**How I built this dashboard**

In this project I am going to learn how to import data from web application where we can modify data dynamiclally and get results in PowerBI directly. I’m going to learn how to convert a google sheet into web application and I’m going to modify the data directly on Google sheet and get results in Power BI directly.

The first thing is to download the dataset. The dataset is sourced from the following link on GitHub: [Tableau-Dashboards-info/Friends dataset.csv at main · DataScienceRoadMapDSRM/Tableau-Dashboards-info (github.com)](https://github.com/DataScienceRoadMapDSRM/Tableau-Dashboards-info/blob/main/Friends%20dataset.csv). Once we have the dataset open it in excel, copy everything and create a new Google sheet, paste everything, rename the file as Friends, rename the sheet as Friends. To convert this into a web API, we are going to use a code which is sourced from [Tableau-Dashboards-info/Google sheet to web API script.txt at main · DataScienceRoadMapDSRM/Tableau-Dashboards-info (github.com)](https://github.com/DataScienceRoadMapDSRM/Tableau-Dashboards-info/blob/main/Google%20sheet%20to%20web%20API%20script.txt). In Google Sheets, go to Extensions > App Script, rename the project Friends. Paste the code, save > Run the code. Give all necessary permissions and see that the execution Log in App script has 2 thing correct, i.e, Execution started and Execution completed. Noe click on deploy > New deployment. Select type > Web App. In web application do 2 things : In description put the name as Friends, for authorization we have to put as anyone with the link in order to use the link in PowerBI.

Copy the web app link and open new powerbi project. Select get data/web / copy the url link in the tab > OK > Connect. This way we will get our dataset in PowerBI.

1. The first thing we are going to do is change the background. Visualization > format page > Canvas background > Colour > #743A37 > transparency 0%.
2. Visualizations > cards > Fields > season as count distinct. Rename it as total seasons. Format > General > Effects > Background > Off > Visual Border > colour > #F8E7D8 > Rounded corners > 20 points. Visual > Callout value >Colour > #F8E7D8 > FONT 25, Category label colour > our colour.
3. Copy paste this card 5 times to get a total of 6 cards. The field for 2nd card will be total episodes as count, rename it to total episode. 3RD card will be average star rating, rename it as average star rating, next we have duration in minutes as the average, rename it as average duration (min) as the average. Next we have the total votes rename as total votes. Last card is the last season year as the maximum.
4. Copy paste the card total seasons. Convert it into clustered column chart. X-axis season, y-axis count of episodes. Format visual > columns > our colour, grid lines > off, y-axis > title off , values >our colour, same for x-axis. General > title > Episodes by season. add data labels > values same colour.
5. Create a ribbon chart for total votes by episodes and season. X-axis > seasons, y-axis > total votes. For the legend we are going to have episodes. Format the visual. General > background > off. Visual border > same colour > 20 points. Visual > gridlines > off for both vertical and horizontal. Legend > text > our colour, legend position > top centre. Title for legend off. X-axis > values > our colour, bold, title > season. Similar for y-axis > title / total votes. General > title > total votes by episodes and season, centre, bold, our colour.
6. Import the image > insert > image.
7. To create a boxplot, in visualization, click on More visuals, select box and whiskers chart > add. Axis > total stars, category 1 > season. Format your visual > Visual > Gridlines > axis grid lines off, tick marks > axis ticks off. Foe dots transparency 100%.