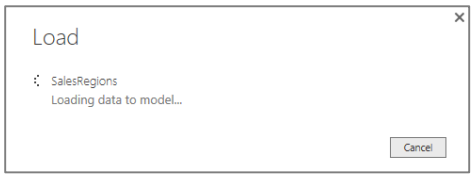
   d) Close Microsoft Excel and make sure you do not save any chnages to **SalesRegions.xlsx**.
2. Import the **SalesRegions** table into the data model of the current project.
   a) Navigate back to Power BI Desktop.
   b) Click on the **Home** tab on the ribbon.
   c) Drop down the **Get Data** menu in the ribbon and select the **Excel** command to display the **Navigator** dialog.



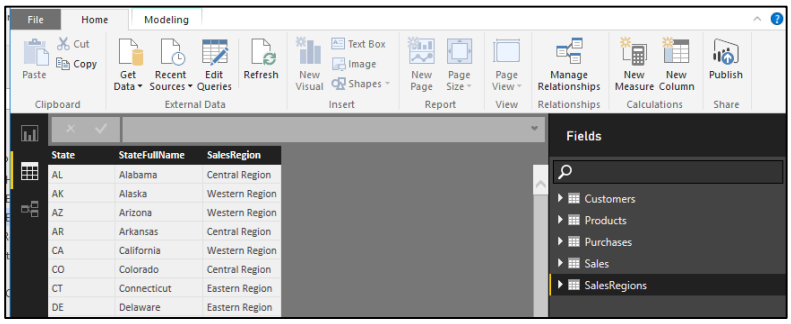
d) In the **Navigator** dialog, click the checkbox to select the **SalesRegion** table. Make sure to select the top checkbox for the **SalesRegion** table and not the bottom checkbox for the **Sales Region** worksheet. Once you have selected the **SalesRegion** table, you should be able to see a sample of its rows in the **Navigator** dialog as shown in the following screenshot.



e) Click the **Load** button to load the data and to create the **SalesRegions** table. Power BI Desktop will display the **Load** dialog until the import process completed.



f) Once the import process completes, you should be able to see **SalesRegions** table in data view.



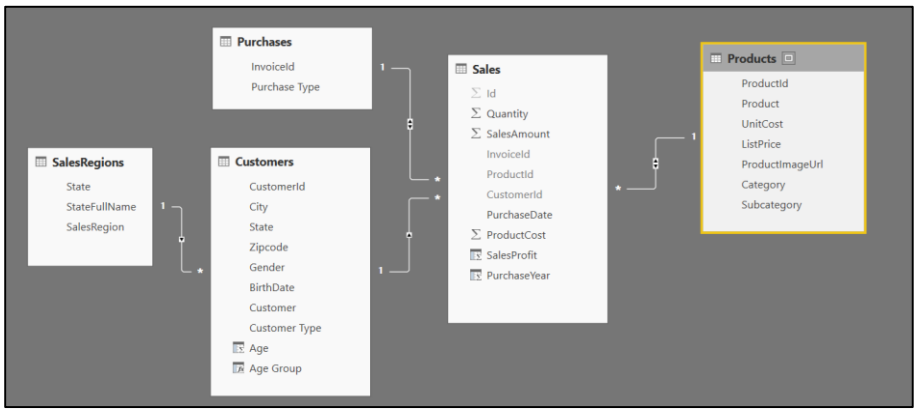
3. Examine the relationship that has been created between the **Customers** table and the **SalesRegions** table.

a) Navigate to relationship view.

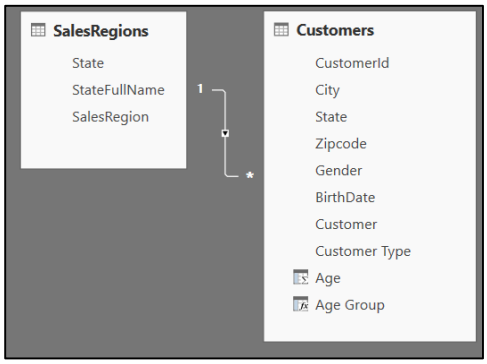
b) You should be able to see **SalesRegions** table.

c) Rearrange the table layout to match the following screenshot.

© Critical Path Training. 2016. All Rights Reserved



d) You should be able to see that a relationship has already been created between the **SalesRegions** table and the **Customers** because both tables contain a similar column named **State**.

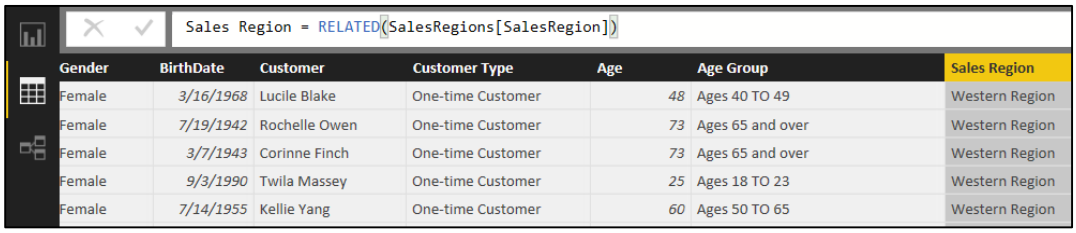


Since there is a relationship between the **SalesRegions** table and the **Customers** table, you can begin to use the **RELATED** function provided by DAX. More specifically, you can use the **RELATED** function to create calculated columns in the **Customers** table that are written to pull in data from the **SalesRegions** table.

4. Add a calculated column to the **Customers** table named **Sales Region** to display the sales region for each state.
   a) Navigate to data view.
   b) Select the **Customers** table in the **Fields** list.
   c) Create a new calculated column by clicking the **New Column** button in the ribbon.
   d) Enter to following DAX expression into the formula bar to create the calculated column named **Sales Region**.

```
Sales Region = RELATED(SalesRegions[SalesRegion])
```

   e) Press the **ENTER** key to add the calculated column to the table. You should be able to see a value in the **Sales Region** column for each row in the **Customers** table.



Gender	BirthDate	Customer	Customer Type	Age	Age Group	Sales Region
Female	3/16/1968	Lucile Blake	One-time Customer	48	Ages 40 TO 49	Western Region
Female	7/19/1942	Rochelle Owen	One-time Customer	73	Ages 65 and over	Western Region
Female	3/7/1943	Corinne Finch	One-time Customer	73	Ages 65 and over	Western Region
Female	9/3/1990	Twila Massey	One-time Customer	25	Ages 18 TO 23	Western Region
Female	7/14/1955	Kellie Yang	One-time Customer	60	Ages 50 TO 65	Western Region

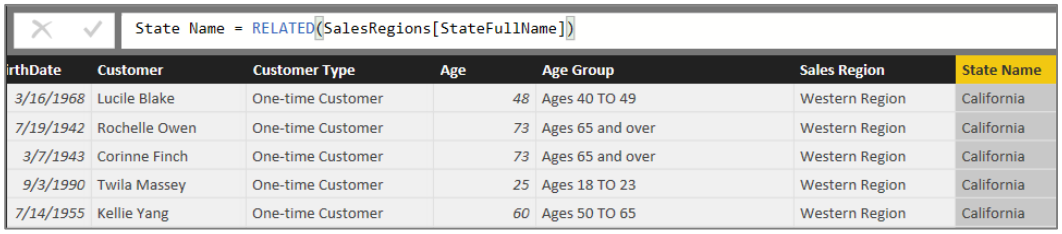
5. Add a calculated column to the **Customers** table named **State Name** to display the full state name.
   a) Create a new calculated column in the **Customers** table by clicking the **New Column** button in the ribbon.

© Critical Path Training. 2016. All Rights Reserved

b) Enter to following DAX expression into the formula bar to create the calculated column named **State Name**.

```
State Name = RELATED(SalesRegions[StateFullName])
```

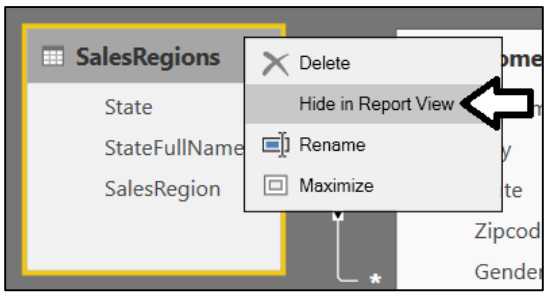
c) Press the **ENTER** key to add the calculated column to the table. You should be able to see a value in the **Sales Region** column for each row in the **Customers** table.



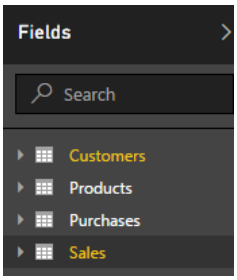
irthDate	Customer	Customer Type	Age	Age Group	Sales Region	State Name
3/16/1968	Lucile Blake	One-time Customer	48	Ages 40 TO 49	Western Region	California
7/19/1942	Rochelle Owen	One-time Customer	73	Ages 65 and over	Western Region	California
3/7/1943	Corinne Finch	One-time Customer	73	Ages 65 and over	Western Region	California
9/3/1990	Twila Massey	One-time Customer	25	Ages 18 TO 23	Western Region	California
7/14/1955	Kellie Yang	One-time Customer	60	Ages 50 TO 65	Western Region	California

Now that you have pulled all the important data from the **SalesRegions** table into the **Customers** table, there is no need to display the **SalesRegions** table in report view. In the next step, you will hide the **SaleRegions** table to simplify view of the data model that is shown in report view.

6. Hide the **SalesRegions** table from report view.
   a) Navigate to relationship view.
   b) Right-click on the the **SalesRegions** table

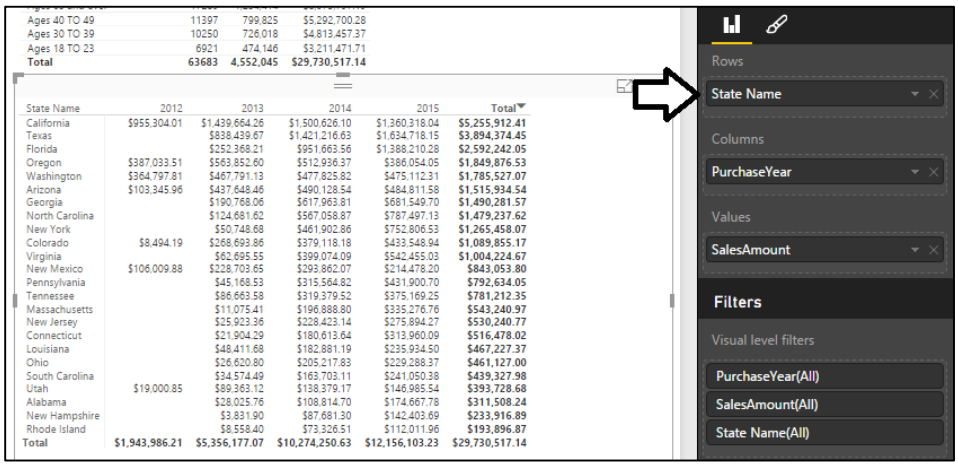


   c) Navigate to report view and verify that the **SalesRegions** table is not displayed as one of the tables in the **Fields** list.

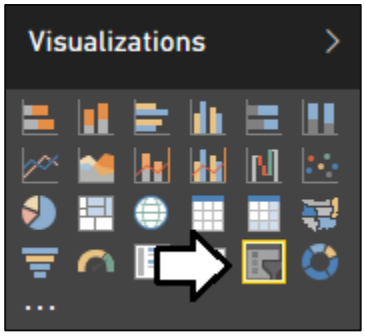


7. Update the matrix with row labels based on the **State** column to use the **State Name** column instead.
   a) Navigate back to report view.
   b) Select the matrix visual with the row labels based on the **State** column.
   c) Remove the **State** field from the **Rows** well in the **Visualization** pane and replace it with the **State Name** column. The visual should now show the full state name instead of the two-character state abbreviation.

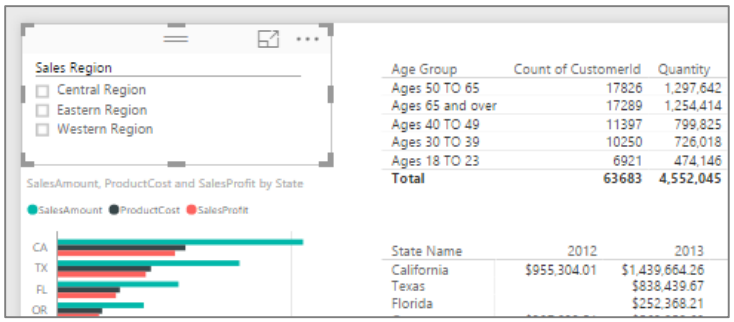
© Critical Path Training. 2016. All Rights Reserved



8. Add a slicer visual to the report to filter all displayed results by sales region.
   a) Navigate to report view if you are not already there.
   b) Make sure that no visuals are selected on the page so that you can create a new visual.
   c) Select the checkbox for the **Sales Region** column in the **Fields** list to create a new visual.
   d) Click the **Slicer** button in the **Visualizations** list to change the visual type to a slicer.

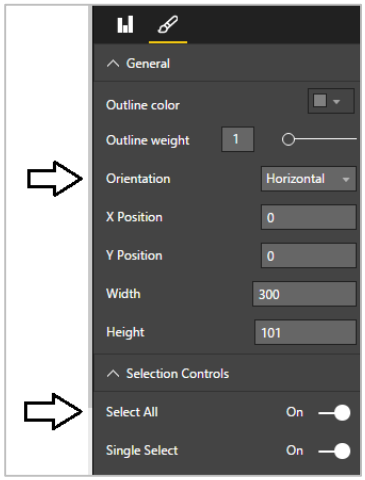


   e) Using the mouse, reposition the slicer visual so it appears in the upper, left-hand corner of the page as shown in the following screenshot.



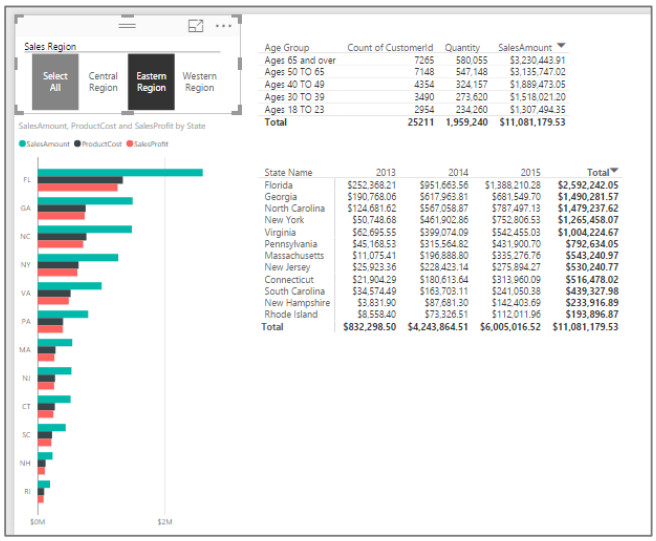
   f) Change the alignment of the slicer visual from vertical to horizontal. Accomplishing this by first clicking the Edit Pen button in the **Visualization** pane to view the appearance properties of the visual and then by modifying the **Orientation** property to a value of **Horizontal**. Also enable the **Select All** property by setting its value to **On**.

© Critical Path Training. 2016. All Rights Reserved



g) The slicer visual should now appear with a horizontal layout with an additional **Select All** node.

h) Click on the different nodes of the slicer visual to see its filtering effect. If you click on the **Eastern Region** node, all the other visuals apply a filter to only shows states assigned to the Eastern sales region.



9. Save the work you have done by clicking the Save button in the upper left corner of the Power BI Desktop window.

Exercise 4: Extending the Data Model by Creating Measures

In this exercise you will create four measures named **Units Sold, Sales Revenue, Product Cost** and **Profit** that will perform sum aggregations on the **Sales** table. You will also create a measure named **Customer Count** that performs a distinct count aggregation on the **Customers** table. As you will see, creating measures to use in your report visuals will give you much greater control over the column names, formatting and aggregations that are displayed in your reports.

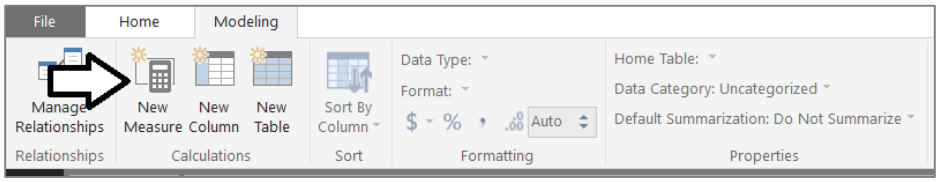
1. Create a measure named **Units Sold** to perform a sum aggregation on the **Quantity** column of the **Sales** table.

© Critical Path Training. 2016. All Rights Reserved

a) Navigate to data view.

b) Select the **Sales** table from the **Fields** list.

c) Create a new measure by clicking the **New Measure** button in the ribbon.

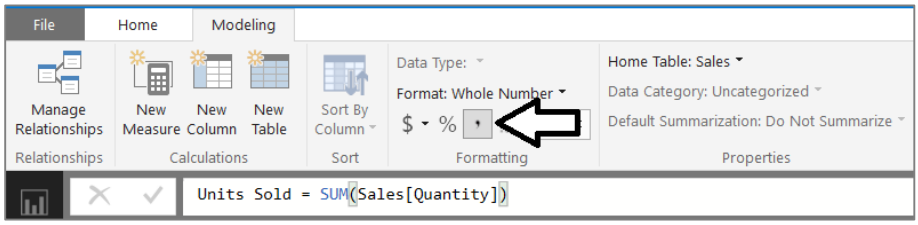


d) Enter to following DAX expression into the formula bar to create the measure named **Units Sold**.

```
Units Sold = SUM(Sales[Quantity])
```

e) Press the **ENTER** key to add the measure to data model.

f) Modify the formatting by clicking and selecting the Comma button on the ribbon to add a comma separator.



2. Create a measure named **Sales Revenue** to perform a sum aggregation on the **SalesAmount** column of the **Sales** table.

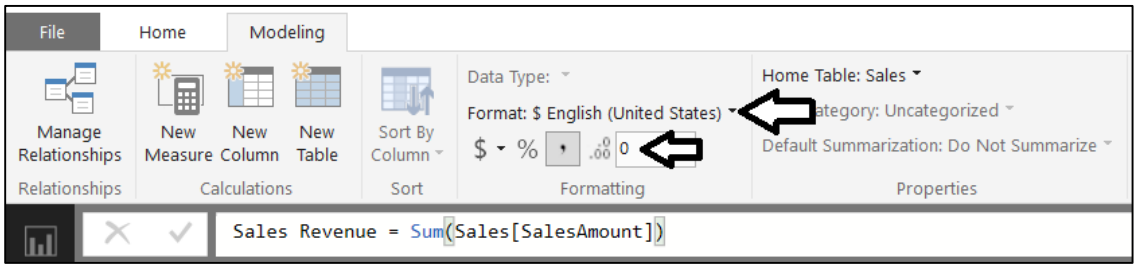
a) Create a new measure by clicking the **New Measure** button in the ribbon.

b) Enter to following DAX expression into the formula bar to create the measure named **Sales Revenue**.

```
Sales Revenue = Sum(Sales[SalesAmount])
```

c) Press the **ENTER** key to add the measure to data model.

d) Modify the formatting by dropping down the **Format** menu on the ribbon and selecting **Currency > $ English (United States)**. Also use the spinner control below the **Format** menu to set the number of decimal places shown to zero.



3. Create a measure named **Product Cost** to perform a sum aggregation on the **ProductCost** column of the **Sales** table.

a) Create a new measure by clicking the **New Measure** button in the ribbon.

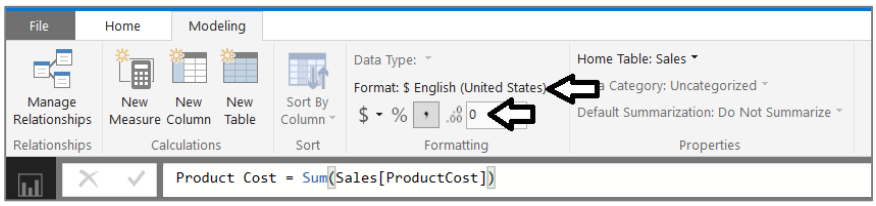
b) Enter to following DAX expression into the formula bar to create the measure named **Product Cost**.

```
Product Cost = Sum(Sales[ProductCost])
```

c) Press the **ENTER** key to add the measure to data model.

d) Modify the formatting by dropping down the **Format** menu on the ribbon and selecting **Currency > English (United States)**. Also use the spinner control below the **Format** menu to set the number of decimal places shown to zero.

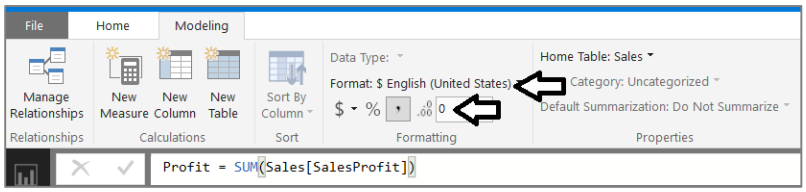
© Critical Path Training. 2016. All Rights Reserved



4. Create a measure named **Profit** to perform a sum aggregation on the **SalesProfit** column of the **Sales** table.
   a) Create a new measure by clicking the **New Measure** button in the ribbon.
   b) Enter to following DAX expression into the formula bar to create the measure named **Profit**.

```
Profit = SUM(Sales[SalesProfit])
```

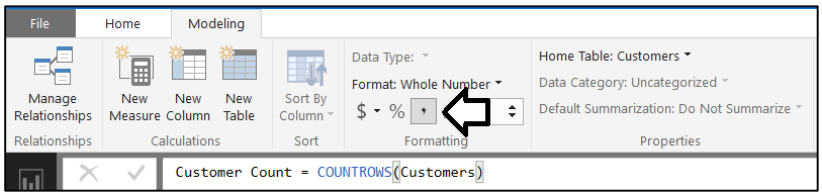
   c) Press the **ENTER** key to add the measure to data model.
   d) Modify the formatting by dropping down the **Format** menu on the ribbon and selecting **Currency > English (United States)**. Also use the spinner control below the **Format** menu to set the number of decimal places shown to zero.



5. Create a measure named **Customer Count** to perform an aggregation to count the number of rows in the **Customers** table.
   a) Make sure the **Customers** table is selected.
   b) Create a new measure by clicking the **New Measure** button in the ribbon.
   c) Enter the following DAX expression into the formula bar to create the measure named **Customer Count**.

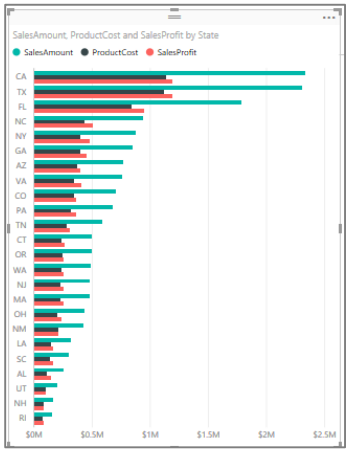
```
Customer Count = COUNTROWS(Customers)
```

   d) Press the **ENTER** key to add the measure to data model.
   e) Modify the formatting by clicking and selecting the Comma button on the ribbon to add a comma separator.

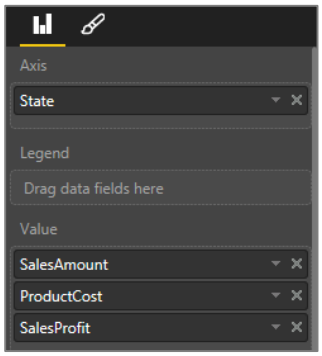


6. Update the visuals in the report to use the new measures you have just created.
   a) Navigate to report view.
   b) Select the clustered bar chart visual.

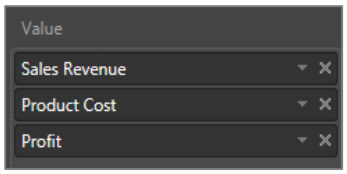
© Critical Path Training. 2016. All Rights Reserved



c) Examine the properties of this visual in the **Visualizations** pane. The **Value** well should currently contain the three fields named **SalesAmount**, **ProductCost** and **SalesProfit**.

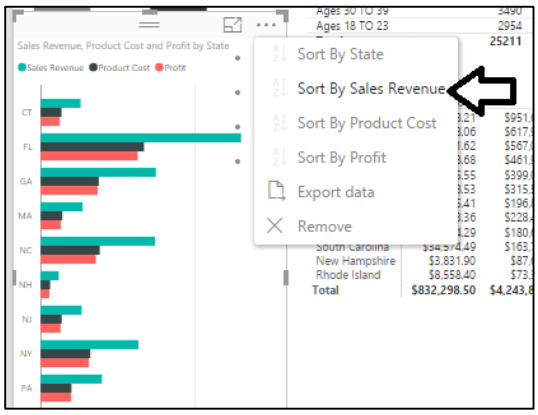


d) Remove those three fields from the **Value** well and replace them with the new measures named **Sales Revenue**, **Product Cost** and **Profit**.

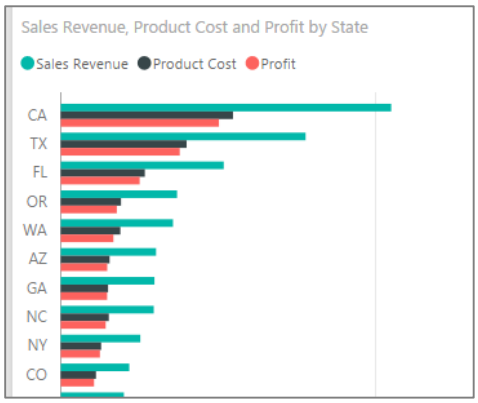


e) You will notice that the sort order of this visual is now based on the **State** field because you removed the **SalesAmount** field that was being used for sorting. Resort the rows in the visual using **Sales Revenue**.

© Critical Path Training. 2016. All Rights Reserved

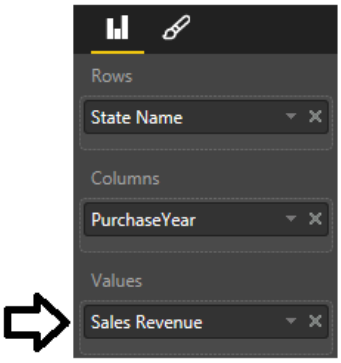


f) Now this visual looks more polished because it is using measures instead of aggregated columns to produce its values.



g) Select the matrix visual which displays sales revenue by state and year.

h) In the **Visualizations** pane, remove the **SalesAmount** column from the **Values** well and replace it with **Sales Revenue** measure.



i) The matrix visual should display its numbers with currency formatting with no decimal places.

j) Click the **Total** column header to sort the state rows by total sales revenue.

© Critical Path Training. 2016. All Rights Reserved

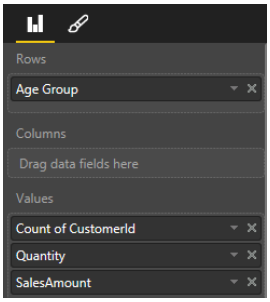
State Name	2012	2013	2014	2015	Total
California	$955,304	$1,439,664	$1,500,626	$1,360,318	**$5,255,912**
Texas		$838,440	$1,421,217	$1,634,718	**$3,894,374**
Florida		$252,368	$951,664	$1,388,210	**$2,592,242**
Oregon	$387,034	$563,853	$512,936	$386,054	**$1,849,877**
Washington	$364,798	$467,791	$477,826	$475,112	**$1,785,527**
Arizona	$103,346	$437,648	$490,129	$484,812	**$1,515,935**
Georgia		$190,768	$617,964	$681,550	**$1,490,282**
North Carolina		$124,682	$567,059	$787,497	**$1,479,238**
New York		$50,749	$461,903	$752,807	**$1,265,458**

k) Select the matrix visual which displays a row for each age group.

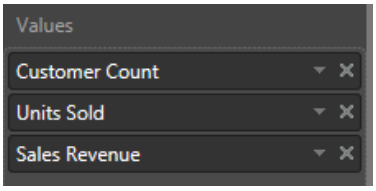
l) Notice how some of the columns contain text (e.g. **Count of CustomerId**) that is not very user-friendly.

Age Group	Count of CustomerId	Quantity	SalesAmount
Ages 50 TO 65	17826	1,297,642	$8,337,090.59
Ages 65 and over	17289	1,254,414	$8,075,797.19
Ages 40 TO 49	11397	799,825	$5,292,700.28
Ages 30 TO 39	10250	726,018	$4,813,457.37
Ages 18 TO 23	6921	474,146	$3,211,471.71
Total	**63683**	**4,552,045**	**$29,730,517.14**

m) Examine the properties of this visual in the **Visualizations** pane. The **Value** well should currently contain the three fields named **Count of CustomerId**, **Quantity** and **SalesAmount**.



n) Remove those three fields from the **Value** well and replace them with the new measures named **Customer Count**, **Units Sold** and **Sales Revenue**.



o) The columns of the table visual are now more user-friendly and the formatting of values looks better as well.

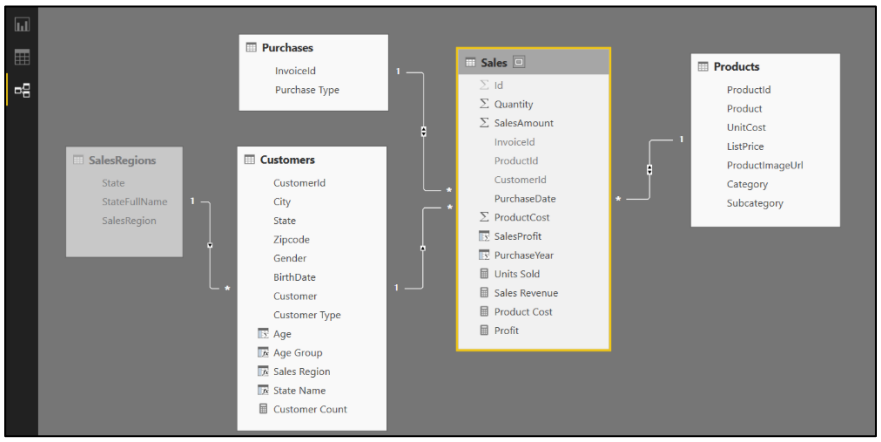
Age Group	Customer Count	Units Sold	Sales Revenue
Ages 18 TO 23	6,921	474,146	$3,211,472
Ages 30 TO 39	10,250	726,018	$4,813,457
Ages 40 TO 49	11,397	799,825	$5,292,700
Ages 50 TO 65	17,826	1,297,642	$8,337,091
Ages 65 and over	17,289	1,254,414	$8,075,797
Total	**63,683**	**4,552,045**	**$29,730,517**

p) Click on the **Sales Revenue** column header to sort the age groups by sales revenue.

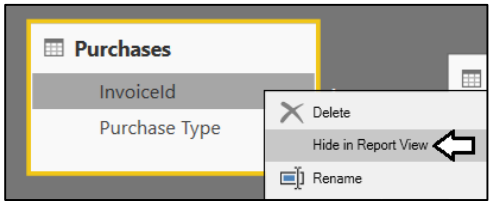
© Critical Path Training. 2016. All Rights Reserved

Age Group	Customer Count	Units Sold	Sales Revenue
Ages 50 TO 65	17,826	1,297,642	$8,337,091
Ages 65 and over	17,289	1,254,414	$8,075,797
Ages 40 TO 49	11,397	799,825	$5,292,700
Ages 30 TO 39	10,250	726,018	$4,813,457
Ages 18 TO 23	6,921	474,146	$3,211,472
Total	**63,683**	**4,552,045**	**$29,730,517**

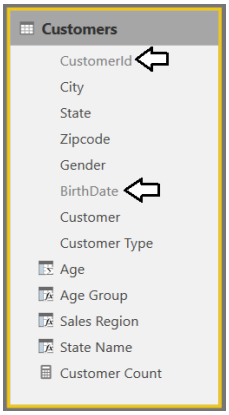
7. You will complete your work in this exercise by cleaning up the data model by hiding fields in report view.
   a) Navigate to Relationship View so you can see all the tables in the project's data model.
   b) Resize each table so that is properly displays all its fields.



   c) In the **Purchases** table, hide the **InvoiceId** column by right-clicking the column and selecting **Hide in Report View**.

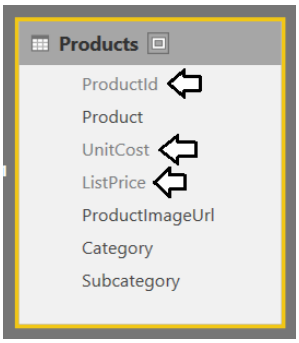


   d) Use the same technique to hide the **CustomerId** field and the **BirthDate** field from the **Customers** table.

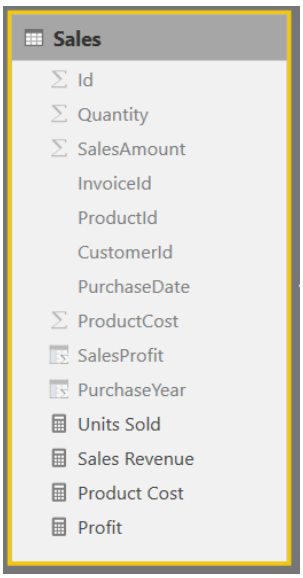


   e) Use the same technique to hide the **ProductId** field, the **UnitCost**, field and the **ListPrice** field from the **Products** table.

© Critical Path Training. 2016. All Rights Reserved



f) In the **Sales** table, hide every field except for the 4 measures named **Units Sold**, **Sales Revenue**, **Product Cost** and **Profit**.

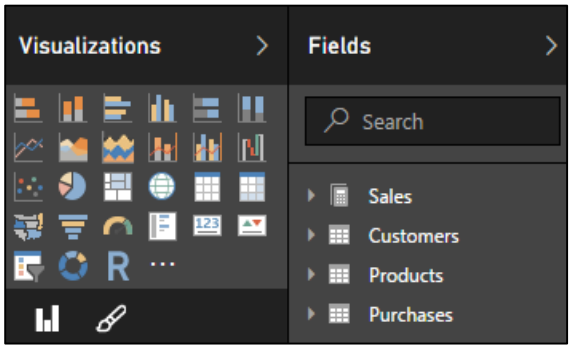


When you hide all the fields in a table except for its measures, Power BI Desktop begins to treat this table as a **fact table**. The main effect this has in Power BI Desktop is that the table will be displayed above in the Fields list when in report view. However, Power BI Desktop has an unfortunate bug where it will not recognize a table as a fact table until you close and reopen the project. In the next step, you will close Power BI Desktop. Then you will restart Power BI Desktop and reopen the current project. This will allow you to see how Power BI Desktop displays measure-only tables in Report View.

8. Save the current project by clicking the **Save** button in the ribbon.
9. Close and restart Power BI Desktop to see the effects of a measure-only table.
   a) Close Power BI Desktop.
   b) Restart Power BI Desktop
   c) Reopen the Power BI Desktop project file named **c:\Student\Projects\Wingtip Sales Analysis.pbix**.
10. Examine the tables of the project's data model in Report View.
   a) Navigate to Report view.

© Critical Path Training. 2016. All Rights Reserved

b) Examine the tables shown in **Fields** list on the right side of the Power BI Desktop application window. You can see that the **Sales** table is displayed first with a calculator icon. The three other tables shown below are displayed with a standard table icon.

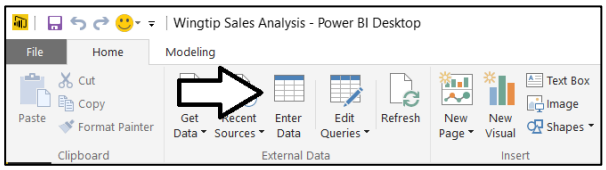


You have to agree that these changes to the data model make it easier to use when in report view.

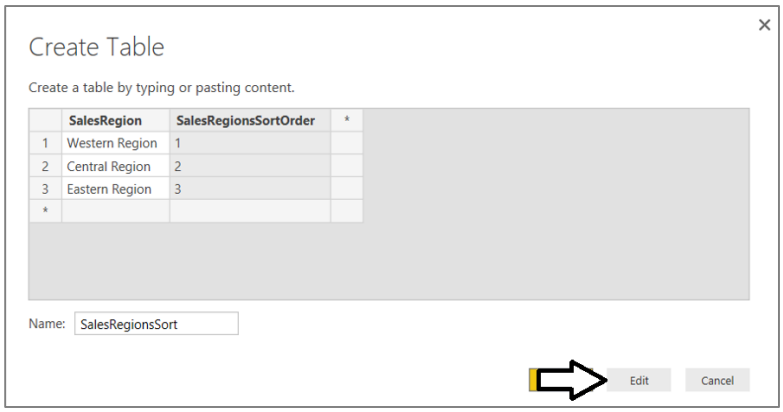
Exercise 5: Extending the Data Model with Geolocation Metadata

In this exercise you will begin by adding explicit support to sort sales regions so that the Western Region is first followed by Central Region and then the Eastern Region. After that, you will configure geolocation metadata to fields in the **Customers** table including **State**, **City** and **Zipcode**. After that, you will create a new report page with a map visual to visualize how customer sales data is spread out across various geographic regions within the United States.

1. Create a new query to named **SalesRegionsSort** which returns data to sort by sales regions in a custom sort order.
   a) Navigate to the **Home** tab in the ribbon.
   b) Click the **Enter Data** button.



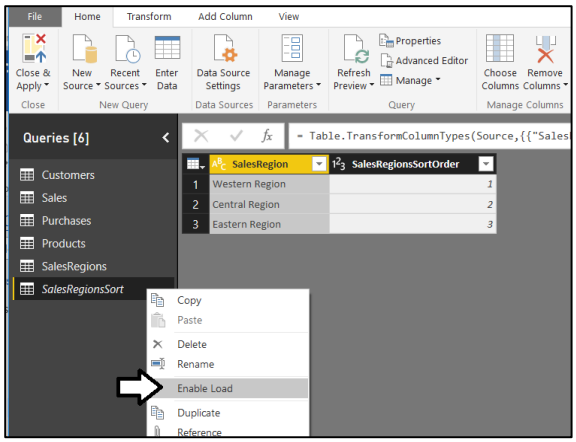
   c) In the **Create Table** dialog, add two columns named **SalesRegion** and **SalesRegionSortOrder**.
   d) Add a row with a **SalesRegion** value of **Western Region** and a **SalesRegionSortOrder** value of **1**.
   e) Add a row with a **SalesRegion** value of **Central Region** and a **SalesRegionSortOrder** value of **2**.
   f) Add a row with a **SalesRegion** value of **Eastern Region** and a **SalesRegionSortOrder** value of **3**.
   g) Give the new table of name **SalesRegionSort** and then click **Edit** to open the Query Editor window.



© Critical Path Training. 2016. All Rights Reserved

h) You should now see the **SalesRegionSort** table in the Query Editor window.

i) In the left navigation, right click on the **SalesRegionsSort** table and then select **EnableLoad** to disable loading the query result into the project's data model.

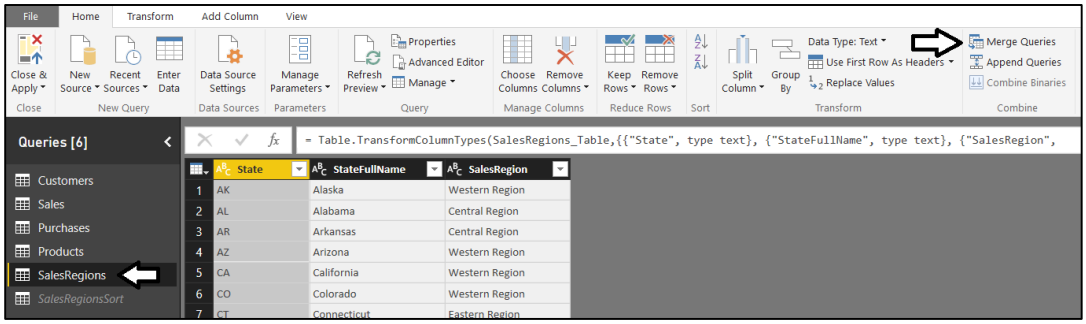


2. Merge the **SalesRegions** query with the **SalesRegionSort** query to add the **SalesRegionSortOrder** column.

a) Select the **SalesRegion** table in the left navigation of the Query Editor window.

b) Navigate to the **Home** tab in the ribbon.

c) Click the **Merge Query** button in the top right corner of the ribbon to open the **Merge** dialog.

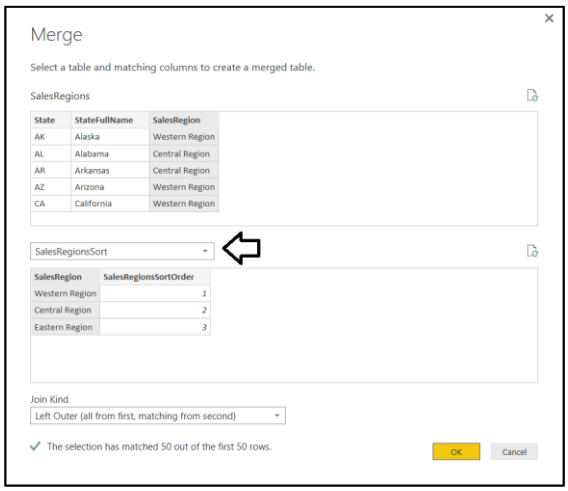


d) In the **Merge** dialog, select **SalesRegionSort** from the dropdown menu.

e) Select the **SalesRegion** column for both the **SalesRegions** query results and the **SalesRegionsSort** query results.

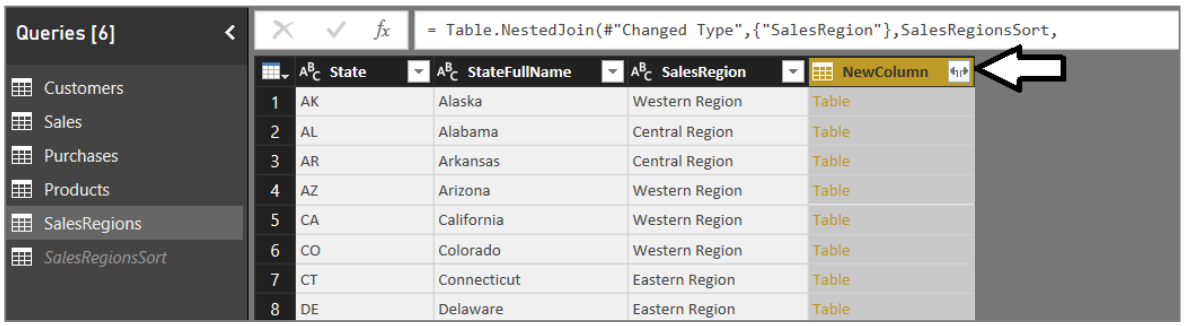
f) Click **OK**.to complete the merge operation.

© Critical Path Training. 2016. All Rights Reserved



Now you should see a new column named **NewColumn** in the results of the **SalesRegion** query.

g) Click the button on the right side of the column header for the **NewColumn** column.

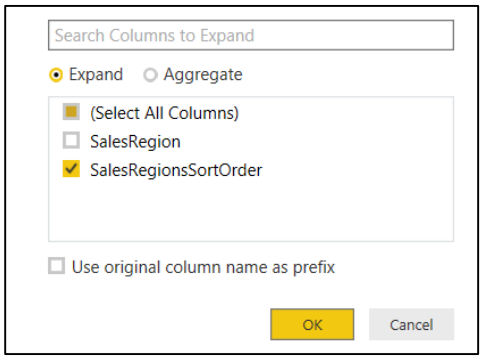


h) In the next dialog, select the **Expand** option.

i) Select only the one column named **SalesRegionSortOrder**.

j) Uncheck the checkbox with the caption **Use original column name as prefix**.

k) Click **OK** to complete the expand operation.

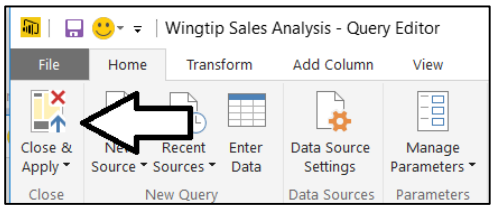


l) You should be able to verify that the results of the **SalesRegions** query now contain the **SalesRegionSortOrder** column.

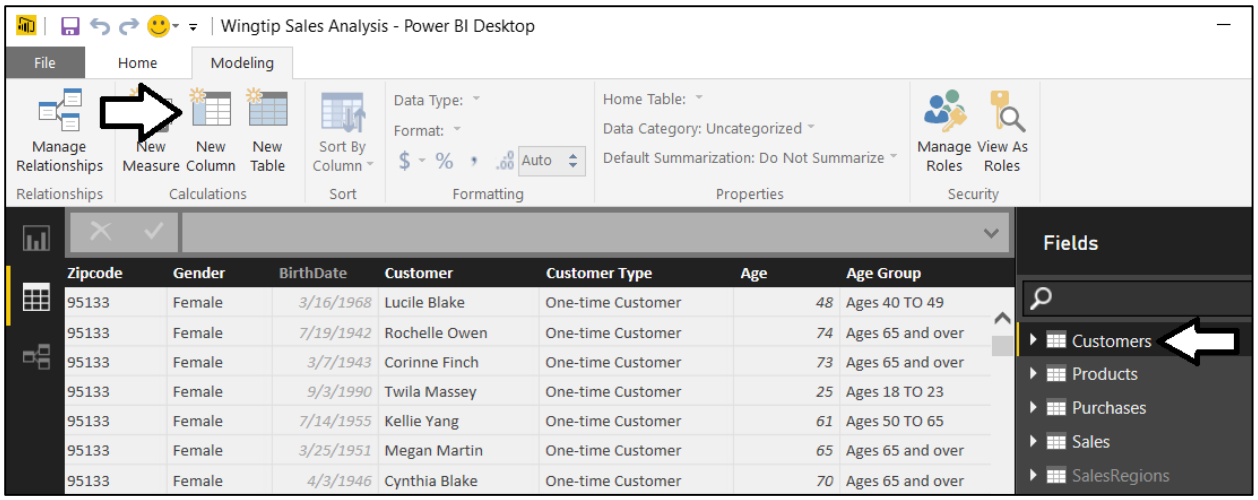
© Critical Path Training. 2016. All Rights Reserved

	State	StateFullName	SalesRegion	SalesRegionsSortOrder
1	AK	Alaska	Western Region	1
2	AZ	Arizona	Western Region	1
3	AL	Alabama	Central Region	2
4	AR	Arkansas	Central Region	2
5	CA	California	Western Region	1
6	CO	Colorado	Western Region	1
7	CT	Connecticut	Eastern Region	3
8	DE	Delaware	Eastern Region	3

m) Click the **Close and Apply** button to execute the **SalesRegions** query which will update the data in the project's data model.



3. Add the **SalesRegionSortOrder** column to the **Customers** table as a calculated field.
   a) Navigate to Data view mode in the Power BI Desktop application window.
   b) Select the **Customers** table in the **Fields** list.
   c) Click the **New Column** button in the ribbon to create a new calculate column.



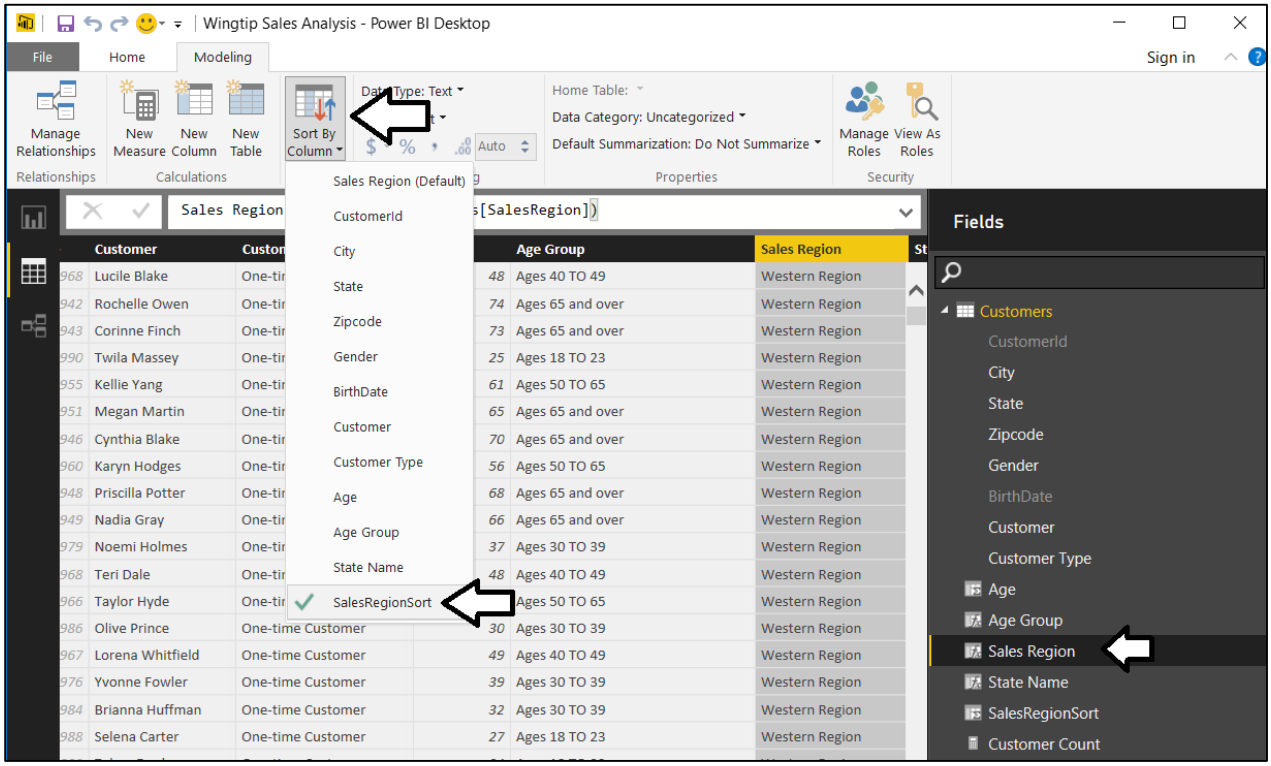
Zipcode	Gender	BirthDate	Customer	Customer Type	Age	Age Group
95133	Female	3/16/1968	Lucile Blake	One-time Customer	48	Ages 40 TO 49
95133	Female	7/19/1942	Rochelle Owen	One-time Customer	74	Ages 65 and over
95133	Female	3/7/1943	Corinne Finch	One-time Customer	73	Ages 65 and over
95133	Female	9/3/1990	Twila Massey	One-time Customer	25	Ages 18 TO 23
95133	Female	7/14/1955	Kellie Yang	One-time Customer	61	Ages 50 TO 65
95133	Female	3/25/1951	Megan Martin	One-time Customer	65	Ages 65 and over
95133	Female	4/3/1946	Cynthia Blake	One-time Customer	70	Ages 65 and over

   d) Type in the following DAX expression to create a new calculated column named **SalesRegionSort**.

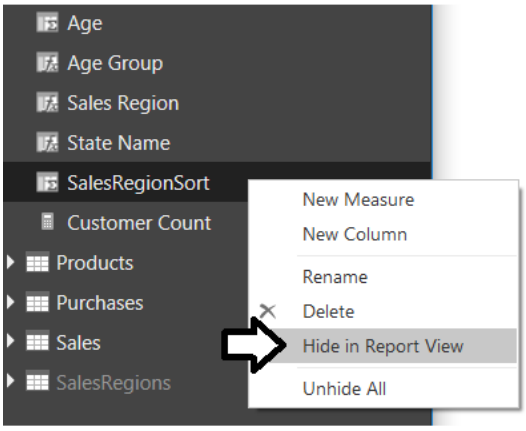
```
SalesRegionSort = RELATED(SalesRegions[SalesRegionsSortOrder])
```

4. Configure a custom sort column for the **Sales Region** field.
   a) Select the **Sales Region** field from the **Fields** list.
   b) Drop by the **Sort By Column** menu in the ribbon and select the **SalesRegionSort** column.

© Critical Path Training. 2016. All Rights Reserved

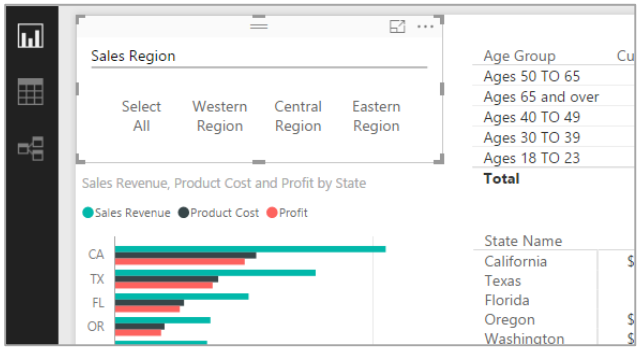


5. In the **Fields** list, right click the **SalesRegionSort** column and select **Hide in Report View**.

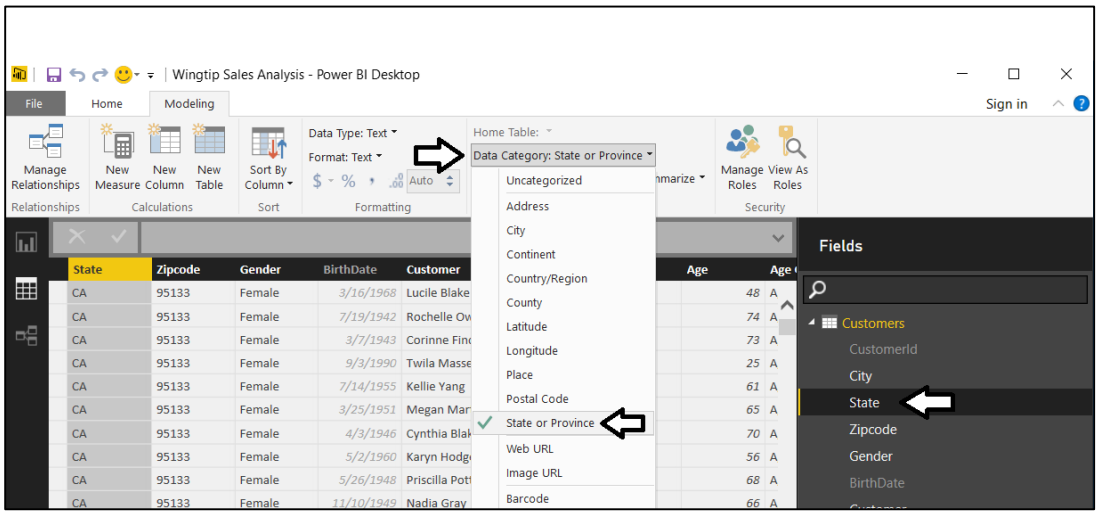


6. Return to Report View and examine the slicer which shows the sales regions. They should now be sorted in a custom sort order with **Western Sales** first followed by **Central Region** and then **Eastern Region**.

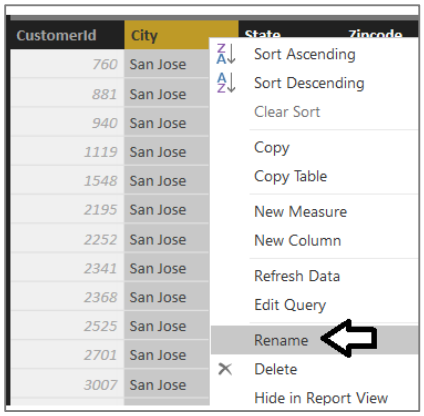
© Critical Path Training. 2016. All Rights Reserved



7. Configure Geolocation Metadata for the **State** field in the **Customers** table.
   a) Return to Data View.
   b) From the **Fields** list on the right, select the **State** field from the **Customers** table.
   c) Drop down the **Data Category** menu from the ribbon and select **State or Province**.



8. Modify the **City** column to contain the state as well as the city to remove ambiguity when determining geographic locations.
   a) Select the **Customers** table in the **Fields** list.
   b) Right-click on the column header for the **City** column and select the **Rename** command.



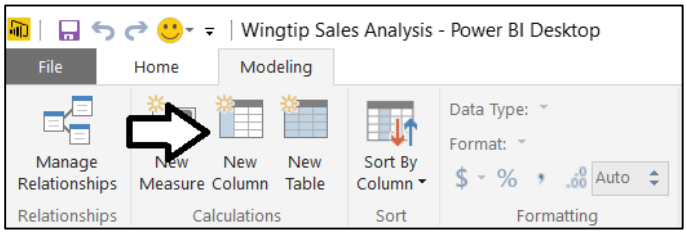
   c) Rename the column to **City Name**.

© Critical Path Training. 2016. All Rights Reserved

CustomerId	City Name	State	Zipcode
760	San Jose	CA	95133
881	San Jose	CA	95133
940	San Jose	CA	95133
1119	San Jose	CA	95133
1548	San Jose	CA	95133

d) Navigate to the **Modeling** tab in the ribbon.

e) Click the **New Column** tab to create a new calculated column.



f) Add in the following DAX expression to create a new column named **City**.

```
City = [City Name] & ", " & [State]
```

g) The **City** column should display the city and the state which will remove ambiguity when used as a geolocation field.

```
City = [City Name] & ", " & [State]
```

	Age Group	Sales Region	State Name	SalesRegionSort	City
48	Ages 40 TO 49	Western Region	California	1	San Jose, CA
74	Ages 65 and over	Western Region	California	1	San Jose, CA
73	Ages 65 and over	Western Region	California	1	San Jose, CA
25	Ages 18 TO 23	Western Region	California	1	San Jose, CA
61	Ages 50 TO 65	Western Region	California	1	San Jose, CA
65	Ages 65 and over	Western Region	California	1	San Jose, CA

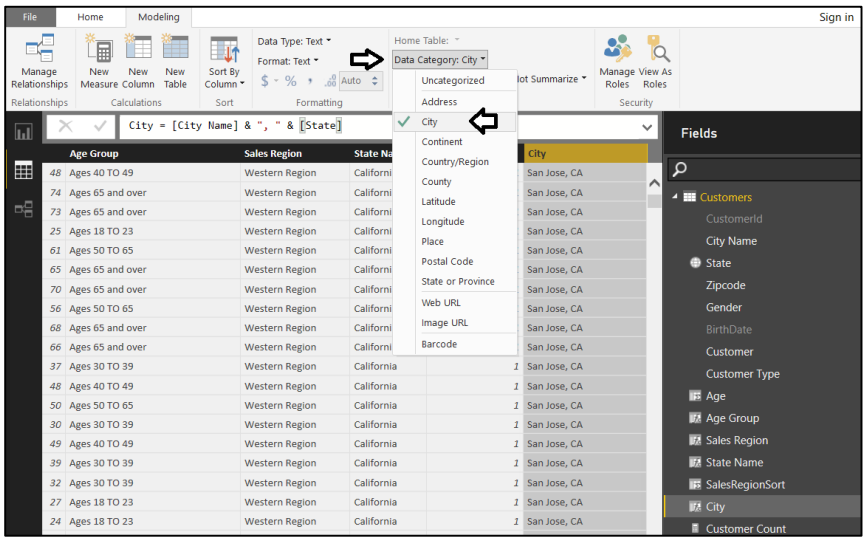
9. Configure geolocation metadata for the **City** field in the **Customers** table.

a) Return to Data View.

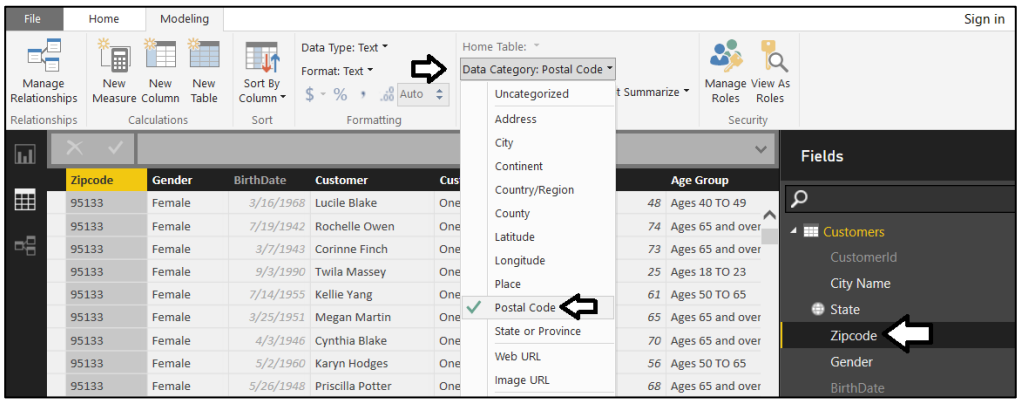
b) From the **Fields** list on the right, select the **City** field from the **Customers** table.

c) Drop down the **Data Category** menu from the ribbon and select **City**.

© Critical Path Training. 2016. All Rights Reserved

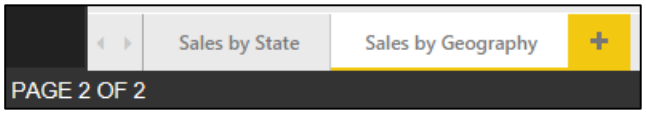


10. Configure geolocation metadata for the **Zipcode** field in the **Customers** table.
    a) Return to Data View.
    b) From the **Fields** list on the right, select the **Zipcode** field from the **Customers** table.
    c) Drop down the **Data Category** menu from the ribbon and select **Postal Code**.



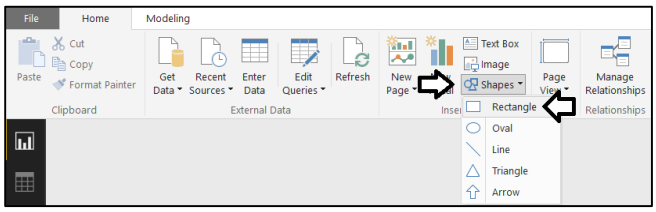
Now that you have configured fields in the **Customers** table with geolocation metadata, it's time to put this metadata to use by creating a new report page with a map visual to visualize how sales revenue is distributed over geographic regions.

11. Create a new report page.
    a) Return to Report View.
    b) Click on the (**+**) button on the page navigation tab to create a new page.
    c) Rename the new page to **Sales by Geography**.

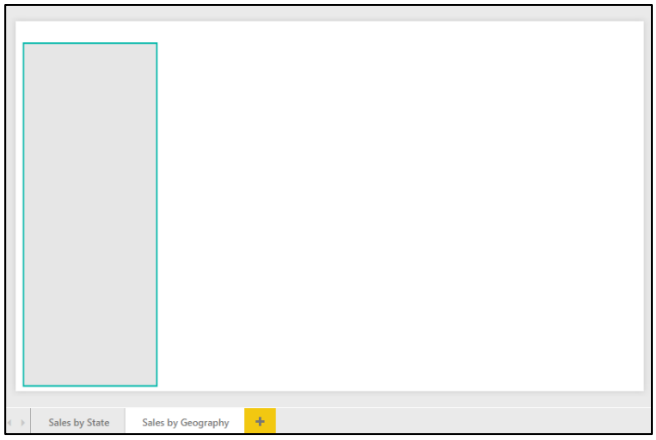


12. Add a background **Rectangle** shape to the report for formatting purposes.
    a) Navigate to the **Home** tab in the ribbon.
    b) Drop down the Shapes menu in the ribbon and select Rectangle to add a new rectangle shape to the report.

© Critical Path Training. 2016. All Rights Reserved

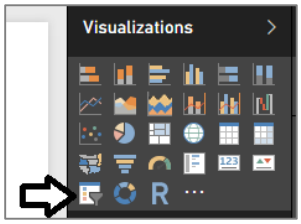


c) Using the mouse, reposition the rectangle shape so it takes up the full height of the report page and about 20% of the width as shown in the following screenshot.



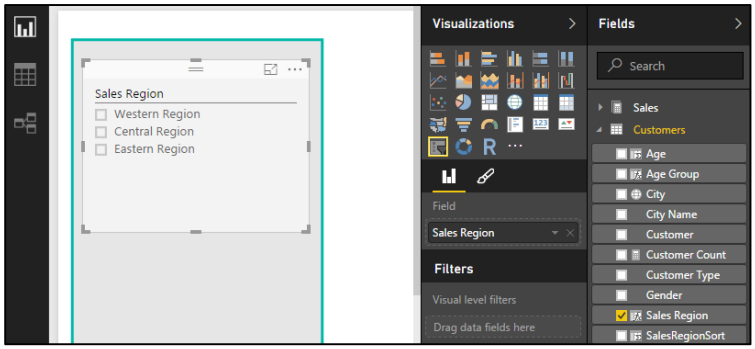
13. Create a new slicer visual to filter by sales region.

a) Click the **Slicer** button in the **Visualizations** list to add a new slicer visual to the report.



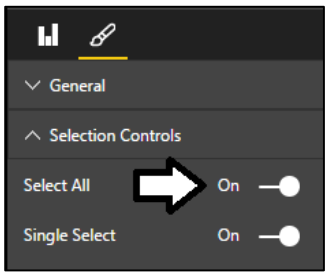
b) Add the **Sales Region** field from the **Customers** table into the **Field** well for the slicer.

c) Reposition the slicer so it sits on top of the rectangle shape as shown in the following screenshot.

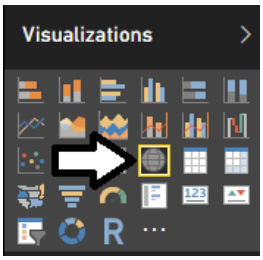


d) With the slicer selected, navigate to the **Selection Controls** sections in the **Format** properties pane and set **Select All** to **On**.

© Critical Path Training. 2016. All Rights Reserved

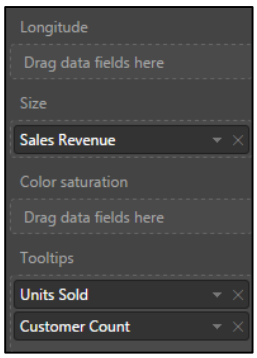


14. Add a new Map visual to display sales revenue by state.
    a) Click the New Visual button on the ribbon to create a new visual.
    b) Click the **Map** button in the **Visualizations** list to change the visual into a Map visual.



    c) Configure the fields for the **Map** visual by adding the **State** field from the **Customers** table into the **Location** well and the **Sales Revenue** field from the **Sales** table into the **Size** well.

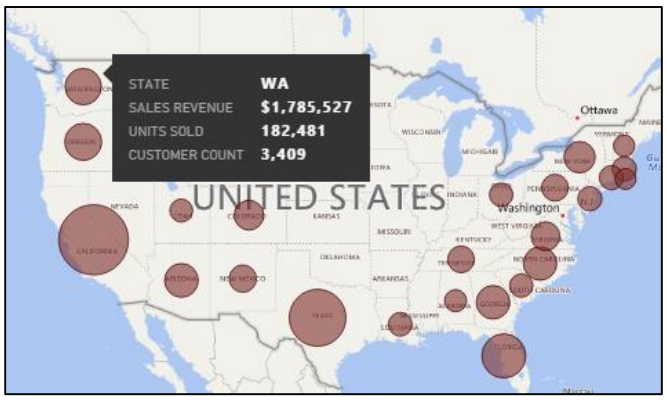
    d) Add the **Units Sold** field from the **Sales** table and the **Customer Count** field from the **Customers** table into the **Tooltips** well.



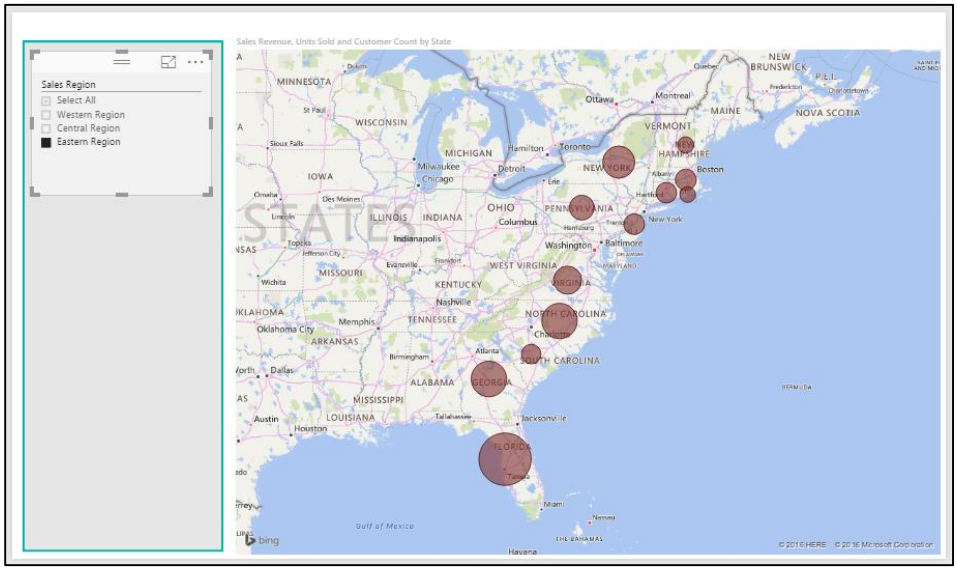
© Critical Path Training. 2016. All Rights Reserved

e) With the Map visual selected, navigate to the **Data Colors** section in the **Format** properties pane and change the default color to a dark red to it stands out more clearly on the map visual.

f) Test out the **Tooltips** setting by hovering the mouse over a red dot for a specific state. You should see that the **Tooltips** setting displays **State**, **Sales Revenue**, **Units Sold** and **Customer Count**.

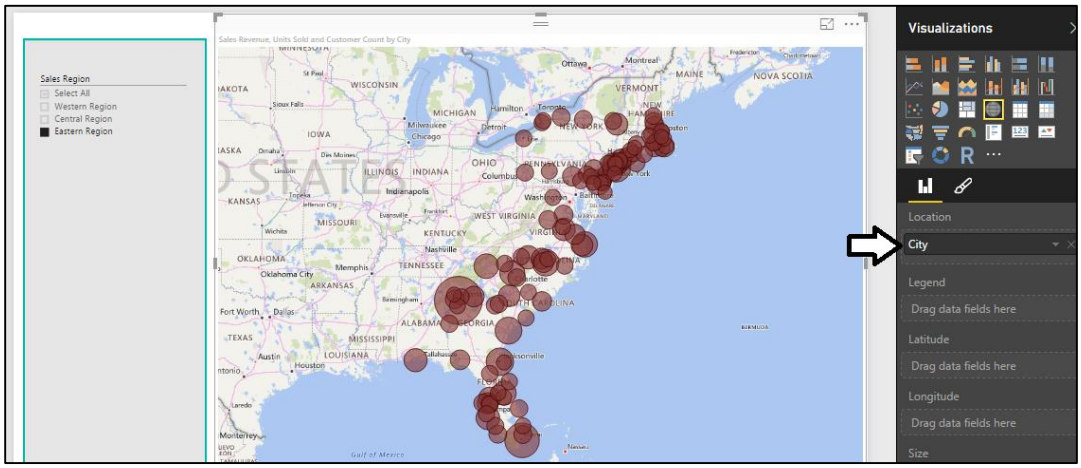


g) Try out the slicer. You should be able to select a single sales region and see the map update to reflect the new filtering.



© Critical Path Training. 2016. All Rights Reserved

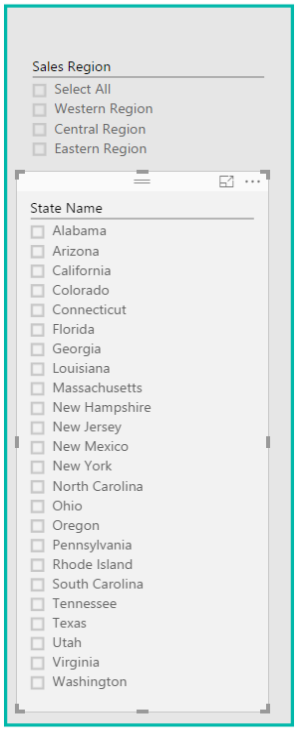
h) With the **Map** visual select, navigate to the **Field** properties pane. Remove the **State** field from the **Location** well and replace it with the **City** field from the **Customers** table. Now the red dots are more granular because they represent cities not states.



15. Add a second slicer below the first slicer to filter by **State**.
    a) Click the **New Visual** button on the ribbon to create a new visual.
    b) Click the **Slicer** button in the **Visualizations** list to change the visual into a slicer visual.
    c) Configure the new slicer by adding the **State Name** field from the **Customers** table into the **Field** well.

    d) Reposition the new slicer to it appears just below the first slicer as shown in the following screenshot.

© Critical Path Training. 2016. All Rights Reserved



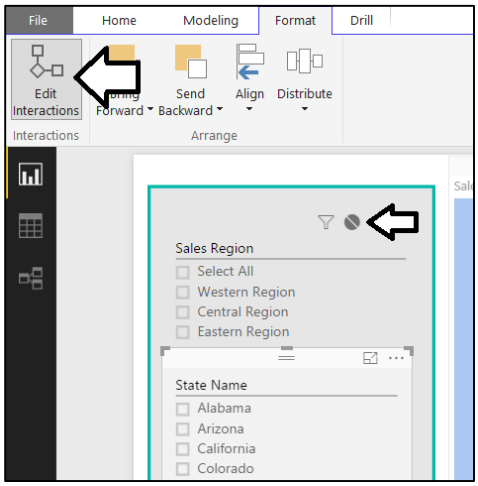
In the next step you will configure the edit interactions between the two slicer visuals. In particular, you do not want the button slider to place a filter on the top slicer.

e) Make sure the bottom slicer is selected.

f) Navigate to the **Format** tab in the ribbon.

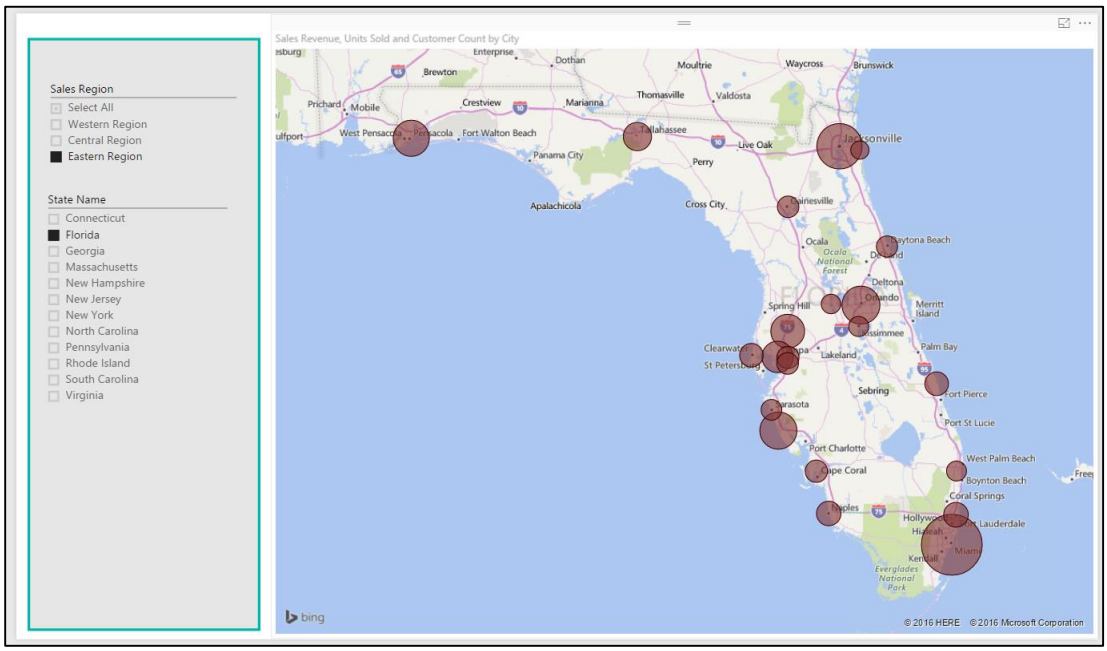
g) Click the **Edit Interactions** button in the ribbon to enter edit interaction mode.

h) In the **Sales Region** slicer, click the circle icon with the line through it to prevent the other slicer from affecting this slicer.



i) Test out the finished report page. You should be able to select a sales region from the top slicer which filters the set of states available in the bottom slicer. You should then be able to select a state and see its cities in the Map visual.

© Critical Path Training. 2016. All Rights Reserved



Congratulations. You have now completed this lab.

© Critical Path Training. 2016. All Rights Reserved