

Designing Interactive Reports in Power BI Desktop



Agenda

- Designing Interactive Reports
- Importing Custom Visuals
- Publishing a Project to the Power BI Service
- Using the Publish to Web Feature
- Publishing Reports in SharePoint Online
- Implementing Row-level Security (RLS)



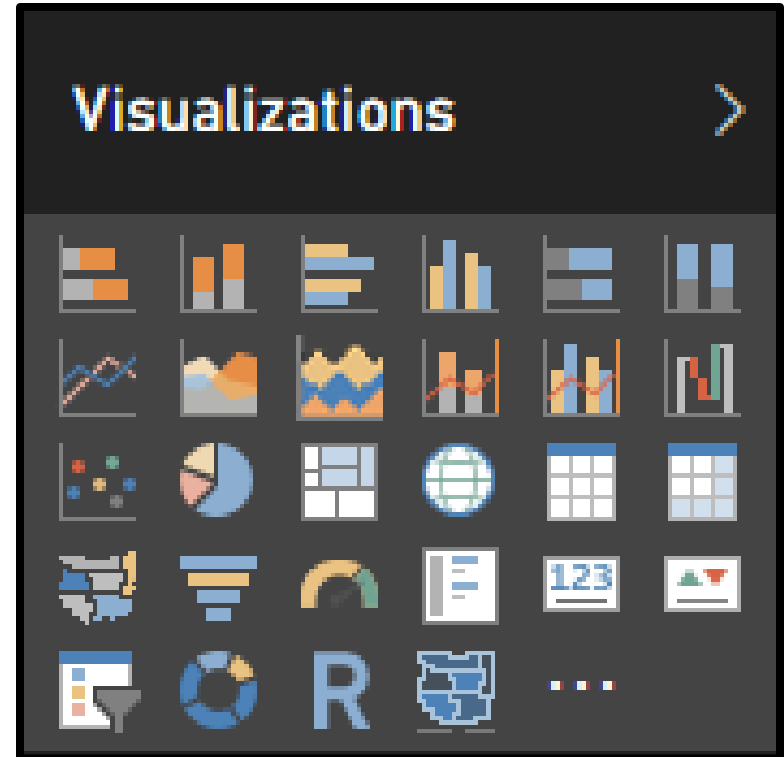
Creating Reports

- Power BI Desktop project contains one report
 - Report within project can contain multiple pages
 - Report pages contains visuals
- Reports can be created using filters
 - You can add filter to a specific visual
 - You can add page-level filters
 - You can add report-level filters
 - You can add interactive filters



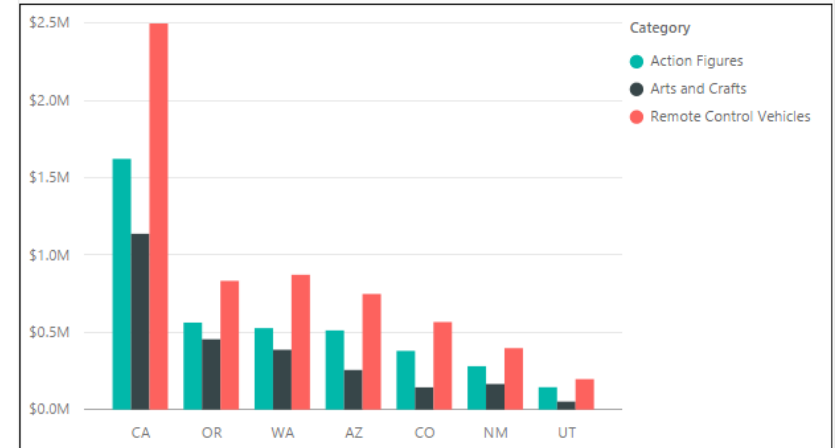
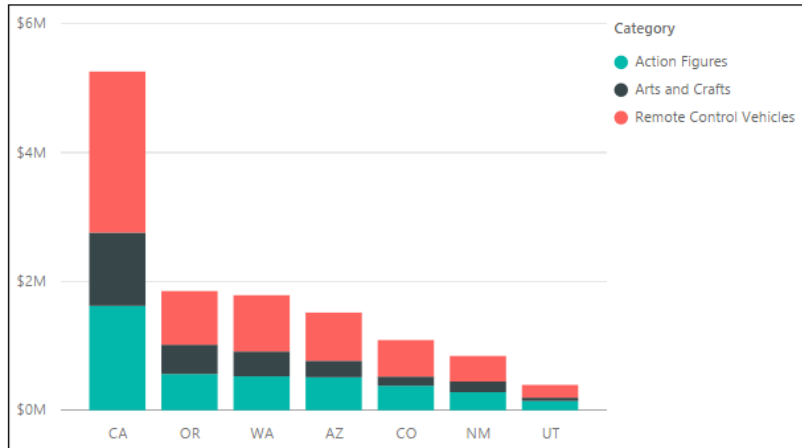
Built-in Visualization Types

- Table and Matrix
- Bar charts and Column charts
- Pie charts and Doughnut chart
- Line chart and Area chart
- Scatter chart and Combo charts
- Card and Multi-row Card
- Treemap
- Waterfall charts
- Funnel charts
- Gauge charts
- Map and Filled Map
- Slicer
- R script visual
- Shape map (in preview)

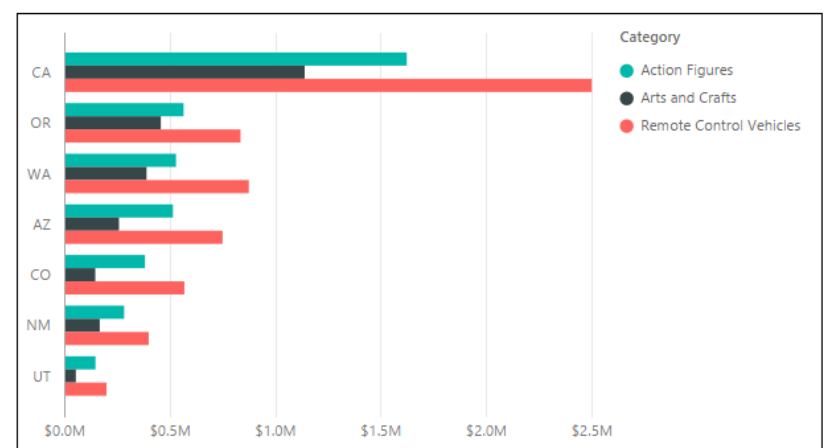
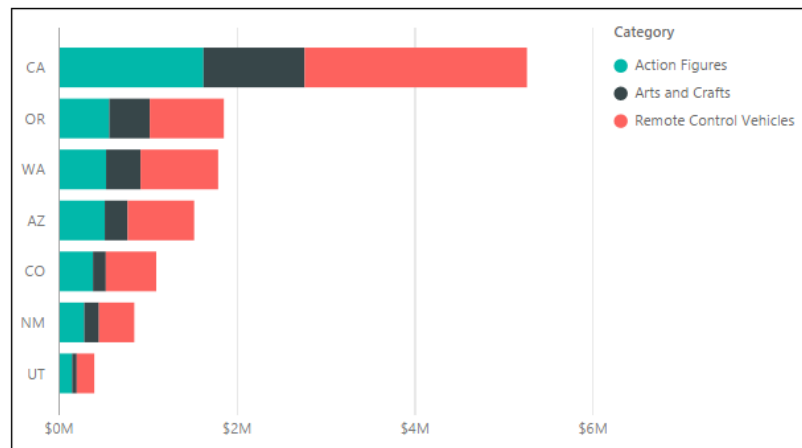


Column Chart and Bar Chart Variations

- Stacked Column Chart and Clustered Column Chart

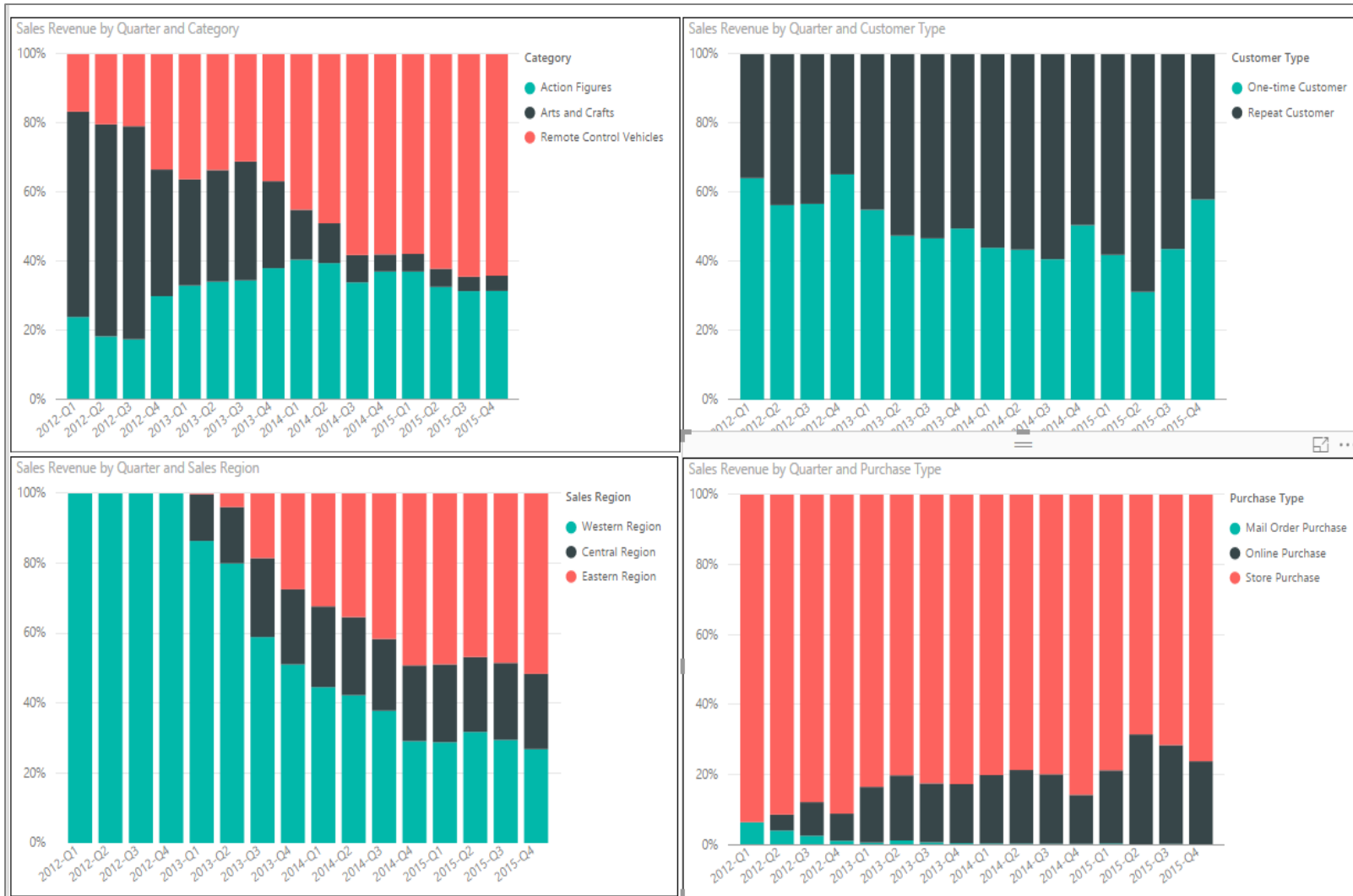


- Stacked Bar Chart and Clustered Bar Chart



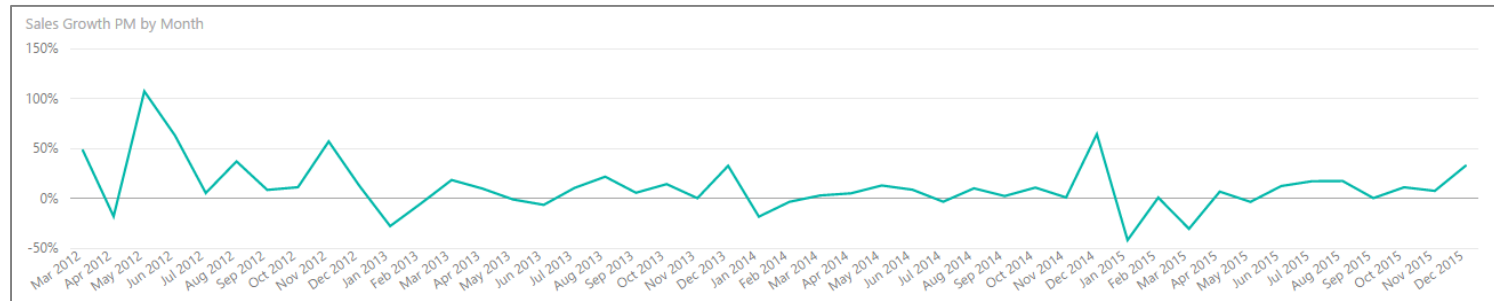
100% Stacked Column Chart

- Used to visual distribution over time across categories

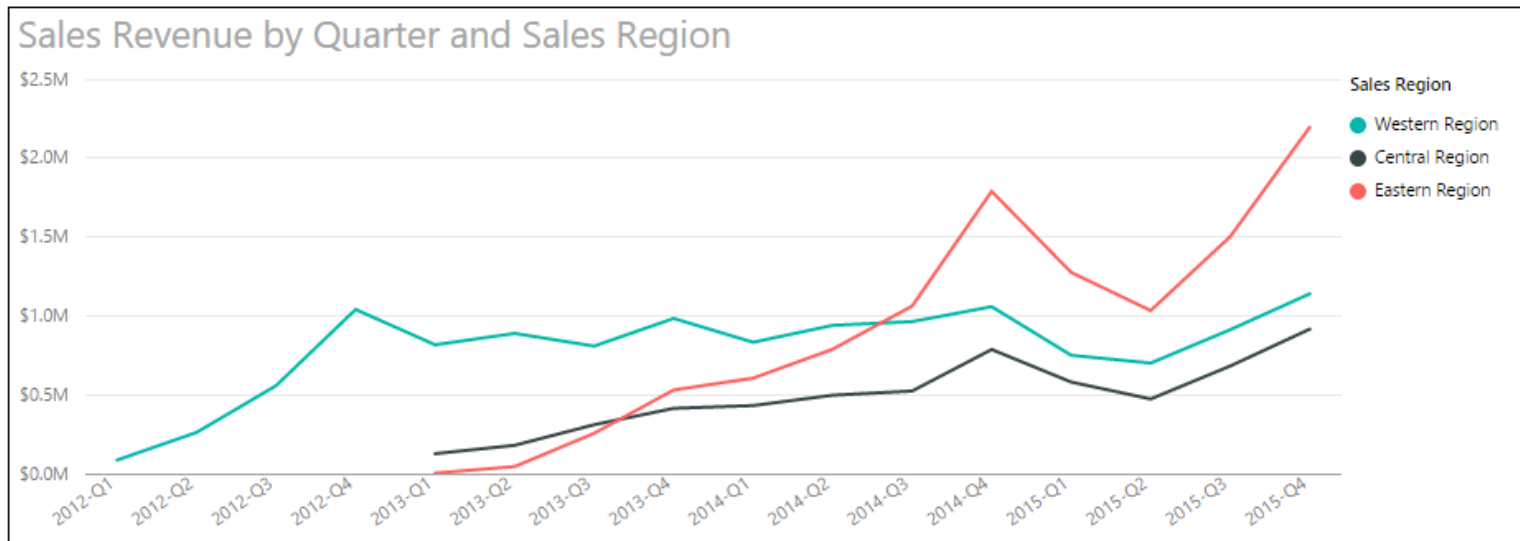


Line Charts

- Visualizes a series of data points across X and Y axis
 - Commonly used for time-based analysis

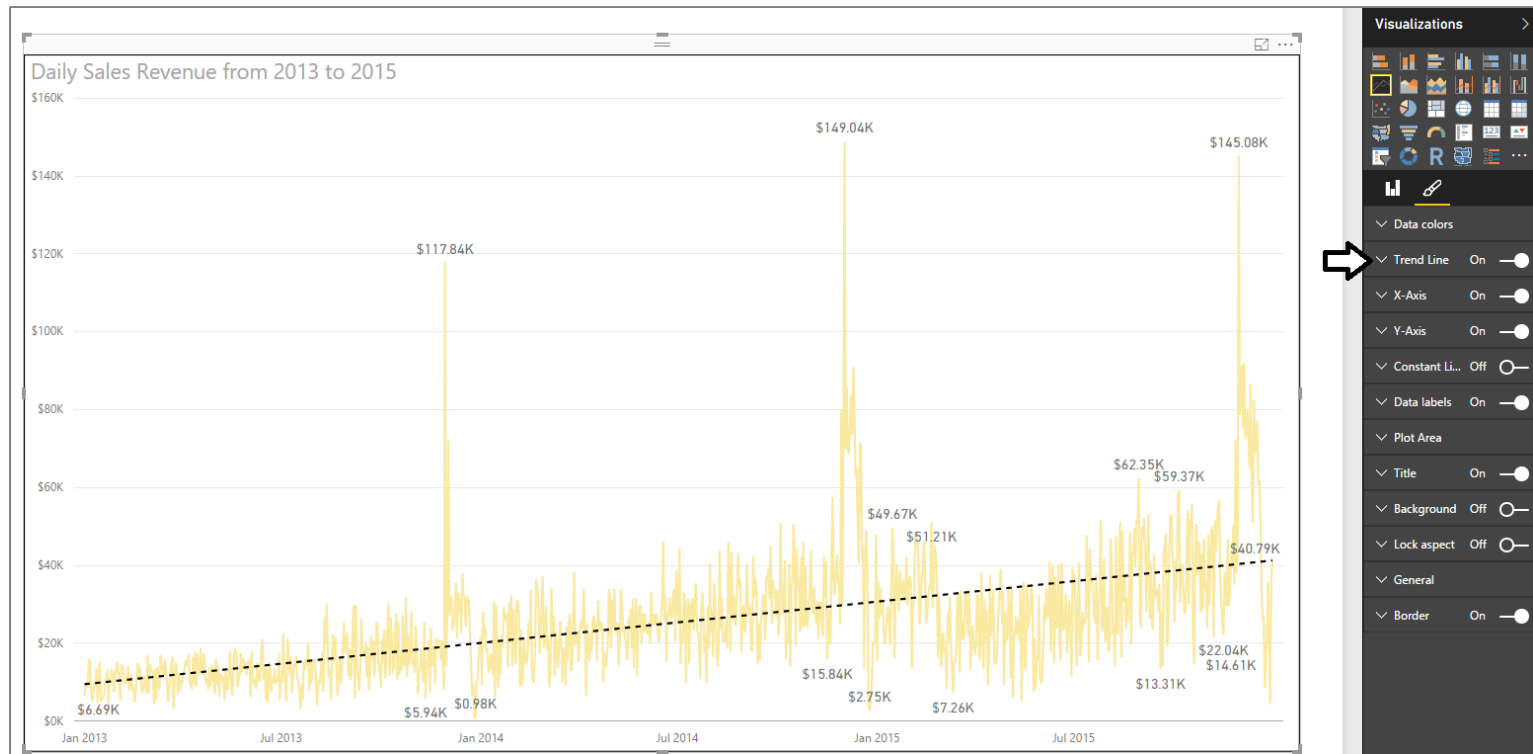


- Add field to Legend to create multiple lines for comparative analysis



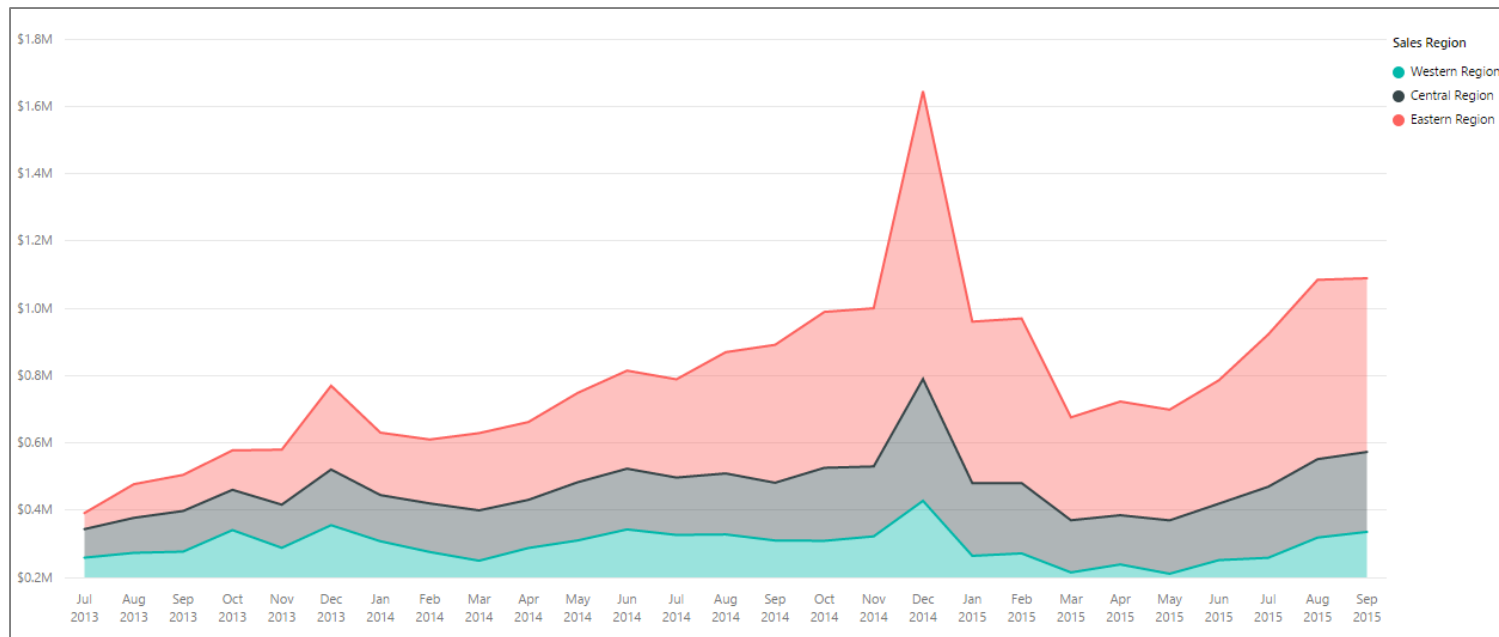
Trend Lines

- Used to visualize trends in series-based data
 - Flattens out the ups and downs
 - Used to determine if values are trending up or down



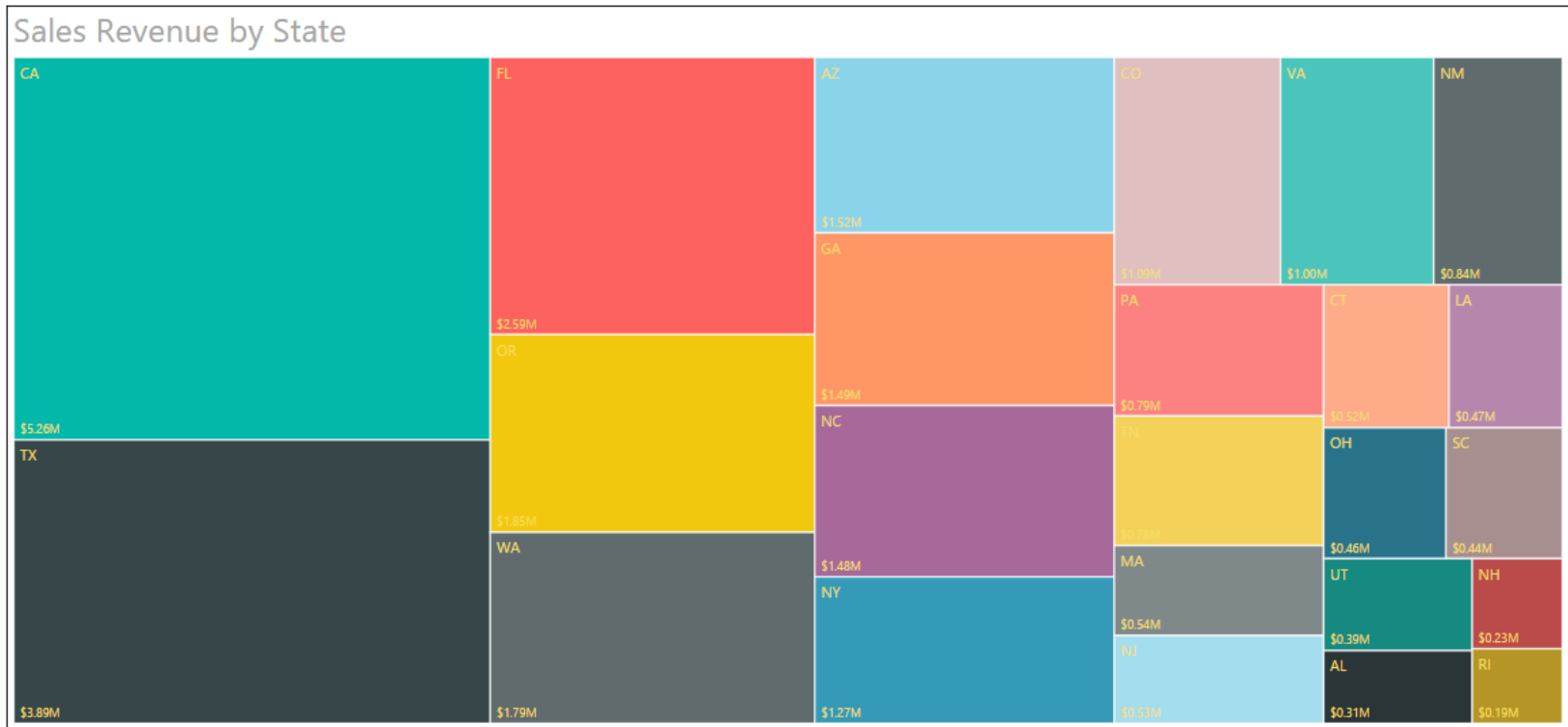
Stacked Area Chart

- Basically, a line chart with a little more personality
 - Areas under lines filled with colors



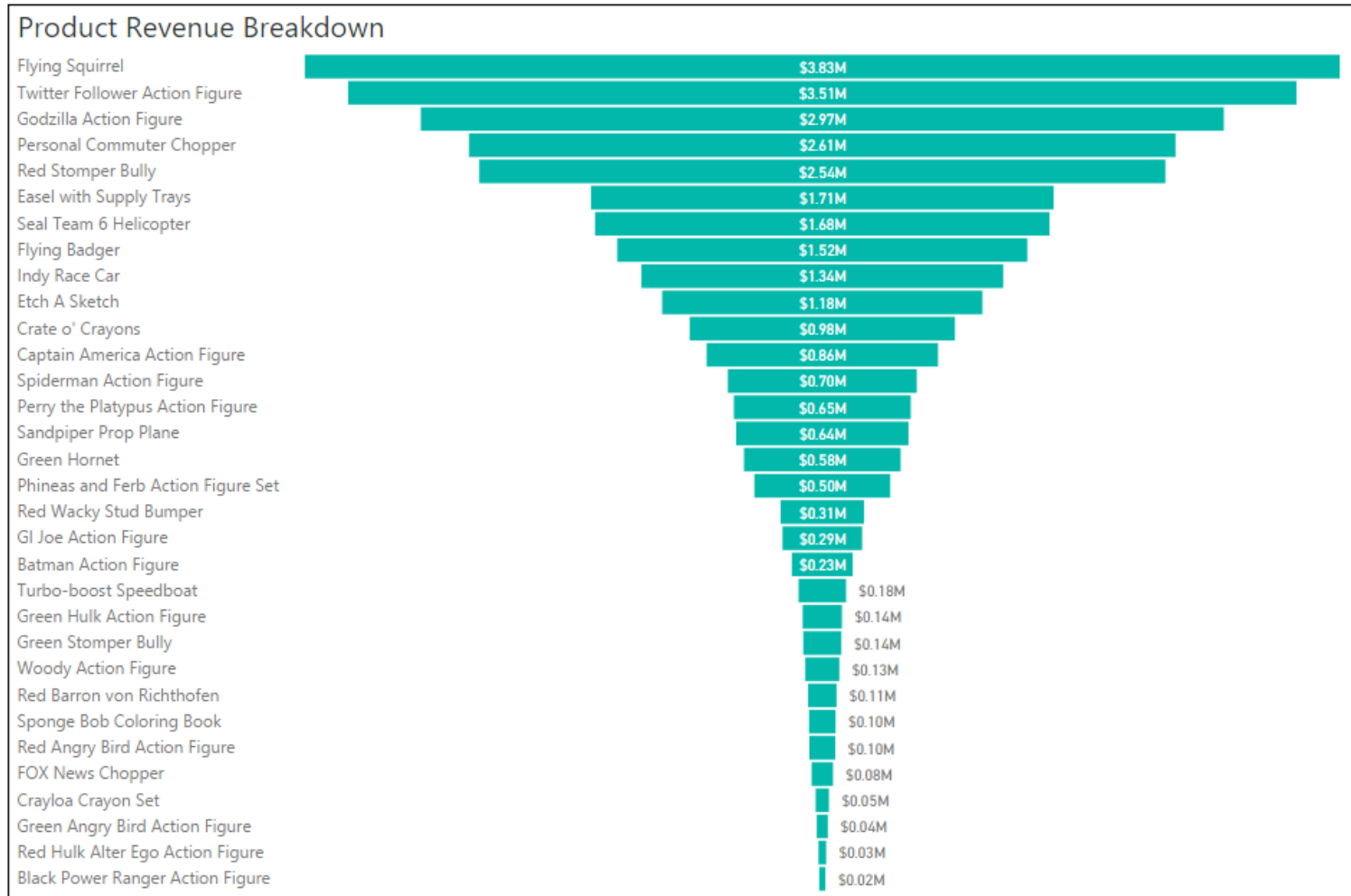
Treemap

- Simple visualization of category distribution



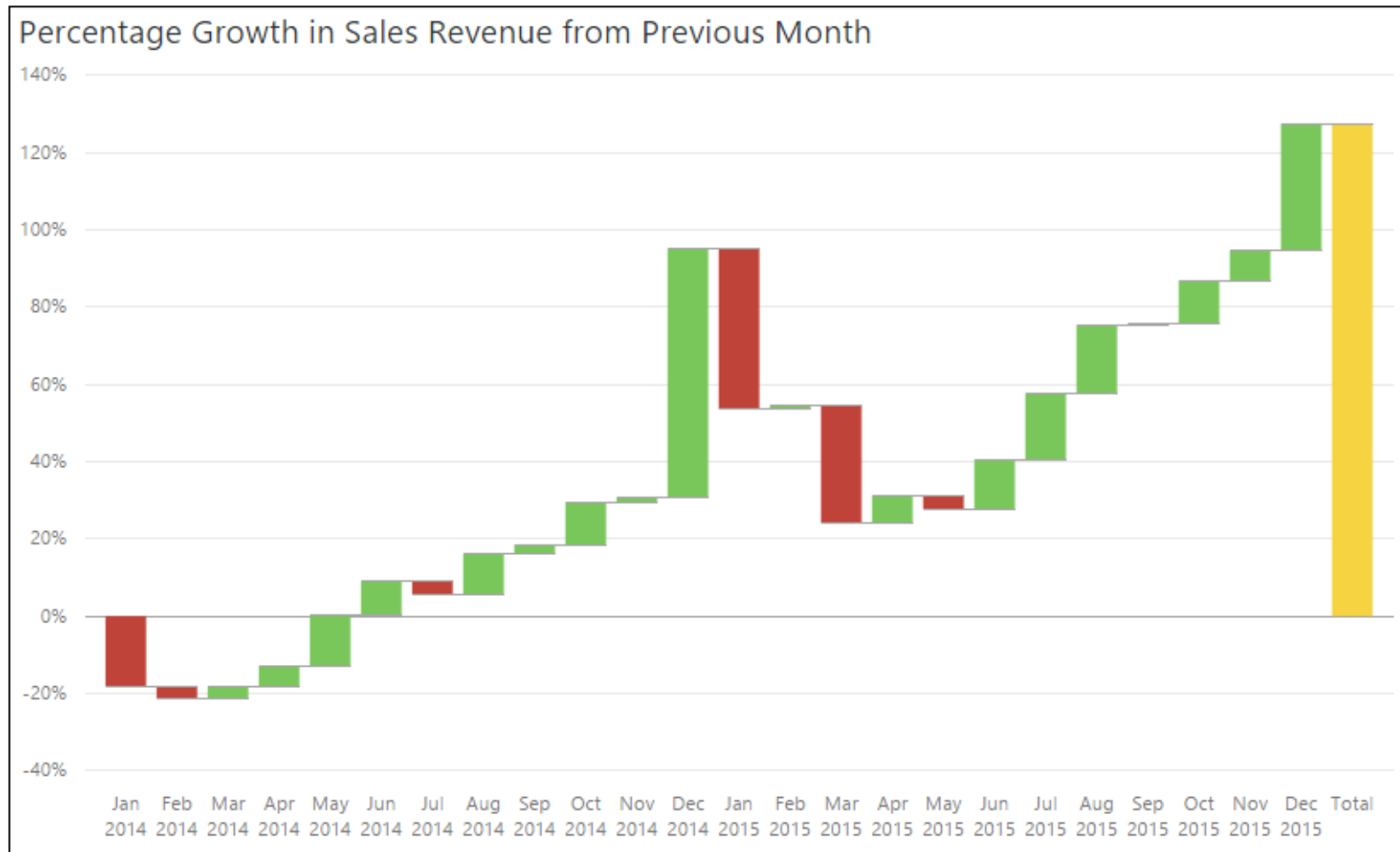
Funnel

- Visualizes distribution across categories as percentage of top value



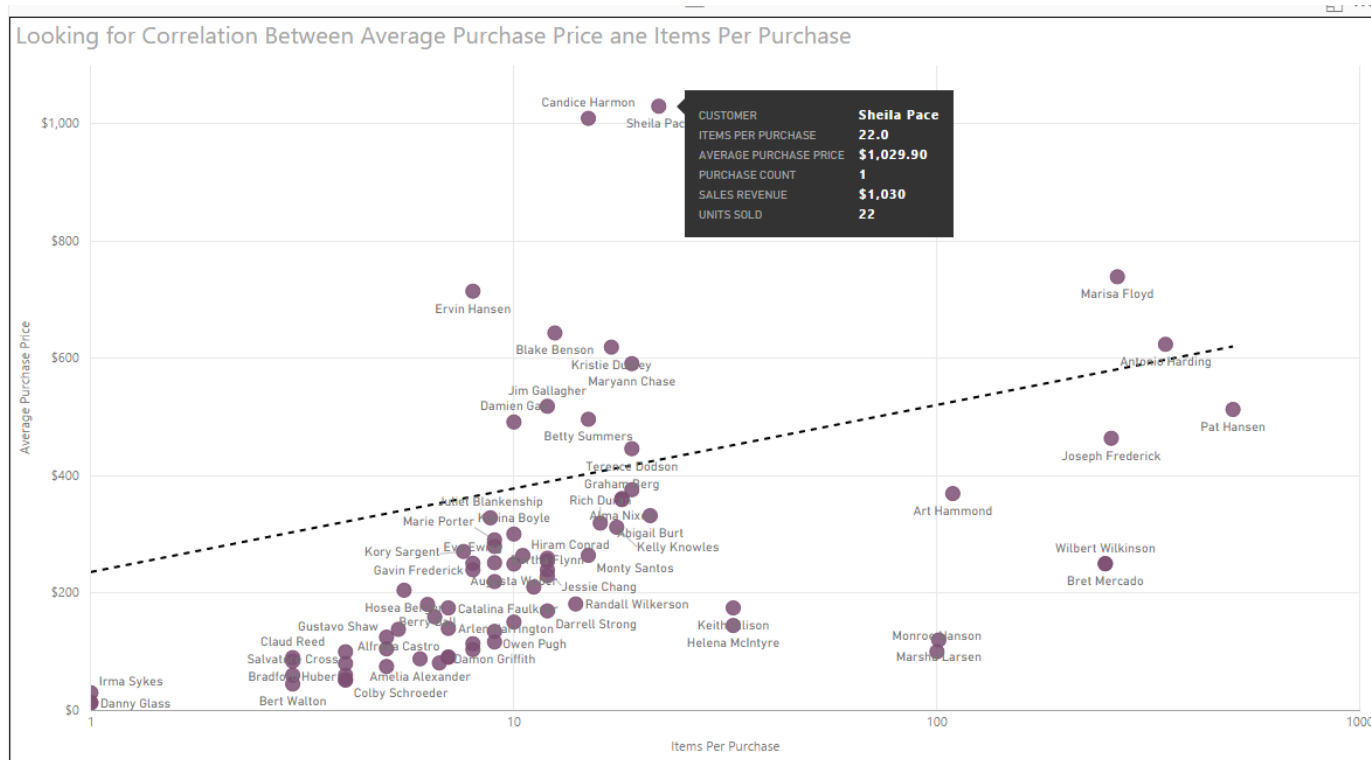
Waterfall

- Visualizes series-based data with positive and negative values



Scatter Chart

- Visualizes set of data points when looking for correlation
 - Scatter chart used to discover correlation between two variables
 - Each data point has two values which are mapped to X and Y axis

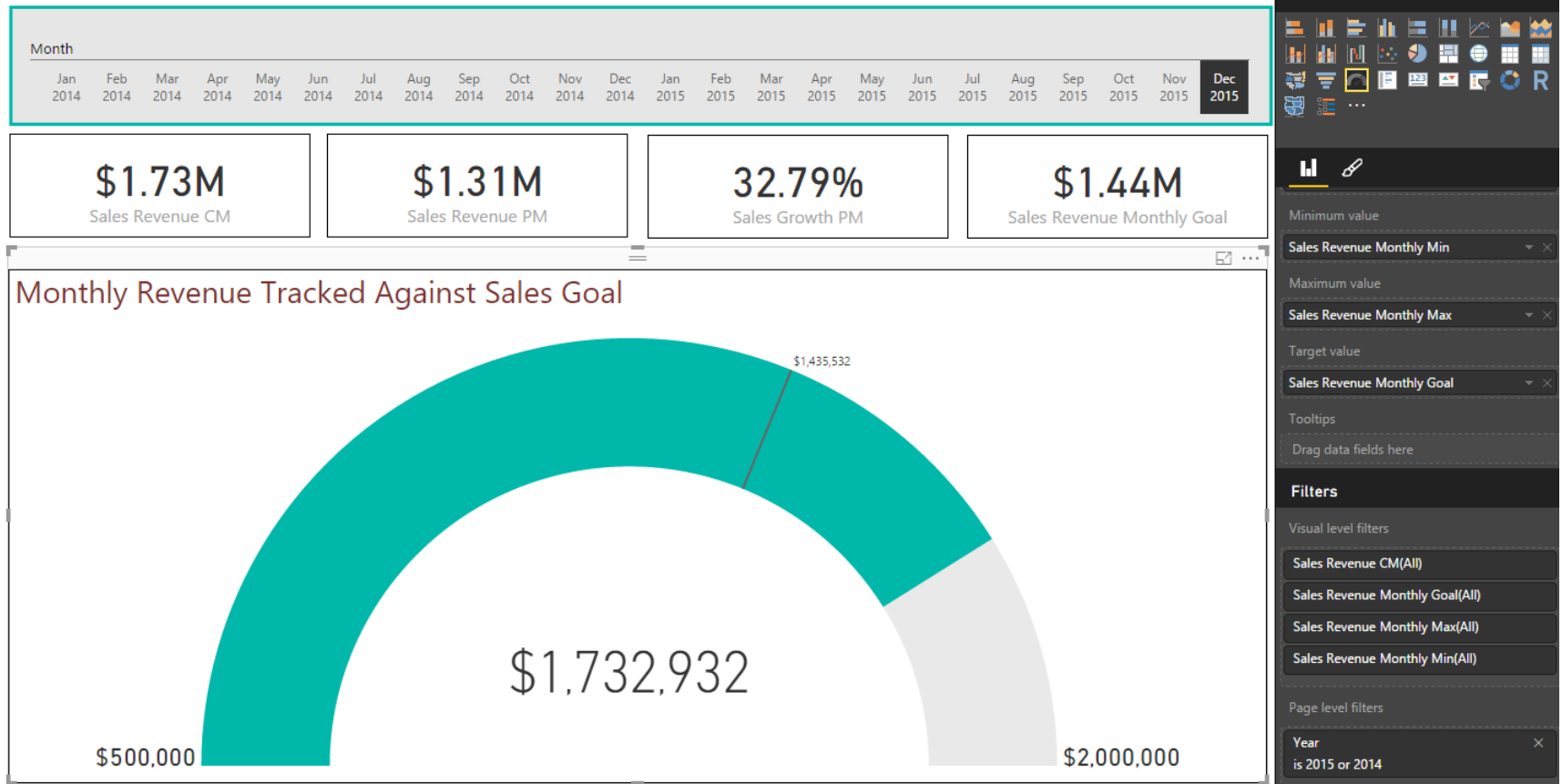


se price?



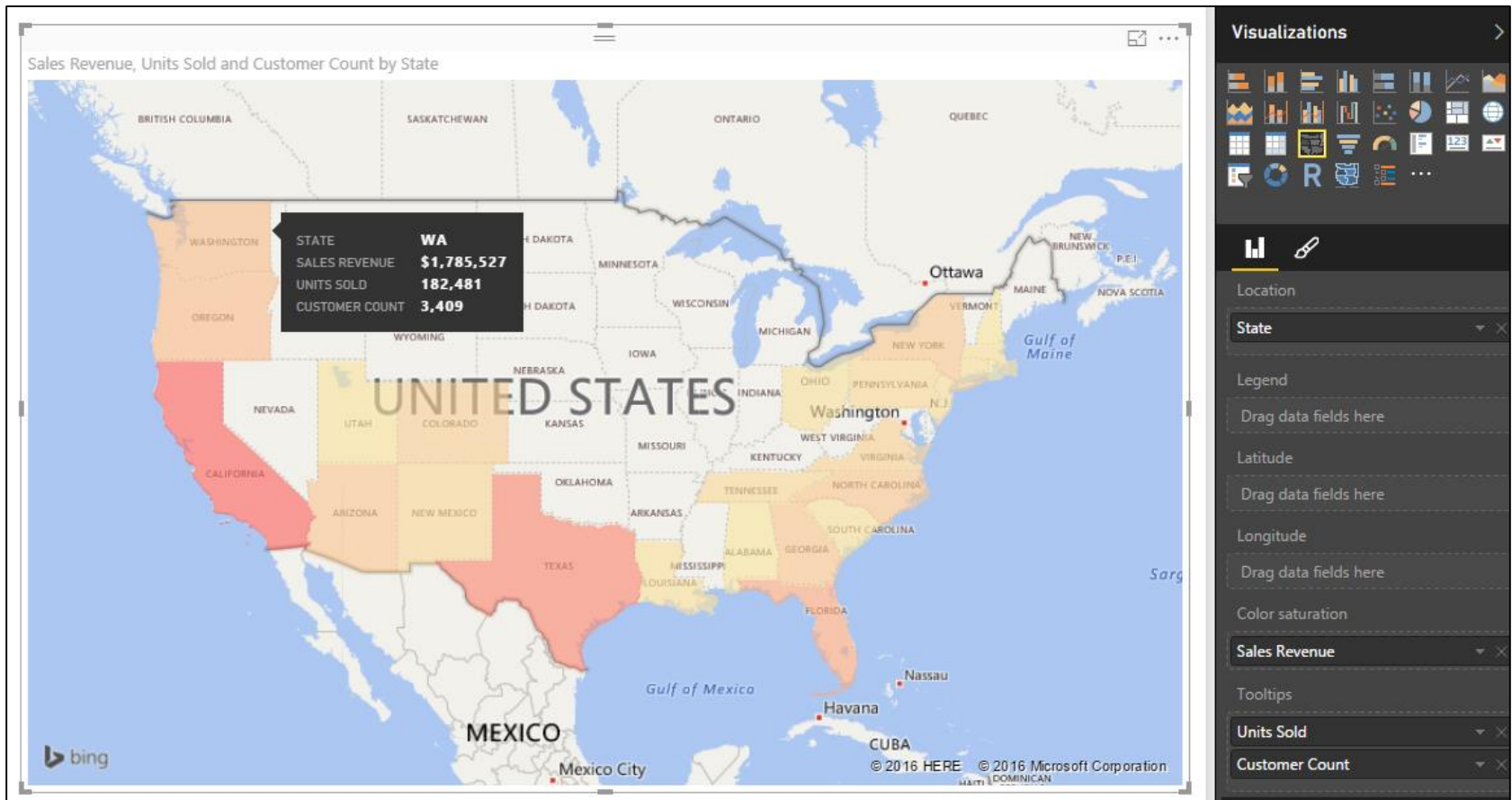
Gauge Visual

- Visualizes how measured value is tracking against goal or budget



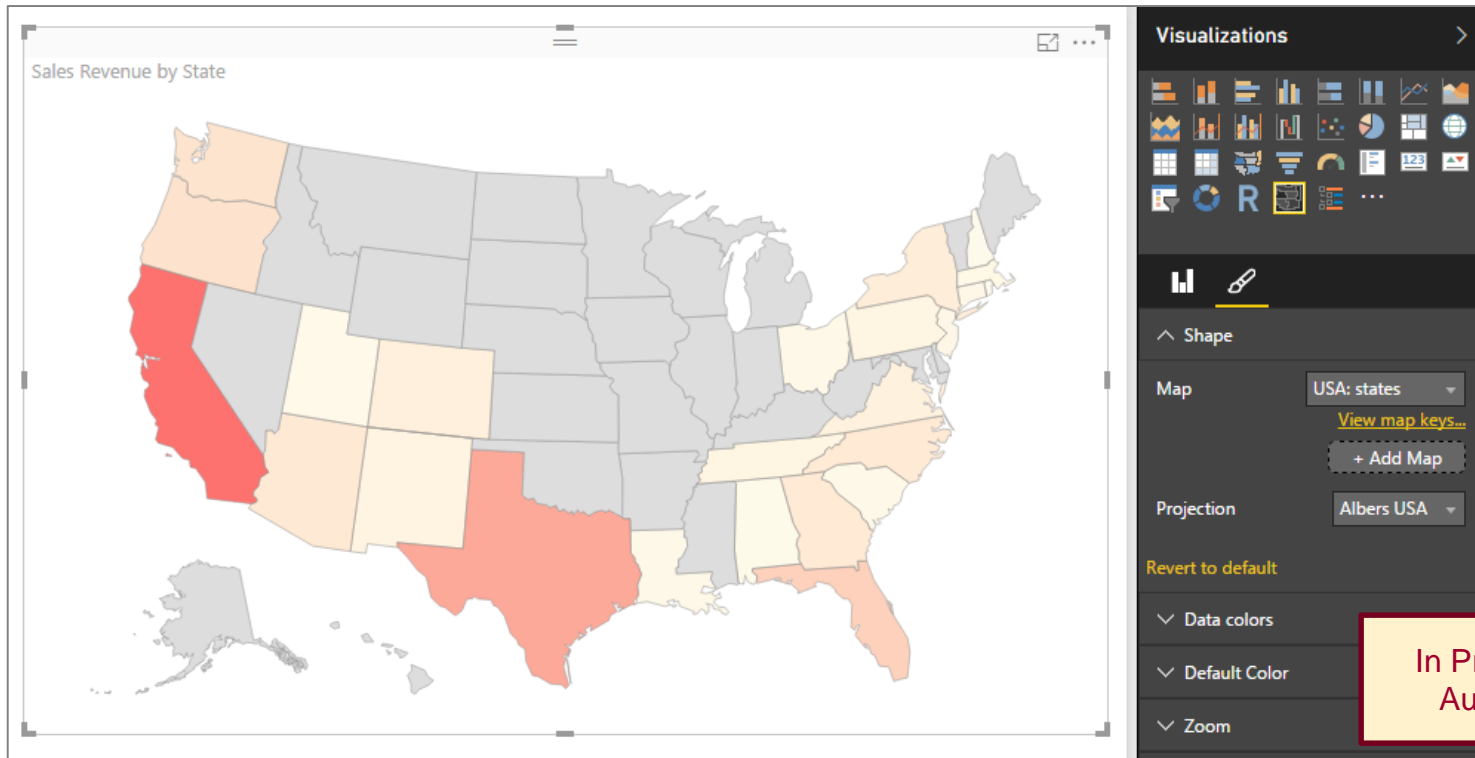
Filled Map

- Visualizes distribution across states and countries



Inline Shape Map

- Similar to filled map with a few important exceptions
 - Based on **TopoJSON** map format created by ESRI
 - Allows for creation of custom maps using JSON
 - Create maps for geography, seating arrangements, floor plans, etc.





DEMO

Exploring Power BI Visuals

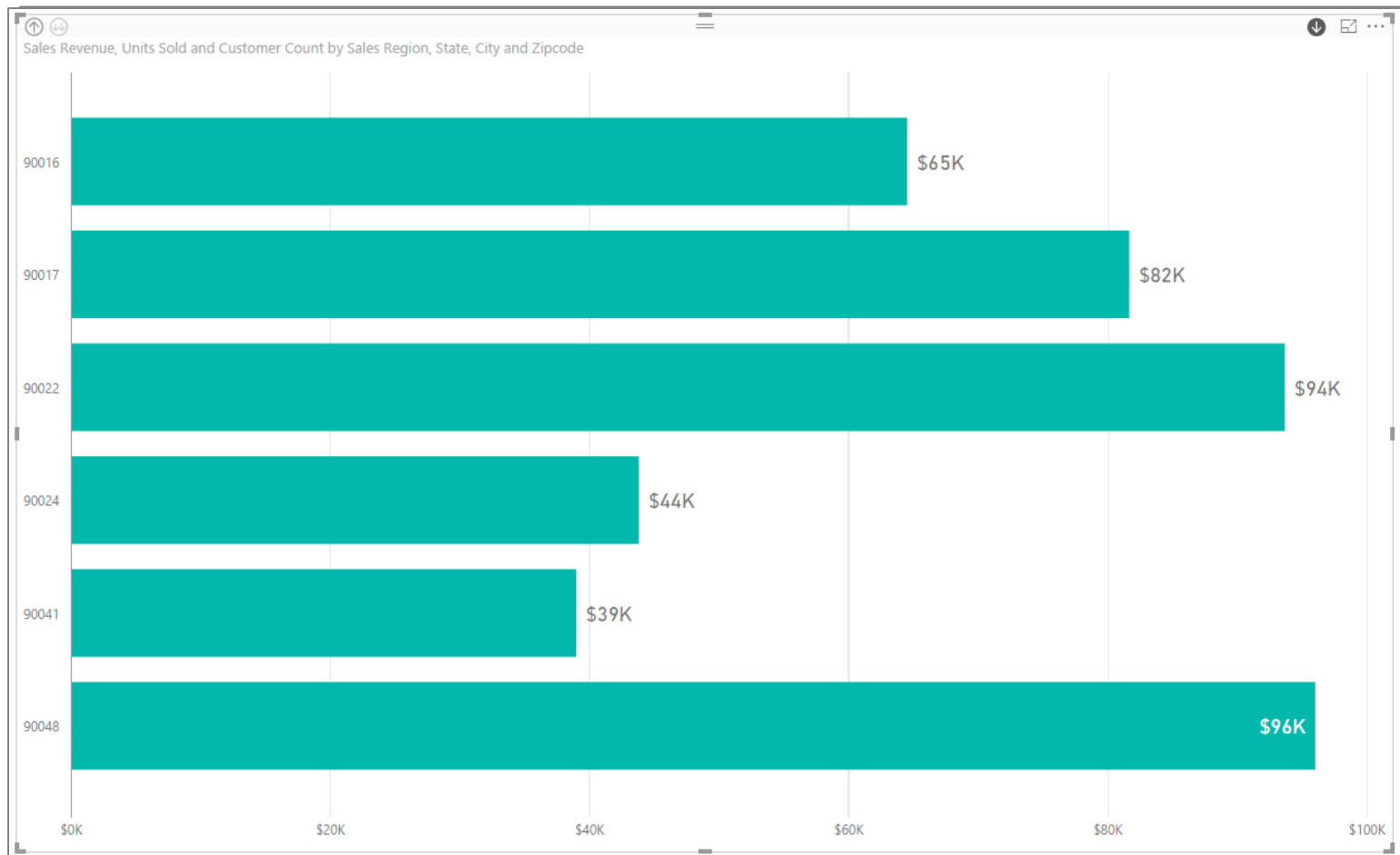
User Interaction with Slicers & Highlighting

- Provides user with interactive filtering control



User Interaction using Drill Actions

- Drill Actions supported when using hierarchies
 - You must enable drilldown mode in visual



Ranking Products By Sales using RANKX

- DAX provides RANKX function for ranking
 - Can be used to track top 5 products by sales revenue

```
Product Rank =  
RANKX(  
    ALL(Products),  
    CALCULATE( SUM(Sales[SalesAmount]) )  
)
```

- You can sort and filter on output of RANKX function

Product Rank ▲	Product	Sales Revenue
1	Flying Squirrel	\$3,828,783
2	Twitter Follower Action Figure	\$3,508,806
3	Godzilla Action Figure	\$2,970,735
4	Personal Commuter Chopper	\$2,613,193
5	Red Stomper Bully	\$2,538,233

Product Rank ▲

is less than or equal to...

Show items when the value:

is less than or equal to ▼

5

☒ And ☐ Or

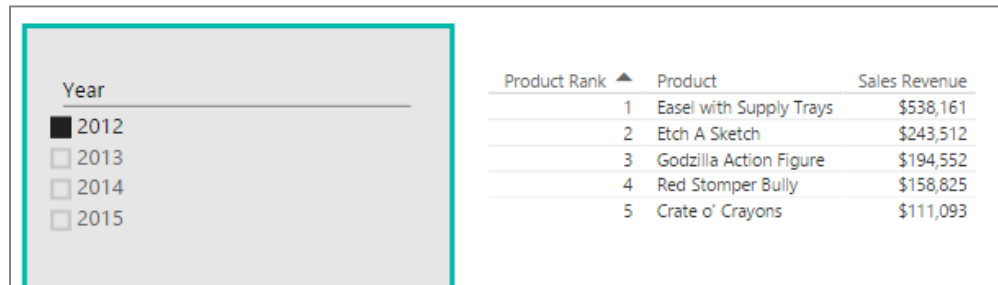
▼

Apply filter



Problems with the Filter Context

- RANKX function is affected by filter context
 - Sometimes you get the results you are expecting



The screenshot shows a filter panel on the left with a 'Year' section containing checkboxes for 2012, 2013, 2014, and 2015. The 2012 checkbox is selected. To the right is a table with three columns: 'Product Rank' (with an upward arrow), 'Product', and 'Sales Revenue'.

Product Rank	Product	Sales Revenue
1	Easel with Supply Trays	\$538,161
2	Etch A Sketch	\$243,512
3	Godzilla Action Figure	\$194,552
4	Red Stomper Bully	\$158,825
5	Crate o' Crayons	\$111,093

- Sometimes you might get unexpected results



The screenshot shows a filter panel on the left with two sections: 'Year' with checkboxes for 2012, 2013, 2014, and 2015, and 'Category' with checkboxes for Action Figures, Arts and Crafts, and Remote Control Vehicles. The 'Action Figures' checkbox is selected. To the right is a table with three columns: 'Product Rank' (with an upward arrow), 'Product', and 'Sales Revenue'.

Product Rank	Product	Sales Revenue
2	Twitter Follower Action Figure	\$3,508,806
3	Godzilla Action Figure	\$2,970,735

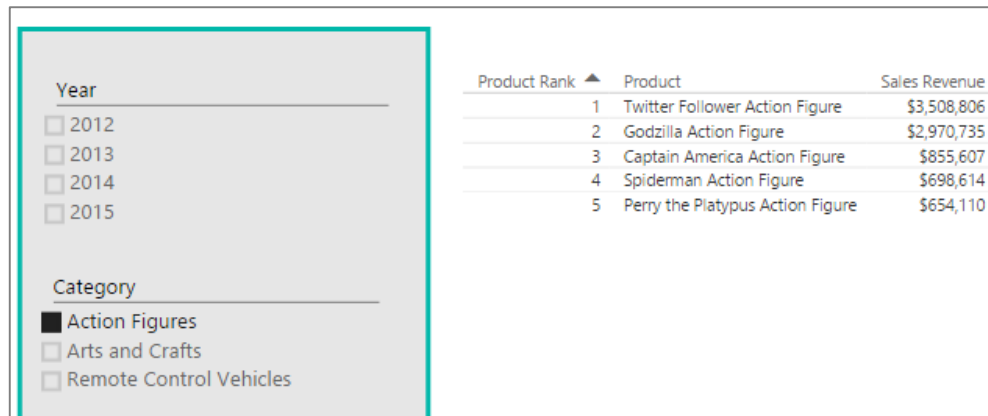


Writing Context Aware DAX Code

- When using RANKX...
 - It's recommended to call **HASONEVALUE** function
 - When calling ALL function, pass one or more columns

```
Product Rank =  
IF(  
    HASONEVALUE(Products[Product]),  
    RANKX(  
        ALL( Products[Subcategory], Products[Product] ),  
        CALCULATE( SUM(Sales[SalesAmount]) )  
    )  
)
```

- Ranking function now evaluates product ranking for specific Category










The screenshot shows a Power BI interface. On the left is a filter pane with two sections: 'Year' and 'Category'. Under 'Year', there are checkboxes for 2012, 2013, 2014, and 2015. Under 'Category', there are checkboxes for 'Action Figures' (which is selected with a black square), 'Arts and Crafts', and 'Remote Control Vehicles'. On the right is a table with three columns: 'Product Rank' (with an upward arrow icon), 'Product', and 'Sales Revenue'. The table contains five rows of data.

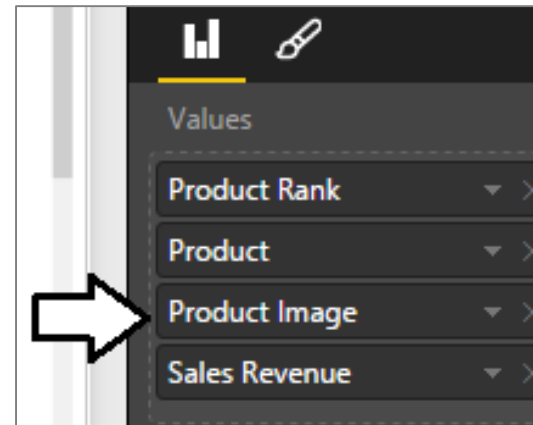
Product Rank	Product	Sales Revenue
1	Twitter Follower Action Figure	\$3,508,806
2	Godzilla Action Figure	\$2,970,735
3	Captain America Action Figure	\$855,607
4	Spiderman Action Figure	\$698,614
5	Perry the Platypus Action Figure	\$654,110



More Ranking Evaluation Problems

- Adding new column to table creates new problem
 - Ranking run separately for each separate Product Image
 - Every product has unique Product Image and is given rank of 1

Product Rank	Product	Product Image	Sales Revenue
1	Batman Action Figure		\$40,395
1	Black Power Ranger Action Figure		\$4,223
1	Captain America Action Figure		\$125,110
1	Crate o' Crayons		\$322,711
1	Crayloa Crayon Set		\$12,868
1	Easel with Supply Trays		\$928,620
1	Etch A Sketch		\$293,175

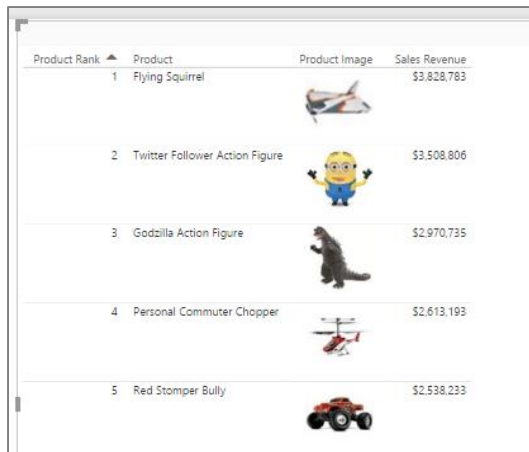


Getting It Right






- Call to RANKX must be modified again
 - You must specify which columns to factor into ranking

```
Product Rank =  
IF(  
    HASONEVALUE(Products[Product]),  
    RANKX(  
        ALL( Products[Subcategory], Products[Product], Products[Product Image] ),  
        CALCULATE( SUM(Sales[SalesAmount]) )  
    )  
)
```

- Context-aware DAX code corrects problems with visual



A screenshot of a Power BI table visual displaying a ranked list of products. The table has four columns: 'Product Rank' (with a sort arrow), 'Product', 'Product Image', and 'Sales Revenue'. The data is sorted by rank, showing the top five products. Each row includes a small image of the product.

Product Rank	Product	Product Image	Sales Revenue
1	Flying Squirrel		\$3,828,783
2	Twitter Follower Action Figure		\$3,508,806
3	Godzilla Action Figure		\$2,970,735
4	Personal Commuter Chopper		\$2,613,193
5	Red Stomper Bully		\$2,538,233



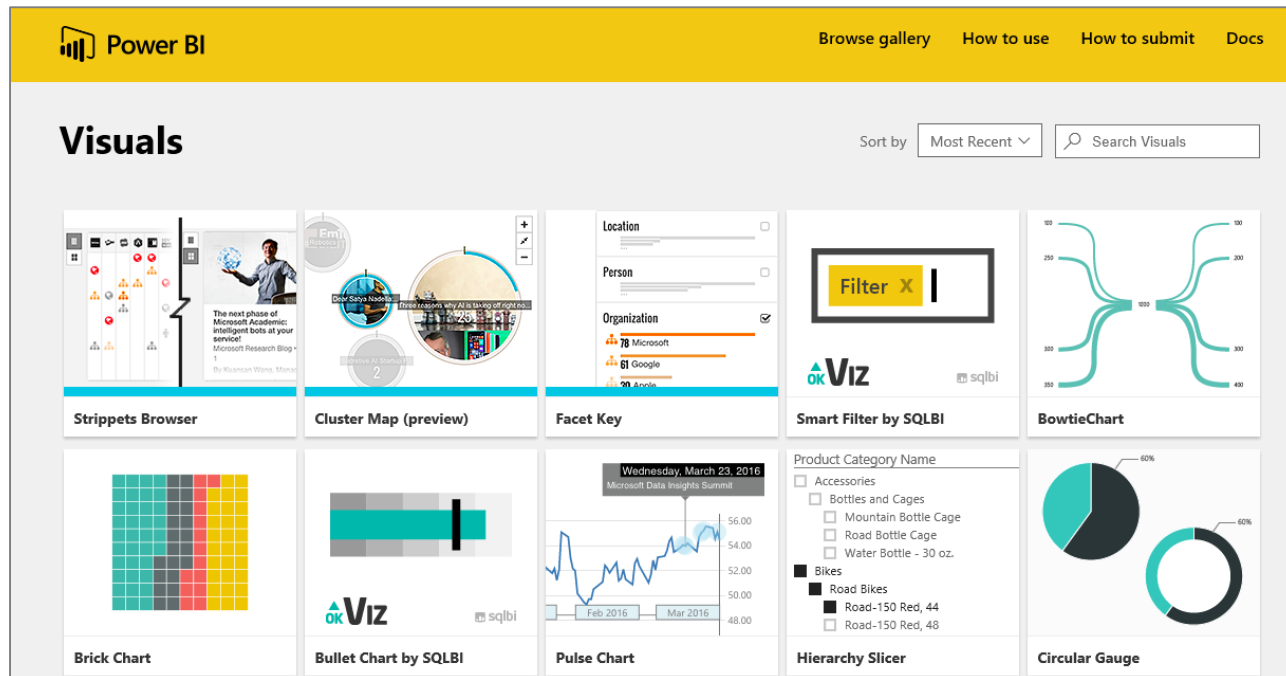
Agenda

- ✓ Designing Interactive Reports
- Importing Custom Visuals
- Publishing a Project to the Power BI Service
- Using the Publish to Web Feature
- Publishing Reports in SharePoint Online
- Implementing Row-level Security (RLS)



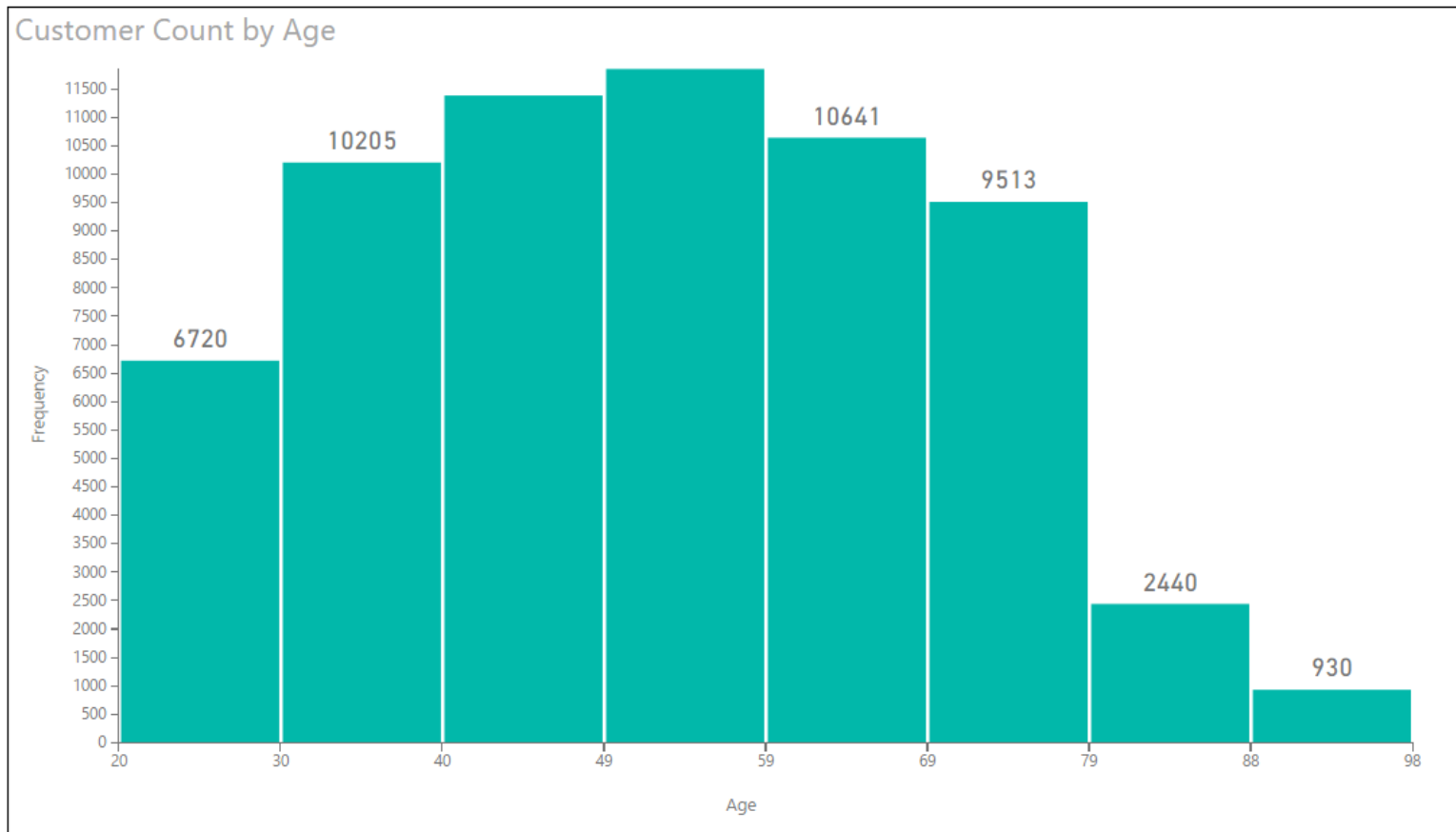
Custom Visual Gallery

- Power BI Framework for Visuals is Extensible
 - Developers can extend Power BI with Custom Visuals
 - Microsoft Hosts gallery of custom visuals
 - Gallery located at <https://app.powerbi.com/visuals/>



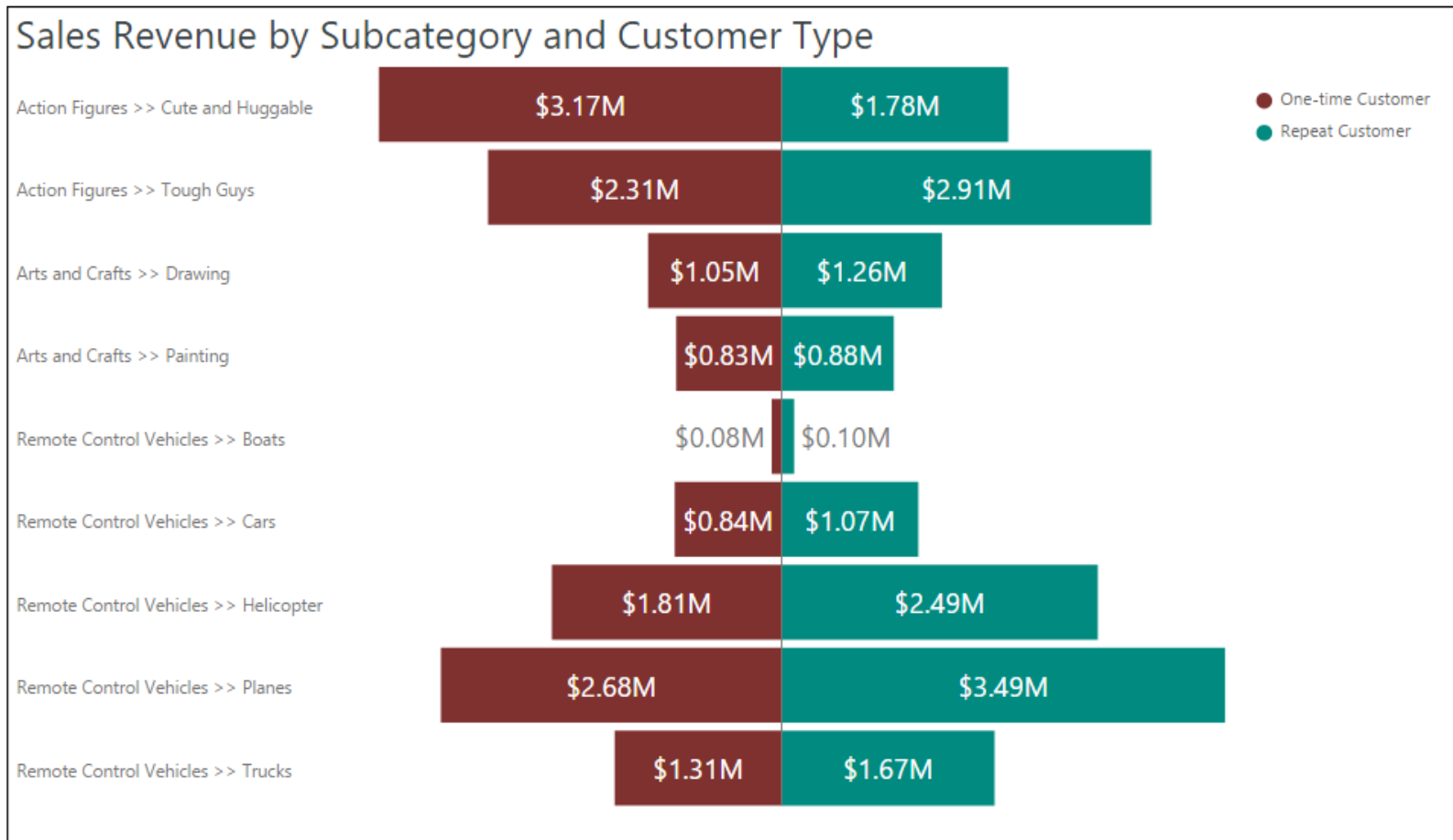
Histogram

- Custom Visual Example 1



Tornado Chart

■ Custom Visual Example 2



Spark Lines

- Custom Visual Example 3

Holiday Season Data Analysis

Daily Sales in Q4 of 2012



Daily Sales in Q4 of 2013



Daily Sales in Q4 of 2014



Daily Sales in Q4 of 2015



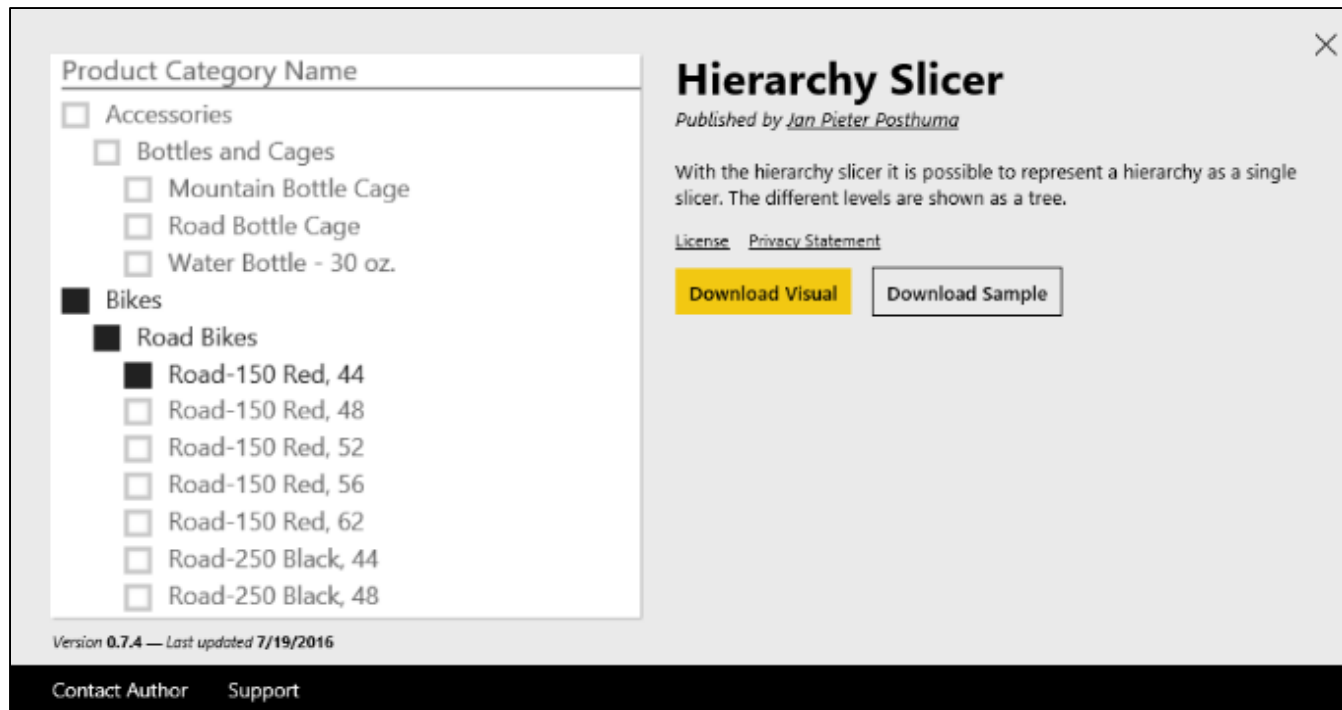
Hierarchy Slicer

■ Custom Visual Example 4



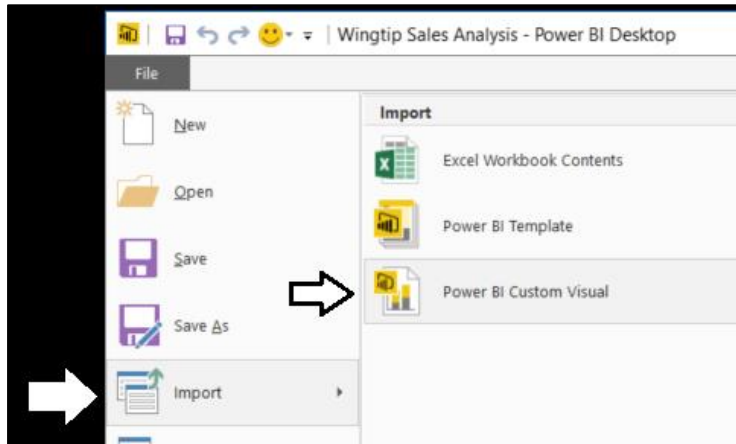
Downloading & Importing a Custom Visual

- Visual must be downloaded from Visuals Gallery
 - Custom Visual files packaged in PBIVIZ File
 - Custom Visual can be imported into Power BI Desktop project
 - Custom Visual can be imported into workspace in Power BI service

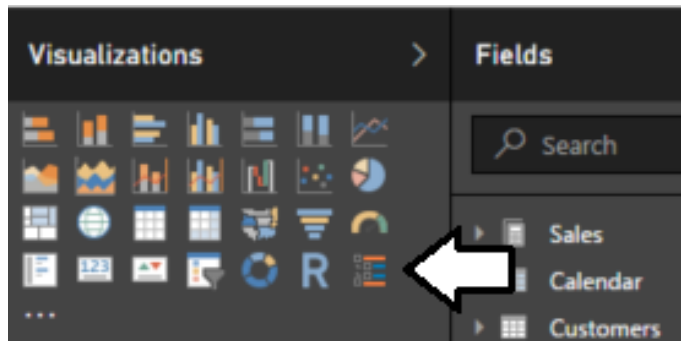


Importing a Custom Visual

- Import custom visual into Power BI Desktop project
 - Execute **Import > Power BI Custom Visual** menu command



- After import, Custom Visual appears in **Visualizations** list



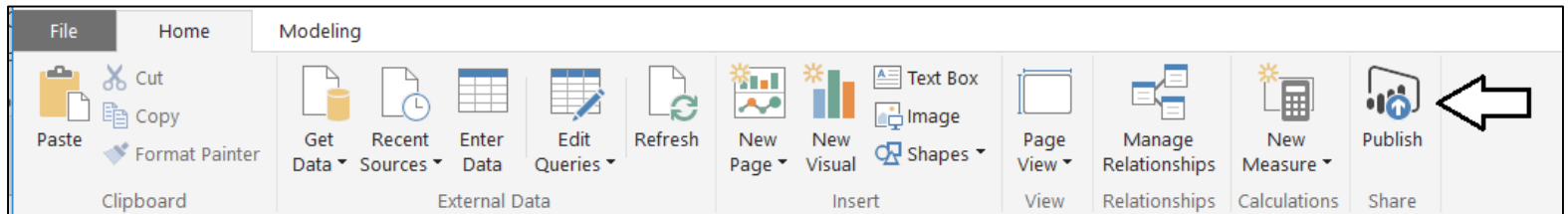
Agenda

- ✓ Designing Interactive Reports
- ✓ Importing Custom Visuals
- Publishing a Project to the Power BI Service
 - Using the Publish to Web Feature
 - Publishing Reports in SharePoint Online
 - Implementing Row-level Security (RLS)

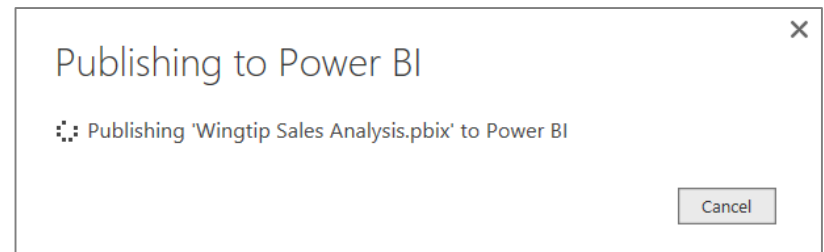
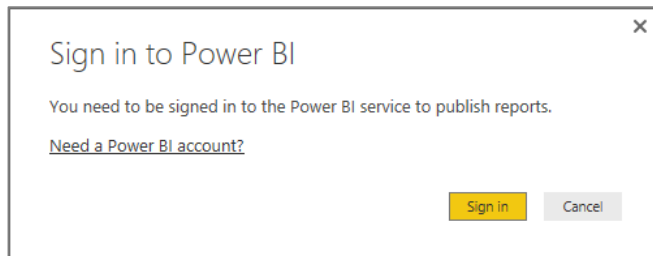


Publishing a Power BI Desktop Project

- Power BI Desktop provides **Publish** command
 - Used to publish project to Power BI service



- Requires logging into your Office 365 account

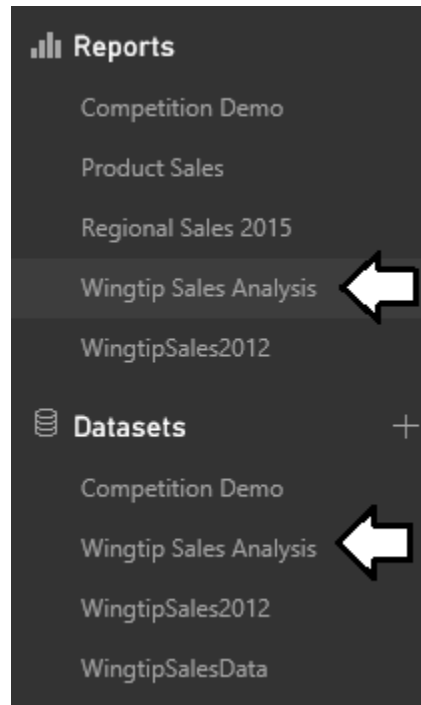


- Published articles added to a specific workspace



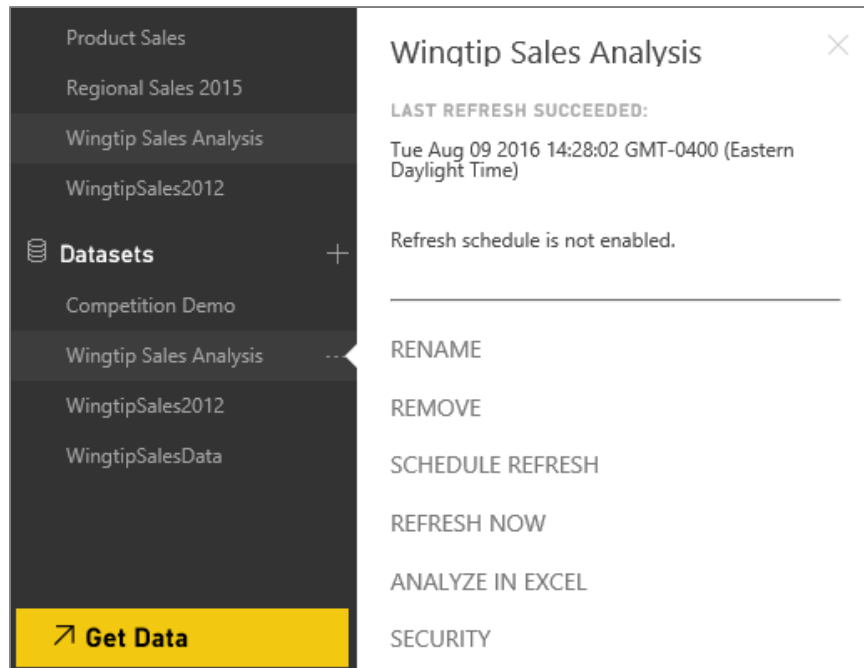
Examining What's Been Published

- What does project publishing add to workspace?
 - One dataset with same name as project
 - One report with same name as project



Dataset Configuration

- You can configure Dataset after its been published
 - Configure data source credentials
 - Configure refresh schedule
 - Configure Row-level Security



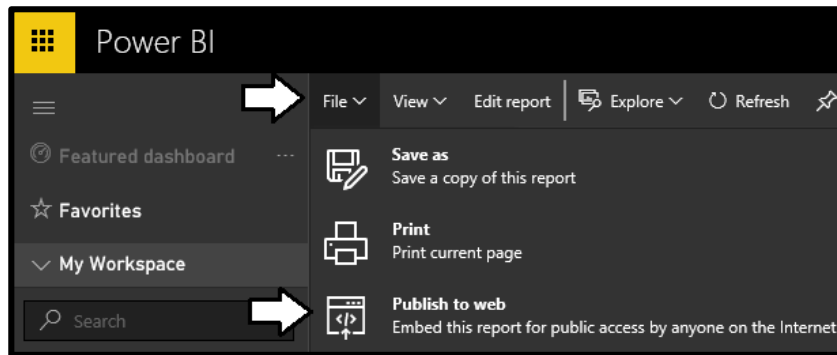
Agenda

- ✓ Designing Interactive Reports
- ✓ Importing Custom Visuals
- ✓ Publishing a Project to the Power BI Service
- Using the Publish to Web Feature
 - Publishing Reports in SharePoint Online
 - Implementing Row-level Security (RLS)

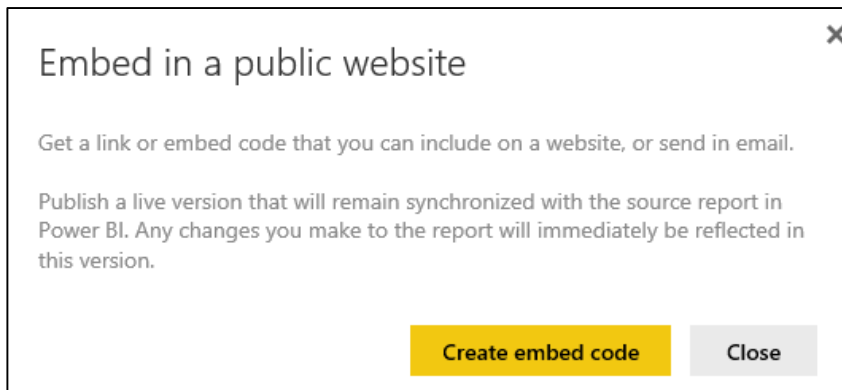


Publish to Web

- **Publish to Web** command available on reports
 - Not supported for reports and datasets which implement RLS



- **Publish to Web** command used to generate embed codes



Generating Embed Codes

- Used to provide anonymous access to report
 - Provide link which can be posted, emailed or texted
 - Provides **iFrame** HTML element for embedding in public web site

Success! ✕

Link you can send in email

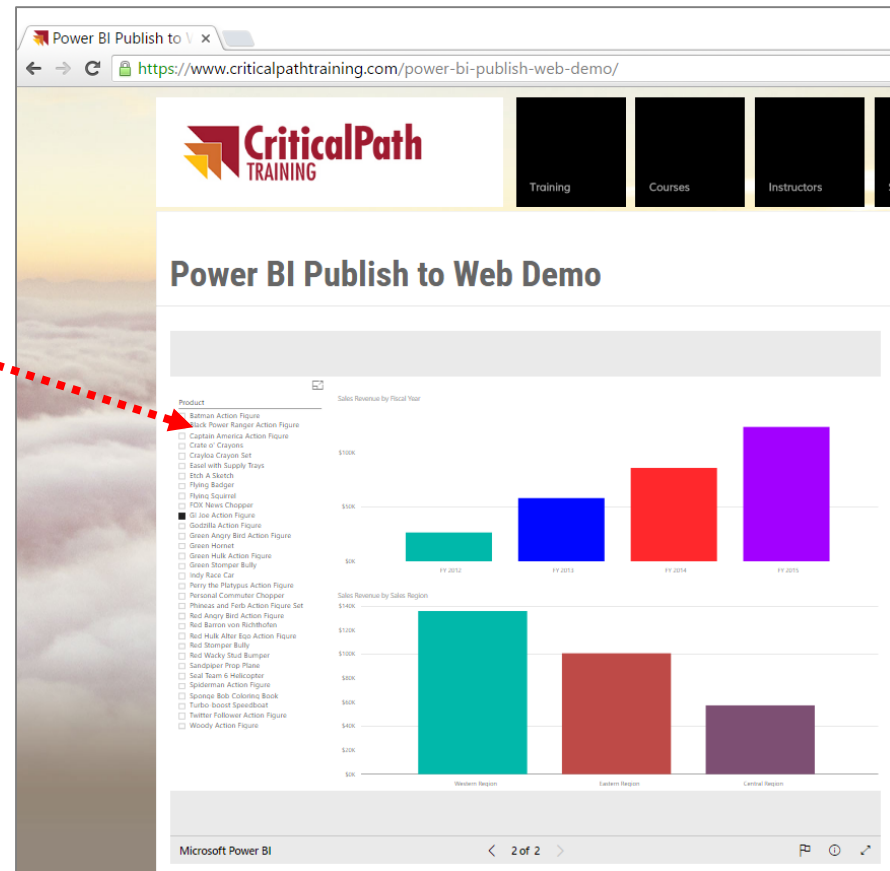
<https://app.powerbi.com/view?r=eyJrIjojYTM3YjlkNzctNWY5My00YTUyLl>

Html you can paste into your blog or website

`<iframe width="933" height="700" src="https://app.powerbi.com/view?r=eyJrIjojYTM3YjlkNzctNWY5My00YTUyLl" />`

Size

Close



Agenda

- ✓ Designing Interactive Reports
- ✓ Importing Custom Visuals
- ✓ Publishing a Project to the Power BI Service
- ✓ Using the Publish to Web Feature
- Publishing Reports in SharePoint Online
- Implementing Row-level Security (RLS)





DEMO

Using the Publish to SharePoint Feature

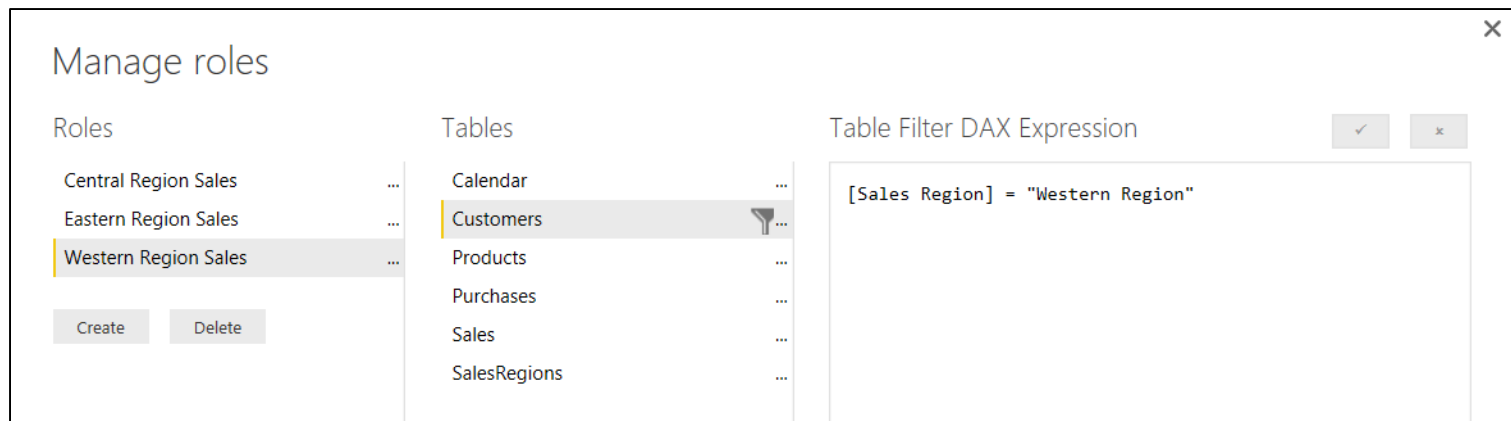
Agenda

- ✓ Designing Interactive Reports
- ✓ Importing Custom Visuals
- ✓ Publishing a Project to the Power BI Service
- ✓ Using the Publish to Web Feature
- ✓ Publishing Reports in SharePoint Online
- Implementing Row-level Security (RLS)



What Is Row-level Security (RLS)

- Security Scheme based on Named Roles
 - Roles are defined using Power BI Desktop
 - Each role is scoped to the dataset within a PBIX project
- Role defined using one or more DAX expressions
 - DAX expressions restrict which rows are accessible





DEMO

Configuring Row-level Security

Summary

- ✓ Designing Interactive Reports
- ✓ Importing Custom Visuals
- ✓ Publishing a Project to the Power BI Service
- ✓ Using the Publish to Web Feature
- ✓ Publishing Reports in SharePoint Online
- ✓ Implementing Row-level Security (RLS)

