# INTRODUCTION TO POWER BI

## TIPS

* **Axis Value – Display units**
* **Show Constant Line name - Data label – Style – Name**
* **Custom visuals - Microsoft AppSource**
* **Format column – Column Tools**
* **Navigate in - Drill Down and Drill Mode (form of cross filtering)**
  + **Two down arrows - move down to the next level of the hierarchy**
  + **Split arrows down - expand down to the next level of the hierarchy**
  + **Single down arrow - enable Drill Mode – selecting a level will drill down and expand all elements in that level**
* **Drill mode – form of cross filtering - Format – Edit interactions – Turn off interactions**
* **Row context iterator in Measures E.g. Orders Above Target Profit margin = SUMX(Orders,IF(Orders[Profit\_Margin]>Orders[Target Profit Margin],1,0)) – iterates over each row and count number of orders with profit margin above target profit margin**
* **Create a table as placeholder for measures – Home – Enter Data – Measures - \_Calculations – Delete measures Column**
* **Apply bin size - right-click Field - Choose New group – Choose Bin size**
* **SWITCH() – Create Column with DAX to categorize - evaluate expression against list of values and return one of multiple result expressions - used to avoid multiple nested IF statement**
* **Use Legends in charts to break out categories**
* **Nested DAX IF statements to Categorize 3 – E.g. Demographics = IF('Databel - Data'[Age]>=65, "Senior", (IF('Databel - Data'[Age]<30, "Under 30", "Other")))**
* **Nested DAX IF statements to Categorize 3 E.g. Grouped Consumption = IF('Databel - Data'[Avg Monthly GB Download]<5, "Less than 5 GB", (IF('Databel - Data'[Avg Monthly GB Download]> 10, "10 or more GB", "Between 5 and 10 GB")))**
* **Turn off Stepped Layout in Matrix**
* **Conditional formatting Table - Cell elements - choose column - enable for font colour - conditional formatting button- Add Rule**
* **Conditional formatting Visual – Format – Colours – fx**
* **Best Practice – Overview page canvas settings – 1640px width x 960px height**
* **Use gradient fx for bubbles**

## Getting Started with Power BI

**Star Schema (common database structure)**

* Fact tables – events or transactions
* Dimension tables - more information about each transaction
* Create relationships through key columns

**Fact table and Dimension tables**

* **Must have a main fact table**
* **And related dimension tables**

**3 Views (top left)**

1. Report (default) – create reports and visuals
2. Data – see associated data
3. Model – see and manage relationships among table in data model

**Report View**

* Canvas area – visualizations are created and arranged
* Filters pane – filter data visualizations
* Visualizations pane – add, change or customize visualizations
* Fields pane – shows available fields
* Can drag filters and visualizations panes to the canvas
* Change filters and axes
* Drag between X and Y axis to segment
* Search in Fields Pane
* Select graph and change in Viz Pane

**Slicers and Tables**

* Add additional fields to existing reports
* Slicer – automatically modifies all connected visualizations
* Use multiple filters and slicers
* Dimension table
* Match page name to title
* Format Slicer (filter) in Visualizations pane
* Chart titles are automatically generated
* Use eraser to clear selection
* **Filter types : Visual Level | Page Level | Report Level| Drill through**

## Transforming Data

**Dataset Errors**

* Unneeded columns
* Inconvenient, inconsistent data
* Extra characters
* Blank rows

**Cleaning Data**

* Transform before loading
* Power Query Editor (M language) - format dataset
* Opens in separate window – need to close and apply
* Applied steps tracks changes – can revert to earlier version
* Use first row as headers
* Edit query in Report view
* Power Bi automatically sums visualizations – can change aggregation in Visualizations pane

**Transforming and Formatting Data**

* Power Query Editor Menu options
* Replace data type / blanks
* Format in Column Tools
* Set Summarization

**Map Visualization**

* Set Data Category
* Map Visualization – Globe icon
* Set bubble size

## Visualizing Data

**Visualization Options – Specific data series values across different categories**

* Bar | Column charts – look at specific data series values across different categories
* Bar chart – vertical
* Column chart – horizontal
* Stacked Bar | Column chart – includes multiple elements - entire Bar | Column is the total
* Clustered Bar | Column chart – includes multiple elements located next to each other
* 100% Stacked Bar | Column chart – total of each stacked bar always equals 100%
* Combo chart – combines column chart + line chart
* Line chart – multiple lines in one chart – overall shape of entire value series, usually over time
* Area charts – based on line-charts with the area between the line and axis filles in
* Pie | Doughnut charts – relationships of parts to a whole
* Tree maps - relationships of parts to a whole

**Visualization Options – One, two or more values for showing the overall level of performance**

* Cards – single value
* Mulli-row cards – multiple values
* Gauge | KPI – actual data compared to budgeted data
* Table - detailed text data in tabular format – grid of related data in multiple columns
* Matrix - detailed text data in tabular format – can be expanded and collapsed by rows or columns

**Change and edit different visualizations**

* Click visualization and chose another visualization icon
* Change properties of visualizations - Formatting icon Paintbrush – above visualizations pane
* Fonts – under values
* Edit table column widths in the table

**Sorting and more formatting**

* Click on 3 dots for more options – sort / sort axis / sort by
* Modify in data view – select column – Sort by column options
* Change size – Formatting options – general – size
* Change slicer – format – visual – slicer styles
* **Change theme – View menu - similar to Office apps**

## Filtering

**Drilling up and down and expanding**

* Must have hierarchies
* Hover over visual to use icons in action bar or right-click
* **Drill down all fields at once** - Select double arrow drill-down goes to next level in hierarchy -
* Each step in the path shows new information
* To go up a level in the hierarchy - select the single upward arrow drill-up icon
* **Drill down one field at a time –** Select drill-down icon to turn it on - option of drilling-down one field at a time
* **Selecting a visual element without turning on drill-down option cross-filters other charts on report page rather than drilling down**
* Expand option adds another hierarchy level to current view
* **Expand all fields at once -** Confirm drill-down option on and select expand down icon - Discrete
* **Expand one field at a time -** Confirm drill-down option off and select expand down icon
* **Show the data as a table to look at how drilling works**
* **Navigate in Drill Down and Drill Mode (form of cross filtering)**
  + **Two down arrows - move down to the next level of the hierarchy**
  + **Split arrows down - expand down to the next level of the hierarchy**
  + **Single down arrow - enable Drill Mode – selecting a level will drill down and expand all elements in that level**
  + **Drill mode – form of cross filtering**

**Filtering**

* Display data based on selected criteria
* Filter out unnecessary information
* Levels of filters – Visualization level | Page level | Report level
* **Turn off filtering – Don’t allow end user to change filters – turn off interactions**
* **Select a Report and look at Interaction icons on other reports - Turn off so other reports do not change**
* **Select visualization - Format – Edit interactions – Turn off interactions**
* Use Filter Pane – drag values
* Possible to filter on single report or all pages
* Some Filter options – Top n
* Drag any Field to Filter pane

**Underlying data and hierarchies**

* 3 dots on visualization – show as table
* 3 dots – export data
* Create hierarchy – Data view or Report View – Right click column name – Create hierarchy
* Rename hierarchy – double click
* Add columns to hierarchy by right clicking on column names
* Edit hierarchy order – Model view - Properties pane
* Use hierarchy – Select visualization – change axis to new hierarchy
* New controls appear with drill down controls – turns black if enabled
* **Show Date without hierarchy – Values - Right click – Select Date instead of Date Hierarchy**

# INTRODUCTION TO DAX IN POWER BI

## Getting started with DAX

**DAX**

* Data Analysis Expressions
* Formula language to create calculations in columns, tables, measures
* Based on Excel
* Used on other Microsoft tools like Power Pivot and Analysis Services

**DAX functions**

* Predefined formulas – arguments
* Specific syntax – order of arguments
* Over 200 different functions in different categories

**Calculated Columns**

* Expands existing dataset – without editing source data
* Evaluates at row level
* Adds new column to existing table
* Calculated at data load – when data is refreshed

**Calculated Measures**

* Enabled complex calculations
* Aggregates multiple rows
* Adds new field that can be added to visualizations
* Calculated at query time – as interact and filter – not run every time the table is accessed
* Write a DAX formula or use Quick Measure tool

**Creating Calculated Columns and Measures**

* Report or Data view
* Select correct table
* New Column – Input formula
* **Create a table as placeholder for measures – Home – Enter Data – Measures - \_Calculations**
* New Measure - Input formula

## Context in DAX Formulas

**Introduction to Context**

* Enables dynamic analysis - results of a formula change to reflect the selected data
* There are three types of contexts - row, filter and query

**Introduction to Row Context**

* Row context - "the current row" – e.g. Calculated column
* Calculated column - context includes values from all columns within current row

**Row context can also be used in measures**

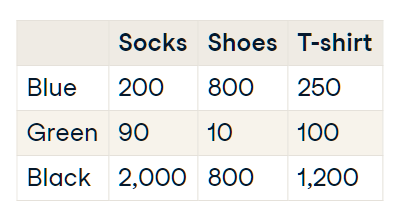
* **ONLY when using iterator functions**
* Loops through every row in a table and runs the formula for each row
* Iterator function - identified by “X” at the end of function name – e.g. SUMX
* Syntax: SUMX(<table>, <expression>)
* E.g. SUMX (Sales, Sales[Price] + (Sales[Price] \* Sales[Tax]) – Iterates over each row and performs the measure calculation without creating a Calculated Column
* E.g. Orders Above Target Profit margin = SUMX(Orders,IF(Orders[Profit\_Margin]>Orders[Target Profit Margin],1,0)) – iterates over each row and count number of orders with profit margin above target profit margin

**Introduction to Filter Context**

* Filter context - set of filters that have been applied before the calculation is carried out
* Applied in several ways, including attributes in a row or column, via a slider, through the filter pane, or in a calculated measure

|  |  |
| --- | --- |
| **Colour** | **Quantity** |
| Blue | 1,250 |
| Green | 200 |
| Black | 4,000 |

* E.g. Table of data - apply a filter of blue
* For Power BI to return a value for colour blue, it will go to the base table and apply a filter where colour is equal to blue.
* Can extend this further by creating a matrix of colour versus product category



* This changes the filter context again, because for each data point that is returned Power BI is applying filter context
* For Blue Socks - a filter is being applied for colour being equal to Blue and product category being equal to Socks

**CALCULATE Function**

* Calculate allows to evaluate an expression with one or more filter contexts
* Takes two arguments - one argument is required, and the other argument is optional
* **SYNTAX: CALCULATE(<expression> [, <filter> [, <filter2> [, …]]])**
* First argument contains the expression to be evaluated - this must return a single value
* Second argument for adding filters is optional - but need to ensure that filters evaluate as a table, and do not clash with one another
* E.g. Sales City is equal to London and Sales Country is not equal to United Kingdom
* Filters inside calculate function will always override any filters from visualization
* Example: Total sales in EMEA region - CALCULATE (SUM(Sales), Sales[Region] = “EMEA”)

**Variables**

* Simplify code
* Increase visibility and performance
* Create and reuse measure inside Calculate

## Working with Dates

**Working with dates**

* Example date: 2020/09/30 12:52
* **Date and Time Functions**
* YEAR (<date>) = 2020
* QUARTER (<date>) = 3
* MONTH (<date>) = 9
* **Format Functions**
* Weekday: FORMAT (<date>, <”dddd”>) = Friday
* Time: FORMAT (<date>, <”h:nn:ss”>) = “12:52:00”
* **Time Intelligence Functions**
* LASTDATE()
* DATESBETWEEN ()
* DATEADD ()
* Evaluate data in time-series
* **Time intelligence from only transactional tables – wrong results for gaps**

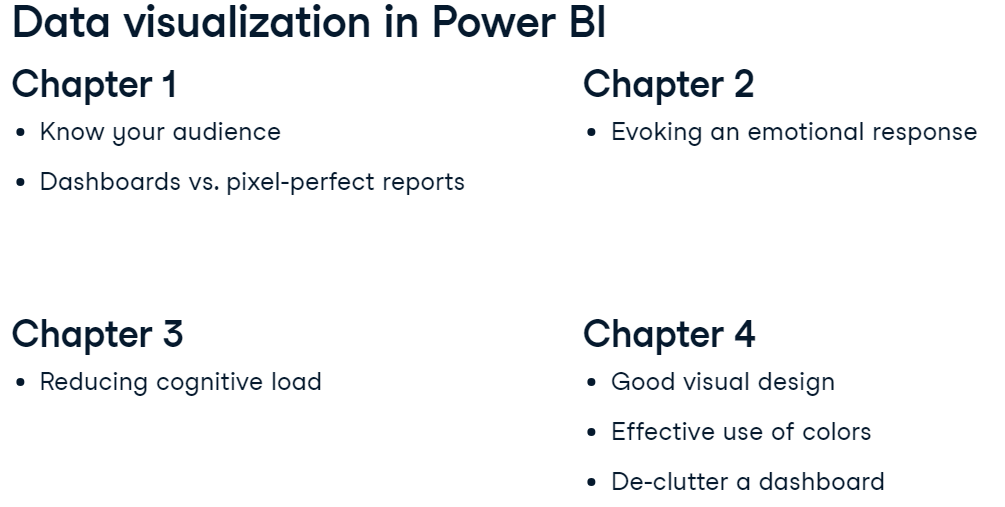
**Create dedicated Standalone Date Table**

* For accurate reporting with time-series and time-intelligence functions
* Filter multiple date attributes
* Custom calendar views
* Use time-intelligence functions
* Two ways to create Date table in DAX
* **CALENDAR ()**
* SYNTAX: CALENDAR (<start\_date>, <end\_date>)
* Dates = CALENDAR (MIN (Sales[OrderDate]), MAX(Sales[OrderDate]))
* Returns table with single continuous date column inclusive of start and end dates
* **CALENDARAUTO ()**
* SYNTAX: CALENDARAUTO (<fiscasl\_year\_end\_month>)
* Returns table with single continuous date column inclusive of earliest and latest dates in model
* E.g. CALENDARAUTO (12)
* fiscal\_year\_end\_month needs to be specified for the last fiscal month you after the last date
* E.g. If last date was '2020-07-27' and we specified 12, we'd see dates until the end of the year

**Date and Quick Measures**

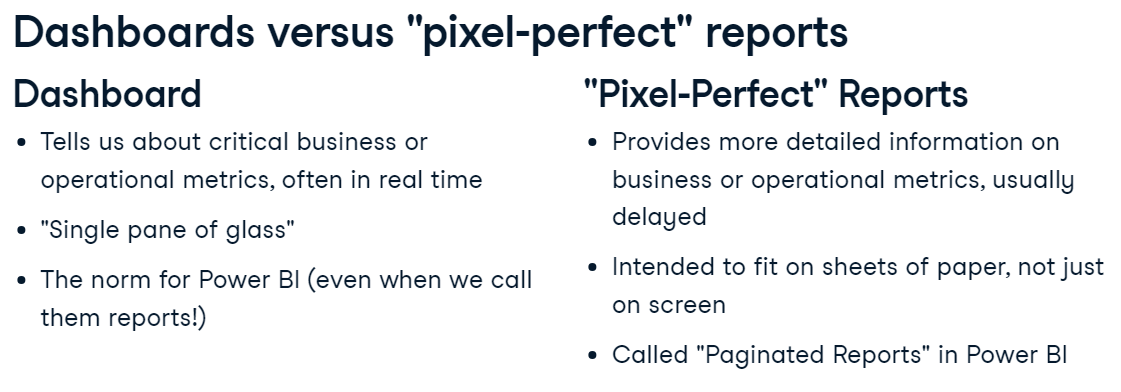
* For accurate reporting with time-series and time-intelligence functions
* Create new table – Create relationship
* Create new calculated column
* Quick Measures – create complex DAX functions by drag and drop
* E.g. Rolling Average –long term trend
* **3-year period = Periods after = 3**

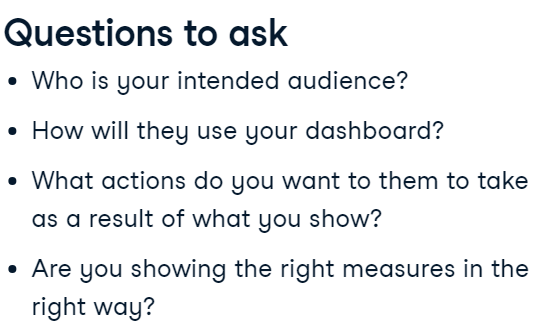
# DATA VISUALIZATION IN POWER BI



## The Audience is King

**Dashboards vs Pixel-Perfect Reports**

* Power BI offers two methods of showing data to users: **Dashboards | Pixel-Perfect Reports**
* 

**Intended audience**

## Getting an Emotional Response

**Getting an emotional response**

* Users are happy because everything looks fine
* Happy users don't always need to act, which makes them even happier
* Users are angry because something has gone wrong
* Angry users are angry in part because they now need to do something.
* Worst-case scenario is if a person has zero emotional response - that's a sign that the dashboard is not relevant to the user

**Bar and Column Charts**

* Currently, three types of column charts are available in Power BI
* **Clustered bar charts - easy to show variety of data over single period of time**
* Include appropriate titles and axis values
* Use Format (Painter) options
* Use Analytics menu lines
* **Stacked bar chart – easy to see aggregates**
* **100% stacked bar chart – easy to compare proportions**
* Use legends to break out categories
* Use page level filters
* Use small multiples fields
* Use show as table (right click) to clarify more data
* Use Small Multiples field to segment and analyse
* **Tip – Axis Value – Display units**
* **Tip – show Constant Line name - Data label – Style – Name**

**Small multiples**

* Show comparison information across two categorical dimensions
* Saves space
* No need for individual charts

## Reduce Cognitive Load

**Reducing cognitive load**

* Tailor page to audience
* Focus one story per page
* Balance between information-rich visuals vs confusing audience
* Use custom visuals

**Line and Area charts**

* Line charts – Easy to track historical changes in small number of features

**Combination charts**

* Combination chart - Easy to compare a rate variable and a counting variable over time

**Tornado Chart**

* Tornado chart – easy to compare two values for each category

## Less is More

**3 Keys to good visual design**

* Does this contribute to the story?
* Is this the right visual element?
* Is it necessary?
* Less control as user can use filters and slicers
* Static text is not useful
* Colour- ‘Pre-attentive attribute’ emphasis
* Negative space (without any information or visual effect) – easy to differentiate sections

**Shares- Pie charts | Tree maps**

* Pie chart – easy to show simple shares of static total
* Tree map – easy to show shares hierarchical categorical data
* Tree map - Use data labels
* Tree map - Add more categories to category to create hierarchy
* Navigate in - Drill Down and Drill Mode (form of cross filtering)
* Two down arrows - move down to the next level of the hierarchy
* Split arrows down - expand down to the next level of the hierarchy
* Single down arrow - enable Drill Mode – selecting a level will drill down and expand all elements in that level

**Gauge**

* Gauge – easy to show current value with target value and maximum value
* Create Measures in DAX for target and maximum values

**Card**

* Card - easy to show single value – clean – important metrics

**Key Performance Indicators (KPIs)**

* Track performance over time vs expectations
* KPI visual
* Set target as measure
* Format - Set trend axis direction E.g. Low is good

**Conditional formatting**

* Conditional formatting – easy to display colour when certain conditions are met
* Useful when a condition is particularly good or bad – colour draws the eye
* E.g. change font colour in table for any values more than xxx
* Table - Cell elements - choose column - enable for font colour - conditional formatting button- Add Rule
* Visual – Format – Colours – fx
* Use Max, Min

# CASE STUDY: ANALYZING CUSTOMER CHURN IN POWER BI

## Exploratory Analysis

**Data analysis flow in Power BI**

* Five different steps in the data analysis flow

1. Data check - to make sure the data makes sense and is ready to work with
   1. E.g. check for duplicate values or missing values
   2. sense check with other internal data sources.
2. Explore Data - ask different questions
   1. E.g. Does an increase in revenue also lead to an increase in profit?
   2. Build your first visualizations in this step.
3. Analyze & Visualize Data - key to choose the right visualization to convey a message
   1. also enables to dig deeper into certain topics to make sure not missed any insights
4. Dashboarding - portray analysis clearly in one or more dashboards
5. Communicate Insights - with stakeholders

**Case Study**

* The problem is customer churn.
* Telecom provider Databel.
* Analyze why customers are churning (leaving Databel)
* Analyze a snapshot of the database at a specific moment in time- no time dimension

**Defining churn (Investopedia)**

* Churn rate / Rate of attrition / Customer churn
* **Rate at which customers stop doing business with an entity**
* Compare churn with the leaky bucket problem
* Can fill with more water (new customers)
* But overall revenue won't increase if existing customers are leaving
* Easier to retain customers than to attract new customers
* For many companies it's a priority to reduce churn

**Calculating churn**

* Simplified formula for churn
* **Divide customers lost by the total number of customers**
* E.g. If total 100 customers in certain period and 10 end up leaving - churn rate = 10%
* Multiple methods to calculate churn
* Makes sense for company to slightly alter formula depending on the industry
* E.g. A traditional e-commerce platform could consider a certain customer a churner if they haven’t made a purchase in last 12 months

**The Databel dataset**

* 29 different columns
* 1 row per customer
* Customer\_id - unique ID identifies individual customer
* Churn Label - indicates if customer churned with "Yes" and "No" labels
* Contains various other dimensions – i.e. demographic fields, premium plans
* Measures
  + Total Charges column - sum of all monthly charges billed to a customer
  + Other measures – use Metadata sheet

## Investigating Churn Patterns

**Analyzing the demographics of Databel**

* Insights discovered so far
* Average churn rate is around 27%
* Main reason why customers churn is related to competitors
* This could raise questions such as "Is Databel competitive enough?"
* Churn rate in California is abnormally high at 63.24%,
* Don’t have a clear explanation yet for the relatively high churn rate of 27%,
* Still so many columns o analyze

**Metadata sheet**

* Holistic analysis plan
* 29 columns in the database
* Good practice to have a structured approach
* Grouped in different categories - create different pages to analyze the different topics

**Advice to Databel**

* Discovered that churn rate for customers who pay for an international plan but don't call internationally is sky-high
* Contact customers who are on an international plan but have not called internationally and propose they downgrade their plan.

## Visualizing your Analysis

**Creating a cohesive story**

* Build a series of different dashboard-style pages which in combination will tell a story
* Build the pages in such a way to enable easy communication of insights
* Unable to build dashboards in Power BI Desktop - available only in Power BI Service
* Try and create a similar experience for users within Power BI Desktop report

**Structuring analysis**

* Not easy to structure an analysis piece in a cohesive way
* It's not informative to publish a bunch of visualizations stored on different pages
* Need to combine the information in such a way that it fits well together
* Ideally, show all discovered insights in a few pages

**Create narrative for stakeholders**

* Overview dashboard or page - DataCamp colour scheme - main insights from analysis
* Interactive - revealing even deeper insights for stakeholders
* Contains main KPIs - Number of customers, number of churned customers, churn rate

**Interactivity makes a report powerful**

* End user clicks on Month-to-Month - churn rate goes up to 46%
* Almost 1600 out of 1800 churners come from this groop
* End user selects the Two-Year contract - churn rate drops to 2.78%

**Overview Page**

* Collate information for best possible insights into key concerns
* Building dashboard-style pages - makes sense to build new graphs
* **Best Practice – Overview page canvas settings – 1640px width x 960px height**

**SWITCH () DAX Function**

* Create new column with DAX SWITCH() to categorize
* Evaluates expression against list of values and returns one of multiple result expressions
* Used to avoid multiple nested IF statements
* SYNTAX: SWITCH(<expression>, <value>, <result>[, <value>, <result>]…[, <else>])
* E.g.
* 