**Course 5 - Analyse Data to Answer Questions**

- **Analysis** - the process used to make sense of the data collected
    - Goal - to identify trends and relationships within data so you can accurately answer the question you're asking
- 4 phases:
    - **Organise data** - collecting the data you need
    - **Format and adjust data** - streamlines things and saves time (Filter, sort)
        - **Sorting** - arranging data into a meaningful order
        - **Filtering** - when you are only interested in seeing data that meets a specific criteria
    - **Get input from others** - soliciting info from others
    - **Transform data** - identifying relationships and patterns between the data and making calculations
- Organising your data into tables help make decisions about data types
    - Help figure out what variables you need and the data type those variables should have
    - Have distinct categories and classifications lets you focus and differentiate your data
- **Sort sheet** - all of the data in a spreadsheet is sorted by the ranking of a specific sorted column - data across rows is kept together
- **Sort range** - nothing else on the spreadsheet is rearranged besides the specified cells in a column - doesn't keep the rows together, data is jumbled
- Sorting with a menu - select column - data -sort range or sheet
- Using SORT function eg. **=SORT(A2:D6,2,TRUE)**
    - **(first cell which data is collected from:last cell you want included, what we're sorting by [if it's B, use 2 - the function doesn't recognize letters],** T**RUE is in ascending**, **False is in descending**)
- **Customised sort order** - when you sort data in a spreadsheet using multiple conditions
    - Highlight all data you want to sort - data tab - sort range - data has a header row - add another sort column
-
- SQL - asterisk selects all columns in data table, need to use single or double quotations if something is in string format, capitalization matters
    - Can pull and combine different data tables
- **ORDER BY** Release_Date DESC (example)
    - (usually last in Query), **column you want to sort, default is ascending order, specify DESC if you want most recent to oldest (no comma)**
    - **SELECT** *
      **FROM** `course-5-373900.movie_data.movies`
      **WHERE** Genre = 'Comedy'
      **AND** Revenue > 300000000
      **ORDER BY** Release_Date **DESC**

- Week 1 Practice Quiz - Saving a subset of data from previous entry - Save Results - BigQuery Table

- Formatting on Spreadsheet - Format - Number\
    - **=CONVERT(**B2, **“**F**”,”**C**”)** - converts Fahrenheit to Celcius
    - Copy and paste new format just as values to avoid confusion - copy, paste special, values only
    - **=CONVERT**(D2, "mph", "m/s") - converts miles per hour to meters per second
        - **(cell you want to convert, what it is originally, what you want to change it to)**
- **=CONCAT(A2,B2)** - merge two values without a space
- **=CONCATENATE**(A2,” “,B2) - need to use this function if want to put space
    - =CONCATENATE(C2," ",D2,**", ",**E2) - to get a date like April 30, 1989
- **Data validation (function)** - allows you to control what can and can’t be entered in your worksheet
    - Used to add drop-down lists to cells with predetermined options for users to choose from
        - **Data - data validation - add rule - drop down** - input into options
    - Create custom checkboxes
        - **Data - data validation - tick box**
    - Protect structured data and formulas
        - **Choose “Reject The input” if data is invalid**
- **Conditional formatting** - spreadsheet tool that changes how cells appear when values meet specific conditions
    - **Format - conditional formatting - if date is - after today**
    - Gives you and easy to understand visual cue

**Converting Data in SQL**

- **CAST (expression AS typename)**
    - **Expression** - the data to be converted
    - **Typename** - data type to be returned

```
SELECT CAST (MyDate AS DATETIME) FROM MyTable
```

In the above SQL statement, the following occurs:

- **SELECT** indicates that you will be selecting data from a table
- **CAST** indicates that you will be converting the data you select to a different data type
- **AS** comes before and identifies the data type which you are casting to
- **DATETIME** indicates that you are converting the data to a datetime value
- **FROM** indicates which table you are selecting the data from

- **COERCION** (works with bigger numbers)
- **UNIX_DATE** - returns the number of days that have passed since Jan 1 1970
- **SAFE_CAST** - return a value of Null instead of an error when a query fails

```
SELECT SAFE_CAST(MyDate AS STRING) FROM MyTable
```

- Using CONCAT to combine strings from multiple tables to create new strings
  `CONCAT(street_address, " to ", unit)`
- From statement - use backtick ` and not apostrophe ‘

**Merging multiple sources in SQL**

- Finding out what routes are most popular with different user types with
  - Create strings of recognizable route names that we can count and sort:

```
SELECT usertype,
```

- want user type as a column

```
CONCAT(start_station_name, 'to', end_station_name) AS route,
```

- combine names of beg station and end station of each trip in new column as route

```
COUNT(*) as num_trips,
```

- Each row represents a trip, need to count every trip so use *

```
ROUND(AVG(cast(tripduration as int64)/60,2) AS duration
```

- Avg trip duration for each route
- Don’t need exact so round up
- Int64 - want data to be in integer form so use cast to change
  - 64 - big query stores numbers in a 64-bit memory system
- Divide by 60 so it returns output duration in minutes instated of seconds
  - to 2 decimal places

```
FROM `bigquery-public-data.new_york_citibike.citibike_trips`

GROUP BY start_station_name, end_station_name, usertype
```

- When using COUNT and AVG, need to use **GROUP BY** to group tgt summary rows

```
ORDER BY num_trips DESC
```

- Tell how we want to organise this data

```
LIMIT 10
```

- only want top 10

| Row | usertype | route | num_trips | duration |
|---|---|---|---|---|
| 1 | Customer | Central Park S & 6 Ave to Central Park S & 6 Ave | 40009 | 50.89 |
| 2 | Customer | Grand Army Plaza & Central Park S to Grand Army Plaza & Central Park S | 15234 | 52.99 |
| 3 | Customer | Centre St & Chambers St to Centre St & Chambers St | 12466 | 35.65 |
| 4 | Subscriber | W 21 St & 6 Ave to 9 Ave & W 22 St | 11594 | 5.35 |
| 5 | Customer | Broadway & W 60 St to Broadway & W 60 St | 10816 | 52.36 |
| 6 | Subscriber | W 21 St & 6 Ave to W 22 St & 10 Ave | 10451 | 6.95 |

**Strings in Spreadsheets**

- **Len** - shows length of the string
  - Case sensitive
- **FIND** - to locate position of string
  - Eg. =FIND(“ “,C3) - to find where the space is (where the string ends)
- **= Right**(J10,4) - return from the right
- **=LEFT**(D3,10) - return from the left

| Function | Usage | Example |
|---|---|---|
| CONCAT | A function that adds strings together to create new text strings that can be used as unique keys | CONCAT (‘Google’, ‘.com’); |
| CONCAT_WS | A function that adds two or more strings together with a separator | CONCAT_WS (‘ . ’, ‘www’, ‘google’, ‘com’)<br><br>*The separator (being the period) gets input before and after Google when you run the SQL function |
| CONCAT with + | Adds two or more strings together using the + operator | ‘Google’ + ‘.com’ |

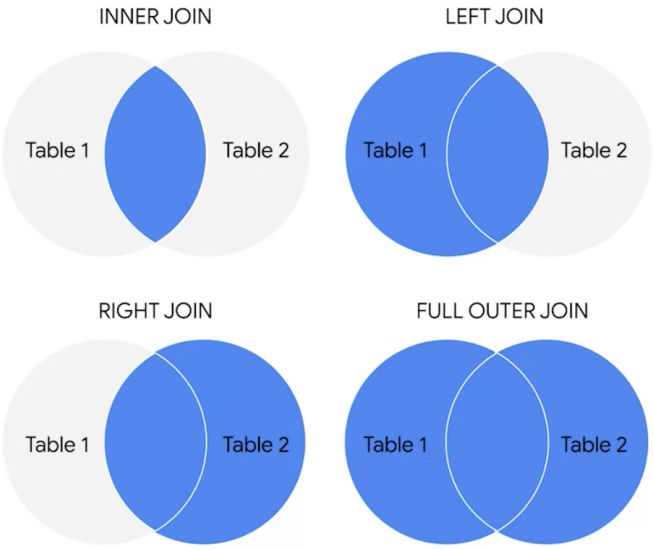
- **CONVERT** - change the unit of measurement
- **JOIN -** combine rows from two or more tables based on a related column
- **= IF**(end>start, end-start, 1-start+end)
  - If end is greater than start, replace the standard end time minus start with 1 minus start + end time
  - Calculating difference between bike rides if elapsed time crosses into the next day
- **Best practices for searching online**
  - Thinking skills, data analytics term, basic knowledge of tools

**Week 3**

- **Aggregation** - collecting or gathering many separate pieces into a whole
- **Data aggregation** - process of gathering data from multiple sources in order to combine it into a single summarised collection
    - Puzzle pieces = data
    - Organisation = aggregation
    - Piles of pieces = summary
    - Putting the pieces together = gaining insights
- **VLOOKUP** - a function that searches for a certain value in a column to return a corresponding piece of info
    - Useful when populating data in a spreadsheet, merging data from one spreadsheet with data in another
    - VLOOKUP(“value to search for”, ‘sheet’, range, which column in number value, False - looking for an exact match, or true - looking for approximate matches)
    - Limitations - can only return a value from the data to the right , version control issues
        - Eg =VLOOKUP("Brazil", A2:B10, 2, false)
          - the population column is to the right of the country column
    - **Locking data** -> data -> protected sheets and ranges -> add
- **= Value** function converts a text string that represents a number to a numerical value
- Troubleshooting questions
    - How should I prioritise these issues
    - In a single sentence, what’s the issue i’m facing
    - What resources can help me solve the problem
    - How can i stop this problem from happening in the future
- **Absolute reference** - a reference that is locked so that rows and columns won't change when copied
- **MATCH** - a function used to locate the position of a lookup value
- **=PRODUCT(I15,J15)** - calculates total pay, first cell - hours, second - pay rate

**Week 3**

- **JOIN** - a SQL clause that is used to combine rows from two or more tables based on a related column (SQL version of VLOOKUP)
- Primary keys reference columns in which each value is unique to that table
- Foreign keys are primary keys in other tables
- 4 common joins:




- 
  - **Inner join** - a function that returns records with matching values in both tables (key values in both tables)
    - JOIN function defaults to inner join without specifying
  - **Left join** - function that will return all the records from the left table and only the matching records from the right
    - The table mentioned first is left, table mentioned second is right
  - **Right join** - function that will return all the records from the right table and only the matching records from the left
  - **Outer join** - function that combines right and left join to return all matching records in both tables
  - Use **aliases** to simplify table or column names

```
- SELECT
- employees.name as employee_name,
- employees.role as employee_role,
- departments.name as department_name
- FROM `course-5-373900.employee_data.employees` as employees
- full outer join
- employee_data.departments on
- employees.department_id = departments.department_id
```

- **From** ‘dataset.table1’
- **Full outer join** Dataset.table 2
- **ON** - identifies how the tables are to be matched for the correct info to be combined
  - Tablename.column = table2name.column

- Eg. 2
  SELECT Shippers.ShipperName,COUNT(Orders.OrderID) AS NumberOfOrders
  FROM Orders
  LEFT JOIN Shippers
  ***- From First table, Leftjoin Second table***

```
ON Orders.ShipperID=Shippers.ShipperID
GROUP BY ShipperName;
```

- **COUNT* -** returns the number of rows in a specified range
- **COUNT -** returns the number of values (null not included) of the specified column
- **COUNT DISTINCT -** only returns the distinct values in a specified range (no repeating values)
- **Subquery** - SQL query that is nested inside a larger query
    - Inner query/ inner select - executed **first**
    - Usually in from or where clauses
    - Can only reference one column per subquery
    - Must be enclosed in brackets
    - Can't be nested in SET common because it is used with UPDATE to adjust specified columns and values in a table
    - To return more than one row, subqueries must contain multiple value operators such as IN
- **Outer select/ outer query** Statement containing the subquery
  Using subqueries to aggregate data:
- **HAVING** - allows you to add a filter to your query instead of the underlying table that can only be used with aggregate functions
    - **Aggregate functions** - perform calculations on one or more values and return a single value (SUM, AVG, COUNT)
    - WHERE cannot be used with aggregate functions

```
SELECT Employees.LastName, COUNT(Orders.OrderID) AS
NumberOfOrders FROM (Orders
INNER JOIN Employees
ON Orders.EmployeeID=Employees.EmployeeID)
GROUP BY LastName
HAVING COUNT(Orders.OrderID) > 10;
```

- **CASE** - returns records with your conditions by allowing you to include if/then statements in your query
- **IF -** function that returns a value if a condition is TRUE, or another if a condition is FALSE
    - Eg. SELECT OrderID, Quantity, IF(Quantity>10, "MORE", "LESS") FROM OrderDetails;
        - The results will return MORE if quantity was more than 10, LESS if quantity was las than 10
- **GROUP BY** - aggregate functions need this statement to group the result by one or more columns
    - Groups rows that have the same values from a table into summary rows
    - Comes at the end of query after WHERE

- Example - need to calculate what % of the orders are fulfilled by each warehouse(need to know which warehouses are delivering the most orders)
- We want all the data from Warehouse data even if it doesn't show up in the Orders table*****VERY CONFUSED, NEED TO CIRCLE BACK

**Week 4 - Calculations**

- Growth from previous year = current sales - last year sales
- %growth = growth from previous year/last year sales
- Conditional formatting - use colour scale to highlight which numbers are higher, lower, middle
- **Conditional functions** - perform a specific task, but only on cells that satisfy some defined criteria:
- **COUNTIF (range, “value”)**
    - Eg. =COUNTIF(B3:B50, “=1”) - count only if the string equals 1
- **SUMIF(range,criteria, sum_range)**
    - Eg. =SUMIF(B3:B50,">1",C3:C50) - add C values, only if the corresponding B strings are greater than one
- **SUMIFS(sum_range, criteria_range1, Criterion1, critera_range2, Criterion2,...)**
    - SUMIFS are similar but they can include multiple conditions
    - Eg. =SUMIFS(B1:B9,A1:A9, “Fuel”, C1:C9, “12/15/2020”)
        - Add values in B if A is Fuel and C is 12/15/2020

| | A | B | C |
|---|---|---|---|
| 1 | Expense | Price | Date |
| 2 | Fuel | $48.00 | 12/14/2020 |
| 3 | Food | $12.34 | 12/14/2020 |
| 4 | Taxi | $21.57 | 12/14/2020 |
| 5 | Coffee | $2.50 | 12/15/2020 |
| 6 | Fuel | $36.00 | 12/15/2020 |
| 7 | Taxi | $15.88 | 12/15/2020 |
| 8 | Coffee | $4.15 | 12/15/2020 |
| 9 | Food | $6.75 | 12/15/2020 |

- **=AVERAGEIF(range, criteria, [sum_range])**
    - =AVERAGEIF(B2:B21, "NY", D2:D21)
- **=MAXIFS(max_range, range1, criteria1, [range2], [criteria2], ...)**
    - Max rang - the array over which you are finding the max
    - Range - array you are checking
    - Criteria you are checking for
    - Order matters!
    - =MAXIFS(D2:D21, B2:B21, "NY")
        - D column was sales, B column was the states
    - =MAXIFS(D2:D21, B2:B21, "NY", E2:E21, "<400")
        - E was max item value (find the max sales for clients in NY where the price per item was less than $400)

- **SUMPRODUCT** - a function that multiples arrays and returns the sum of those products
    - =sumproduct(array1,array2…)
      =SUMPRODUCT(B3:B7,C3:C7) B was quantity, C was unit price = total rev
    - **Array** - a collection of values in cells (not the cells themselves
- **Profit margin** - a percentage that indicates how many cents of profit has been generated for each dollar of sale
    - If margin is 20%, you are earning 20 cents/dollar
    - =SUMPRODUCT(B3:B7,C3:C7,D3:D7)D was margin% = profit margin

- **Pivot tables** - lets you view data in multiple ways to find insights and trends
    - Cleans, organizes data and do calculations
    - **Rows** - organizes and groups data you select horizontally
    - **Columns** - organizes and displays values from your data vertically
    - **Values** - used to calculate and count data - where you input the variables you want to measure
    - **Calculated field** - a new field within a pivot table that carries out certain calculations based on the values of other fields (eg. finding an avg value)
    - **Filters** can also be added
- Adding in SQL:
    - Select columnA, columnB,
      columnA + columnB AS columnX
    - FROM table_name
- **Modulo** - an operator (%) that returns the remainder when one number is divided by another
- <> or != means not equals to
- SAFE_DIVIDE - avoids error of dividing by 0
- **Underscores** are used to connect text characters to avoid using spaces which can confuse certain servers and applications
    - Avoids potential issues while keeping the names readable

- **EXTRACT command -** allows you pull one part of a given date to use
    - EXTRACT(Year FROM starttime) AS year,
      Count(*) As number_of_rides
      From table_name
      GROUP BY year
      ORDER BY year DESC
      (default is ascending order)

- **Data validation** - rechecking the quality of your data helps ensure the data is complete, accurate, secure and consistent
    - method - if checking totals column is accurate, make the calculation and use SQL to see if they equal each other
    - WHERE total != total calc, if nothing returns, then the values are correct
- **6 types of data validation**
    - **Data type** - check that the data matches the data type (data values for school grades 1-12 must be numeric)

- **Data range -** check that data falls within an acceptable range (data values for school grade must be between 1-12)
- **Data constraints** - check the data meets certain conditions like attributes of the field, number of characters (school grades 1-12 must be whole numbers)
- **Data consistency** - check that data makes sense in the context of other related data ( shipping dates should not be earlier than production dates)
- **Data structure** - check that the data follows or conforms to a set of structure (web pages must follow a prescribed structure to be displayed properly vs MP3 files)
- **Code validation** - check that the app code systematically performs any of the previously mentioned validations during user data input (common problems - more than one data type allowed, data range checking not done)

- **Temporary tables** - a databases table that is crated and exists temporarily on a database server
    - Useful when you have a lot of tables you're performing calculations on at the same time, if you have a query that needs to join 7-8 of them, you could join the tables with the feast number of rows and store their output in a temp table
        - Then join temp table to other bigger tables
    - when you have lots of diff databases you're running queries on, run initial queries in separate database, then use temp table to collect the results of all these queries
        - Final report would run on the temp table
    - when you have a larger number of records in a table and you only need to work with a subset of them - avoid filtering the data over and over
    - They are automatically deleted from the database when you end SQL session
    - Can be used as holding are for storing values - **pre-processing** data
    - Can be used for **data Staging** - collecting results of multiple, separate queries - useful when you need to perform a query on the collected data or merge the collected data
    - Can store a filtered subset of the database
- **WITH** clause - type of temporary table that you can query from multiple times
    - Approximates a temporary table without adding a table to the database
    - **WITH**
      New_table_name AS(
      **SELECT ***
      **FROM**
      existing_table
      **WHERE**
      Condition to filter
      )
      - the AS instructs the database to put all of the data identified in the next part into the new table
      - the () is a subquery with regular select,from, where clause
    - **WITH** trips_over_1_hr as (
      SELECT *

FROM table
WHERE
Tripduration >= 60
)

- ## Count how many trips are over 60 min long
  Select
  Count(*) As cnt
  From trips_over_i_hr
  - can keep running queries on this temp table over and over as long as we're analyzing bike trips over 60 mins

- Temp tables are less complicated and easier to understand

```sql
WITH
  longest_used_bike as (
    SELECT
    bikeid,
    SUM(duration_minutes) AS trip_duration
    FROM `bigquery-public-data.austin_bikeshare.bikeshare_trips`
GROUP BY
  bikeid
ORDER BY
  trip_duration DESC
  LIMIT 1
  )

## find the station at which longest bikeshare ride started
SELECT
  trips.start_station_id,
  COUNT(*) AS trip_ct
FROM
  longest_used_bike AS longest
INNER JOIN
  `bigquery-public-data.austin_bikeshare.bikeshare_trips` AS trips
ON longest.bikeid = trips.bikeid
GROUP BY
  trips.start_station_id
ORDER BY
  trip_ct DESC
LIMIT 1
```

Press Alt+F1 for ac

- JOIN was used to find something specific from the original dat set

- **SELECT INTO -** temp table clause that copies data from one table into a new table, but doesn't add the new table to the database - useful when you want to make a copy of a table with a specific condition (like WHERE)
  - Better for one person usage

- Bigquery doesn't currently recognize this clause

```
SELECT
*
INTO
AfricaSales
FROM
GlobalSales
WHERE
Region = "Africa"
```

  - **INTO -** tells database to store data in a new temp table

- **CREATE TABLE** - temp table clause is good to use when several people need to access the same temp table, this adds the table into the database
  - The two methods above are tables that the database is responsible for managing
  - This method is managed by the user - after working with this table, you would delete or drop it from the database at end of session
  - Useful for tables that are more complex, like if the code's difficult to replicate

```
CREATE TABLE table_name (
      column1 datatype,
      column2 datatype,
      column3 datatype,
   ....
)
```

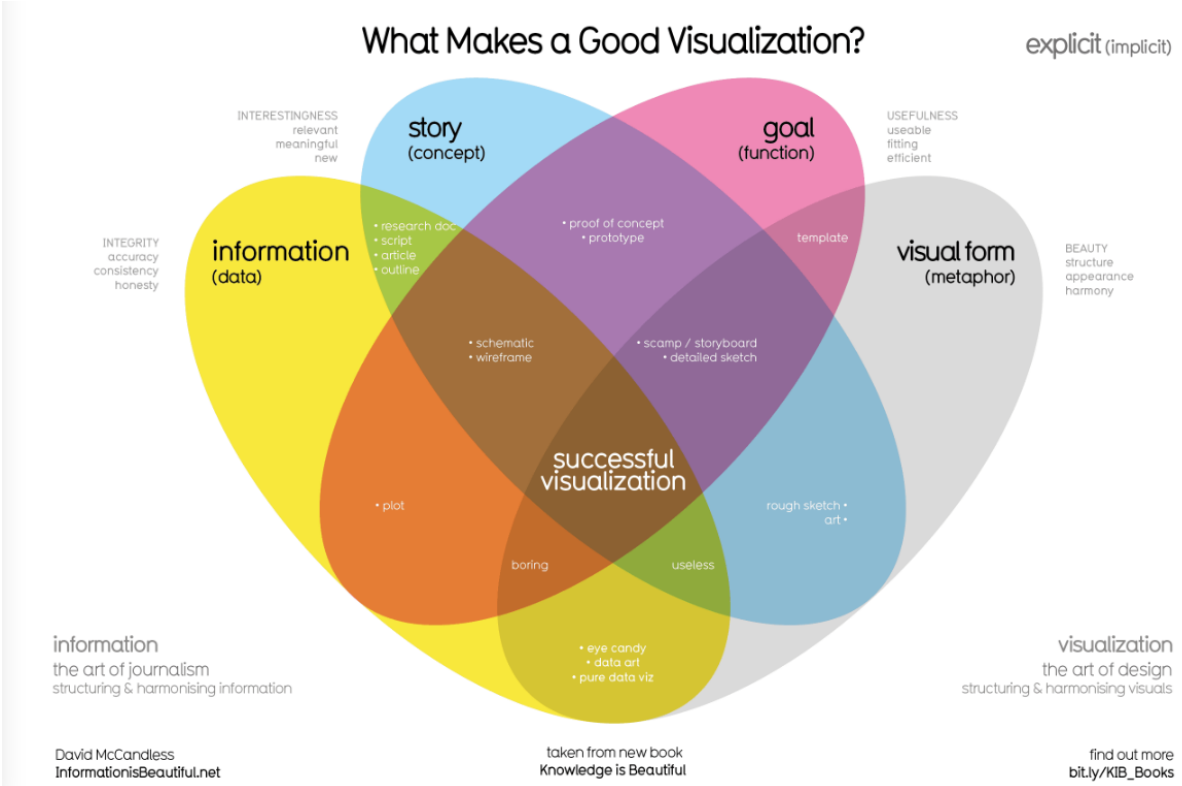
After you have completed working with your temporary table, you can remove the table from the database using the **DROP TABLE** clause. The general syntax is as follows:

```
DROP TABLE table_name
```

- *RDBMS - Relational Database Management System*
- Best practices with temp tables:
  - **Global** temp tables are made available to all database users, deleted when connections that use them have closed

- **Local** temp tables are made available only to the user whose query established the temp table, if you have created it and no one else needs it, you can drop it
- **Dropping a table** - diff from deleting a temp table; dropping removes the info contained in the rows of the table but also removes the table variable definitions(columns) themselves
    - Deleting a temp table removes the rows of the table but leaves the columns ready to be used again

## Course 6 - Share Data Through the Art of Visualisation

- **Data visualization -** graphic representation and presentation of data
- Looking at visuals to understand and draw conclusions about data
- Creating visuals using raw data to tell a story
- Quick rule - your audience should know exactly what they're looking at within the first 5 secs
    - 5 secs after that, they should understand the conclusion your visualization is making
- Successful visualization need all 4 - **McCandless Method**
    - **info(data)** - building block for your data
    - **story** (**concept) -** adds meaning, inspiration to the info
    - **goal(function) -** makes the data useful and usable, generate insights
    - **visual form (metaphor)** - creates beauty and structure
- 

    - Visual without goal, story or data could be just art
    - Data plus visual without goal or function is eye candy
    - Data with a goal but no story or visual is boring

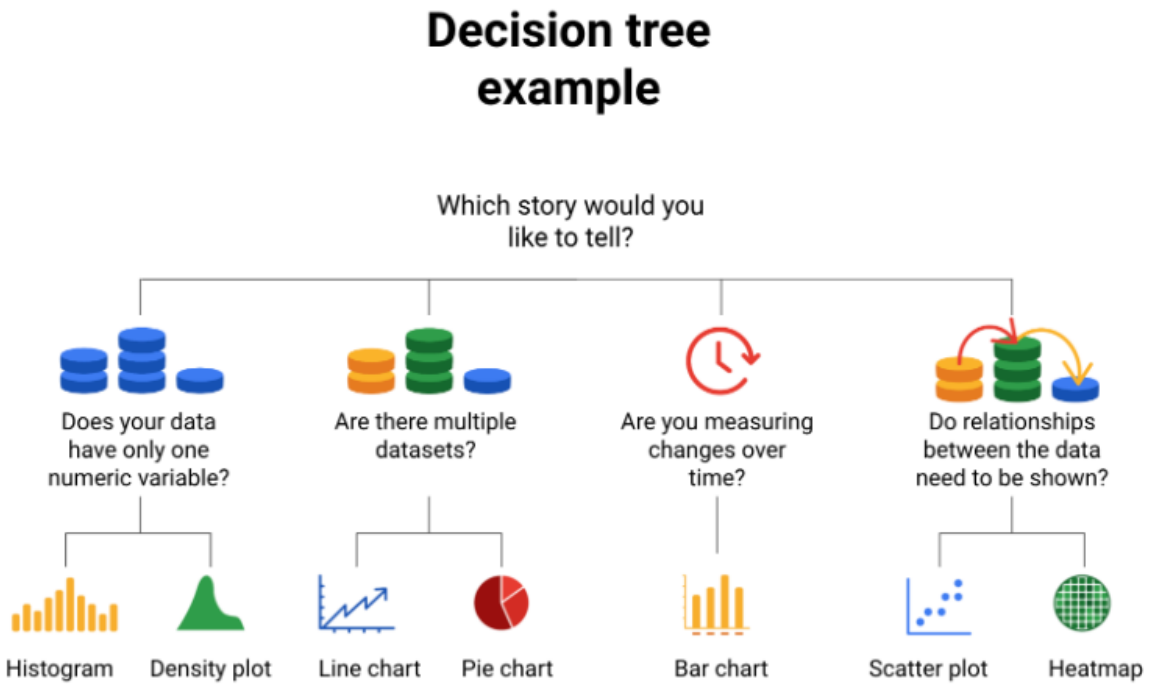
- **Kaiser Fung's Junk Charts Trifecta Checkup**
    - What is the practical question, what does the data say, what does the visual say
- **pre-attentive attributes -** the elements of a data visualization that people recognize automatically without conscious effort
- Basic building blocks that make visuals immediately understandable are called marks and channels
    - **Marks** - basic visual objects - points, lines, shapes; 4 qualities:
        - **Position** - where a specific marks is in space in relation to a scale or other marks
        - **Size**
        - **Shape** - whether a specific objects is given a shape that communicate something about it
        - **Color**
    - **Channels** - visual aspects or variables that represent characteristics of the data; basically marks that have been used to visualize data; how effective they are at communicating will be based on 3 elements:
        - **Accuracy** - are the channels helpful in accurately estimating the values being represented
        - **Popout** - how easy it is to distinguish certain values from others (length, size, line, width,shape, enclosure, hue, intensity)
        - **Grouping** - how good is a channel at communicating groups that exist in the data (proximity, similarity, enclosure, connectedness, continuity)
- **Bar graphs/column charts** - uses size contrast to compare two or more values
    - X-axis - represent categories, time periods, etc
    - Y-axis - scale of values
    - Ideal for comparing similar data side by side
- **Line graphs** - helps audience understand shifts or changes in your data; usually used to track changes in time, trends
    - Better for showing smaller changes
    - Can compare changes for more than one group
- **Pie chart** - show how much each part of something makes up the whole
- **Histogram** - like a bar graph, shows how often data values fall into certain ranges
    - Good for ranking by value to show ascending/descending order
    - Categorizing on x-axis by distinct numeric values
    - Ideal for comparing the distribution of two variables by individual grouping
- **Heatmaps** - use color to compare categories in a data set
- **Scatter plots** - show relationships between different variables
    - Typically used for two variables for a set of data
- **Distribution graph** - displays the spread of various outcomes in a data set - x-axis don't have to be values

Design Principles:

| What to avoid | Why |
|---|---|
| Cutting off the y-axis | Changing the scale on the y-axis can make the differences between different groups in your data seem more dramatic, even if the difference is actually quite small. |
| Misleading use of a dual y-axis | Using a dual y-axis without clearly labeling it in your data visualization can create extremely misleading charts. |
| Artificially limiting the scope of the data | If you only consider the part of the data that confirms your analysis, your visualizations will be misleading because they don't take all of the data into account. |
| Problematic choices in how data is binned or grouped | It is important to make sure that the way you are grouping data isn't misleading or misrepresenting your data and disguising important trends and insights. |
| Using part-to-whole visuals when the totals do not sum up appropriately | If you are using a part-to-whole visual like a pie chart to explain your data, the individual parts should add up to equal 100%. If they don't, your data visualization will be misleading. |
| Hiding trends in cumulative charts | Creating a cumulative chart can disguise more insightful trends by making the scale of the visualization too large to track any changes over time. |
| Artificially smoothing trends | Adding smooth trend lines between points in a scatterplot can make it easier to read that plot, but replacing the points with just the line can actually make it appear that the point is more connected over time than it actually was. |

| Principle | Description |
|---|---|
| Choose the right visual | One of the first things you have to decide is which visual will be the most effective for your audience. Sometimes, a simple table is the best visualization. Other times, you need a more complex visualization to illustrate your point. |
| Optimize the data-ink ratio | The data-ink entails focusing on the part of the visual that is essential to understanding the point of the chart. Try to minimize non-data ink like boxes around legends or shadows to optimize the data-ink ratio. |
| Use orientation effectively | Make sure the written components of the visual, like the labels on a bar chart, are easy to read. You can change the orientation of your visual to make it easier to read and understand. |
| Color | There are a lot of important considerations when thinking about using color in your visuals. These include using color consciously and meaningfully, staying consistent throughout your visuals, being considerate of what colors mean to different people, and using inclusive color scales that make sense for everyone viewing them. |
| Numbers of things | Think about how many elements you include in any visual. If your visualization uses lines, try to plot five or fewer. If that isn't possible, use color or hue to emphasize important lines. Also, when using visuals like pie charts, try to keep the number of segments to less than seven since too many elements can be distracting. |

- **Correlation charts** - show relationship among data
    - **Correlation** - measure of the degree to which two variables move in relationship to each others; only indicating they have a pattern with each other (eg. higher temps is correlated with higher ice cream sales)
    - If one goes up and the other also goes up - positive correlation
    - Caution - audience might think it's **causation -** an action directly leading to an outcome (eg. lightning causes thunder)
- General rule - you should visually represent only the data that your audience needs in order to understand your findings; need to consider where you want them to focus
- **Change**: This is a trend or instance of observations that become different over time. A great way to measure change in data is through a line or column chart
- **Clustering**: A collection of data points with similar or different values. This is best represented through a distribution graph
- **Relativity**: These are observations considered in relation or in proportion to something else. You have probably seen examples of relativity data in a pie chart.
- **Ranking**: This is a position in a scale of achievement or status. Data that requires ranking is best represented by a column chart.
-




- Elements of art - **lines -** adds visuals and builds structure**, shapes** - create visual contrast**, color** (hue, intensity (bright or dark), **value** (how light or dark), **space**, **movement**
- **Nine principles of design:**
    - **Balance** - when key visual elements are distributed evenly
    - **Emphasis** - needs a focal point so audience knows where to concentrate - use color and value for contrast to make elements stand out
    - **Movement** - the path the viewer's eye travels as they look at data visualization or literal moment created by animations
    - **Pattern** - use similar shapes and colors to create patterns - highlight similarities between diff data sets or break up a pattern with a unique shape, color, line
    - **Repetition** - repeating chart types, shapes or colors adds to the effectiveness of visualization
    - **Proportion** - using various colors and sizes helps demonstrate that you are calling attention to specific visual over others
      (next 3 are used to useful checks after data visualization is finished)
    - **Rhythm** - having a sense of movement or flow
    - **Variety** - keeps the audience engaged - make it look interesting but not too much that it's confusing using diff elements of art
    - **Unity -** final data visualization should be cohesive

- **Data composition** - combining the individual parts in a visualization and displaying them together as a whole (stacked bar, donut, pie, treemap, stacked areas)
  Eg. what % of our traffic comes from each platform

- Relationships - scatterplot, buble, column/line, heatmap
  Eg. how has clicks increased with increases spend
- **Elements for effective visuals:**
  - Clear meaning - communicate their intended insight
  - Sophisticated use of contrast - separate the most important data from the rest
  - Refined execution - deep attention to detail using the elements of arts
- Always think of manager and stakeholders first when creating
- Consider the needs of your audience, type of data you are visualizing and the design thinking process

- **Design thinking** - process used to solve complex problems in a user-centric way
  - Identify alternative strategies for your visualizations that might not be clear right away
  - Look at the product through the eyes of the customer
  - **Empathize** - think about the emotions and needs of the target audience, avoid areas where people might face obstacles interacting with your visuals
  - **Define** - find your audiences needs, problems and your insights - think about which data to show
  - **Ideate** - generate your data viz ideas, creating drafts with color combos, experimenting with diff shapes
  - **Prototype** - put charts, dashboards, etc. together
  - **Test -** show to team members all options before presenting

- Need to have headlines, subtitles and labels
- Make it easy to understand what your chart's about
- Be onsite, avoid abbreviations, acronyms
- Bold, or few sizes larger, at the top aligned to the left
- Subtitle should match the rest of the charts, supports the headline by adding more context and description
- Labels are more effective than legends, give less work to the audience
- **Annotation** - briefly explains data or help focus the audience on a particular aspect of the data

| Visualization components | Guidelines | Style checks |
|---|---|---|
| Headlines | - **Content**: Briefly describe the data<br>- **Length**: Usually the width of the data frame<br>- **Position**: Above the data | - Use brief language<br>- Don't use all caps<br>- Don't use italic<br>- Don't use acronyms<br>- Don't use abbreviations<br>- Don't use humor or sarcasm |
| Subtitles | - **Content**: Clarify context for the data<br>- **Length**: Same as or shorter than headline<br>- **Position**: Directly below the headline | - Use smaller font size than headline<br>- Don't use undefined words<br>- Don't use all caps, bold, or italic<br>- Don't use acronyms<br>- Don't use abbreviations |
| Labels | - **Content**: Replace the need for legends<br>- **Length**: Usually fewer than 30 characters<br>- **Position**: Next to data or below or beside axes | - Use a few words only<br>- Use thoughtful color-coding<br>- Use callouts to point to the data<br>- Don't use all caps, bold, or italic |
| Annotations | - **Content**: Draw attention to certain data<br>- **Length**: Varies, limited by open space<br>- **Position**: Immediately next to data annotated | - Don't use all caps, bold, or italic<br>- Don't use rotated text<br>- Don't distract viewers from the data |

-
- Over 1B people in the world have a disability - keep this in mind when designing
- **Ways to make data more accessible:**
    - **Labels -** text explanations placed directly on visualizations
    - **Alternative text** - provides a textual alternative to non-text content
    - **Text-based format** - export data from charts and diagrams to sheets or excel
    - **Distinguishing** - separating foreground from background, use bright colors, contrasting textures and shapes
    - **Simplify** - don't include too much info, long chunks of text
- Red-green color blindness is the most common - avoid placing red on green vice versa
- Blue-yellow color blindness is less common - difficult to tell difference between blue and green or red and yellow - don't put on top of each other
- **Accessibility** - making sure that you are creating visuals that anyone can interact with, whether they have a long-term or temp impairment
    - The more inclusive it is, the more clear and impactful for everyone

- **Design chart in 60 mins.**
    - **Prep (5 min)** - brainstorm how you want your data to appear while considering the amount and type of data you have
    - **Talk and Listen (15)** - identify object of your work, establish expectations by asking questions and concentrating on the feedback from stakeholders
    - **Sketch and design (20)** - draft, define the timing and output of your work to get a clear and concise idea of what you're crafting
    - **Prototype and Improve(20)** - generate a visual solution and gauge its effectiveness at accurately communicating your data; repeat process until a final visual is produced

**WEEK 2**

- **Dynamic visualizations** - interactive or change over time
- **Tableau** - business intelligence and analytics platform that helps people see, understand and make decisions with data
- The more power you give the user, the less control you have over the story you want the data to tell - find balance between interactivity and controlled
- Comes down to which is the easier for the user to understand the point you're trying to make
- Other platforms for interactive data - **Looker, Google Data Studio -** both all browser based (Tableau is desktop and browser)
- **Dimensions** - the options on the left under tables
    - Below are different measure you can track for these dimensions
- **Diverging color palette** - displays two ranges of values using color intensity to show the magnitude of the number and the actual color to show which range the number is from
    - Green usually associated with positive, red - negative
- 4 rules recap:
    - **Five-second rule** - data should be clear, effective, convincing
    - **Color contrast** - use diverging color palette
    - **Conventions and expectations** - audience/cultural
    - **Minimal labels**
- Understand what data you are presenting (changes over time, frequency, categorical comparisons), apply the 4 rules, what does your audience need to see to understand the analysis, find a chart style that fits, make sure it's both accessible and aesthetically pleasing
- **Lasso tool** - used to select a data point in Tableau
- **Pan tool -** rotating perspective while keeping a certain object in view
- **Companion table** - displays the same data as the viz but in a table format

**Week 3**

- **Spotlighting** - scanning through data to quickly identify the most important insights
    - Eg. write each insight from analysis on piece of paper and spread them out on a whiteboard
    - Examine it and look for broad universal ideas sand messages
    - Look for things that keep popping up or numbers and words that are repeated often, patterns
    - Highlight them and group them together, find meaning behind the numbers
    - Find which insights are most likely to help solve your business problem or give you answers
- **Data storytelling** - communicating the meaning of a dataset with visuals and a narrative that are customized for each particular audience
    - Why something is happening with data, narrative, visuals
    - Know how to eliminate the less important details
    - Eg. year in review with spotify, doordash, etc.

    - Word clouds - unlock stories from big block of text where each word cold never be seen - shows which one shows up more often
    - 3 steps:
        - **1. Engage your audience** - capturing and holding their attention on an intellectual and emotional level
            - Know your audience (what is their role, what is their stake, what do they hope to get from the data insights)
        - **2. Create compelling visuals** - should be clear and able to stand on their own
            - Highlight the meaning behind the numbers at a glance
            - Incorporate some interactive layered data so the curious can explore
        - **3. Tell the story in an interesting narrative** - beg, mid, end, organized
            - Clearly articulate recurring themes and data points
            - Provide recommendations at the end
            - Choose a primary message/spotlight (clear and concise)

- **Data journalism** - journalists engage their audience by coming visuals, narrative and context into data-drive articles - have a lot in common with data analysts
    - Set context, analyze variable, draw conclusions
    - Best practice - include companion table, make sure percentages add up correctly, legend align with colors in chart, labeling effectively helps you achieve a cleaner chart and no legend required

- **Dashboard -** tool that organizes info from multiple datasets into one central location for tracking, analysis and simple visualization
    - Very important in step 2
    - Like a car's dashboard, data analytics dashboards take a tons of info and bring it to life
    - Monitors live, incoming data, keeps things neat and tidy
    - Start with most important data points, decide on placement or layout of graphs, charts, other visuals - need to be cohesive(balanced, make good use of space)
    - Resize, reorganize
    - Ensure that charts and graphs are most effective by placing them in a balanced layout and making good use of available space
    - Can choose either horizontal or vertical layout, tile or floating layouts
        - **Tiled** items - part of a single-layer grid that automatically resizes based on the overall dashboard size
        - **Floating** items can be layered over other objects
    - Sharing dashboards put storytelling power in viewers' hands - they'll craft their own narrative and draw their own conclusions
- **Static data** - providing screenshots or snapshots of data in presentation or building dashboards
    - Pros - can tightly control a point-in time narrative, allows for complex analysis to be explained in depth th to a larger audience

    - Cons - insight loses value and continues to the longer the data remain static, snapshots can't keep up with the pace of data change
- **Live data -** building dashboards, reports, and views connected to automatically updated data.
    - Pros - more dynamic and scalable, most up to date for people at the time when they need it
        - Allows for up to date curated views into data with the ability to build a scalable single source of truth for various uses
        - Immediate action can be taken on data that changes frequently
        - Alleviates time/resources septa on processes for every analysis
    - Cons - take up engineering resources to keep pipelines live and scalable (may be outside scope of some companies' data resource allocation)
        - Can lose control of narrative, which can cause data chaos(teams conflicting on conclusions based on the same data)
        - Potentially cause lack of trust if data isn't handled properly
- Filtering on tableau - Keep Only
    - Appropriate uses for filters in Tableau include highlighting individual data points, limiting the number of rows or columns in view, and providing data to different users based on their particular needs.
-
-

- The narrative you share with stakeholders needs:
    - **Characters** - people affected by your story - customer, stakeholders
        - When adding info about characters to your story, adds a personal account and bring more human context to the facts the data revealed - think about why they care
    - **Setting** - what's going on, how often it's happening, what tasks are involved
    - **Plot** - conflict, what creates tension in the current situation - challenge from competitor, inefficient process, new opportunity
    - **Big reveal** - resolution, how the data has shown that you can solve the problem the characters are facing by being more competitive, improving a process or inventing a new system
    - **Aha moment** - when you share your recommendations and explain why you think they'll help your company be successful

- Good visuals need a **theme** - controls the color, font types, sizes, formatting, positioning, make sure it matches tone and data
- Include title, subtitle, date (created, last updated)
- Keep texts to less than 5 lines and 25 words per slide - you want your audience focused on what you're saying and not busy reading slides
- Avoid slang, abbreviations
- Great visuals don't leave room for interpretation, the meaning is instantly understood
- Don't include too many details - what is the single most important thing I want my audience to learn from my analysis
- Create a new slide for every important point

- Big reveal and aha moment – your visuals must communicate these message with clarity and excitement
- You can either copy and paste, linked or embed a visual into a slideshow, main difference has to do with where yous tore them and how you update them after you place them in you slideshow
    - Copy and paste - you can edit it directly within a slideshow
        - if your visual or data points exist in other places (Tableau dashboard) any changes you make will not affect the original
        - Also means your visuals won't be updated if the original dataset changes
    - Linking - the visual lives within it's original file and the slideshow connects to it with the visual's URL
        - When you make changes to the original (spreadsheet for eg.) the changes will automatically appear in your presentation
        - Useful if data is likely to change over time
    - Embedded - also lives in original source file but it doesn't get automatically updated if the source file changes
        - You can make changes to it in your presentation without affecting the original

**Week 4**

- Strategic Framework - creates logical connections that tie back to the business tasks and metrics
    - Business task - the question or problem your analysis answers
    - Present your data in the context of your business task
- Establish the hypothesis early in the presentation to help your audience understand the data
- Mccandless method - move from general to specific
    a. Introduce the graphic by name
    b. Answer obvious questions before they're asked
    c. State the insight of your graphic
    d. Call out data to support insight
    e. Tell your audience why it matters - present the possible business impact of the solution and clear actions stakeholders can take
- Explain solution to your business tasks by using examples and visualizations
- Ask yourself if this data point or chart support the point I want people to walk away with

- Presentation tips - keep it kindergarten simple, have a story, make it fun, have allies in your audience, don't create eyesore charts
- Bad data presentation - no story, logical flow, no titles, too much text, inconsistent format(no theme), no conclusion at the end
    - People don't know where to focus their attention

- Good presentation - people are logically guided through the data
- they can understand what the data means and has takeaways about how they can use their understanding to make a change
- Clear objective at the beginning and conclusion at the end, common themes
- First slide - simple title, presenter name, date
- Second - table of contents - purpose statement (what are we talking about), tell your story with data, conclusion, appendix(additional info that may not work with overall flow of presentation)
- Have transition slides in big letters- “What are we talking about”, “Present Data”,”Conclusion” say the next step, what we did next
- Have an objective slide
- Populate text or labels as you begin to discuss it, audience won’t be overwhelmed
- Explain the visual first before moving on (why you used it, x-axis, etc)

**Proven Presentation tips**

1. Channel your excitement
2. Start with broader ideas
3. Use the five second rule - wait 5 seconds after showing a data visualization, ask if they understand, give your audience another 5 seconds, tell them the conclusion
4. Preparation is key - writing a script and repeating in your head

- Keep in mind:
    - Your audience will not always see the steps you took to reach a conclusion
    - The curse of knowledge - because you know something, its hard to imagine someone not knowing it
    - Your audience already has a lot on their mind, they can be easily distracted
    - Share the right amount to keep your audience focused
- How you speak: how you present is just as important as what you present
    - Keep your sentences short
    - Build in intentional pauses
    - Keep the pitch of your sentence level
    - Be mindful of nervous habits - stay still, move with purpose, practice good posture, make positive eye contact
- Anticipating questions:
    - Understand stakeholder’s expectations
    - Make sure you have a clear understanding of the objective and what the stakeholders wanted
    - Do a test-run of presentation to colleague - helps assess potential questions, assumptions, areas that might be unclear
    - Start with zero assumptions - don’t assume your audience if familiar with jargon, past events, or other background info
- Be prepared to consider any limitations of your data by:
    - Critically analyzing the correlations
    - Looking at the context
    - Understanding the strengths and weakness of the tools

# Before the presentation

1. Assemble and prepare your questions.
2. Discuss your presentation with your manager, other analysts, or other friendly contacts in your organization.
3. Ask a manager or other analysts what sort of questions were normally asked by your specific audience in the past.
4. Seek comments, feedback, and questions on the deck or the document of your analysis.
5. At least 24 hours ahead of the presentation, try and brainstorm tricky questions or unclear parts you may come across- this helps avoid surprises.
6. It never hurts to practice what you will be presenting, to account for any missing information or simply to calm your nerves.

# During the presentation

1. Be prepared to respond to the things that you find and effectively and accurately explain your findings.
2. Address potential questions that may come up.
3. Avoid having a single question derail a presentation and propose following-up offline.
4. Put supplementary visualizations and content in the appendix to help answer questions.

**Handling Objections:**

- Usually About data, analysis, findings
    - Where you got the data, what systems it came from, what transformations happened to it, how fresh and accurate it is (have this info in the beginning to set up data context, keep log of transformations in appendix in case)
    - Is your analysis reproducible (keep change log), who did you get feedback from
    - Do these findings exist in previous time periods, did you control for the differences in your data
- Responding to possible objections
    - Communicate any assumptions (tell your audience that your team cleaned and formatted the data before analysis)
    - Explain why your analysis might be diff than expected
    - Acknowledge objections that have merit and take steps to investigate further

**Q&A best practices**

- Listen to the whole question
- Repeat the question (makes sure you're understanding the question, gives the person a chance to correct you if not, anyone who didn't hear will now know, gives a moment to get your thoughts together)
    - Appendix - useful for answering questions - have more detail info about data

- Understand the context questions are benign asked in - who is your audience, what concerns they have, project goals, stakeholders interests
- Involve the whole audience - not just one on one convo
- Keep your responses short and to the point - if they have more questions, then go into more detail
- If audience is losing interest - redirect to anew question or ask a question to audience to re-engage them

- Become an expert data translator
- It’s not the data itself that is important to the company, it’s the understanding of it and the impact that it can take
- Presenting the data is the most important aspect as a data analyst, doesn’t matter how compelling the analysis is or how accurate if people don’t understand it, ti does not’ offer value to the business
- Taking something technical and simplifying it down so everyone in the room understands
- Important aspects - define your purpose, keep it concise, have logical flow, make it visually compelling, how easy is it to understand

- If your data has limitations you can prepare to explain them by understanding the strength and weaknesses of your tools, consider the context, critically analyze any correlations