

Name: Alanoud AlHwairy

GitHub: AlanoudAlHw@gmail.com

Riyadh metro report

Abstract

The goal of this project was to use EDA to predict the new schedule in Riyadh metro in order to help improve operations, quality and gain customer satisfaction. I worked with data provided by MTA. After refining a dataset, I built 3 graphs to visualize and communicate my results using Python and matplotlib library.

Design

The objective of this study is to use past application data to predict and optimize the new schedule for Riyadh metro and avoid the crowd by estimating the rush hours in the past 3 months in addition know which the most demand station by the passengers, the question has to answer what is the busiest top 5 stations, what is metro's rush hours, what is the most crowded day?

Data

In view of solving the problem and obtaining a realistic result, we decided to use the MTA Turnstile dataset for the month of June, July and August for the year 2021. The data set has 2722610 rows and 11 columns.

Algorithm

In data cleaning stage I observe that the dataset does not contain NULL values but, there is some duplications were found in the dataset and there are some negative values found after calculating the difference between the ENTRIES column. There are a few outliers in the DAILY ENTRIES Column and all I removed it before working on it.

Tools

- Python programming language
- Pandas, Numpy, Matplotlib, sqlalchemy.
- Jupyter notebook.
- SQLite.

Communication

Presentation slides:

Problem understanding قطار الرياض riyadh metro



Overview:

The Riyadh Metro has a capacity of 1.16 million passengers per day, and includes 85 stations that transport many passengers at different times.

Problem statement :

The objective of this study is to use past application data to predict and optimize new schedule for Riyadh metro and avoid the crowd by estimating the rush hours in past 3 months in addition know which the most demand station by the passengers

Scope:

In view of solving the problem and obtaining a realistic result, we decided to use the MTA Turnstile dataset for the month of June, July and August for the year 2021

Data Validation



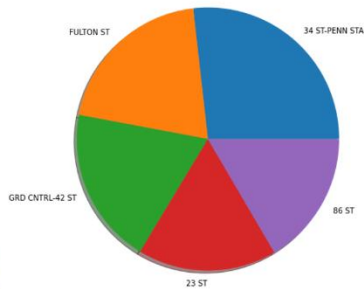
- I observe that the dataset does not contain NULL values.
- There some Duplications were found in the dataset
- There are some Negative values found after calculating the difference between the ENTRIES column.
- There are a few OUTLIERS in the DAILY ENTRIES Column.

The Graphs



This pie chart studies the relationship between the number of passengers in the three months and the most frequented stations

The Most Popular Stations

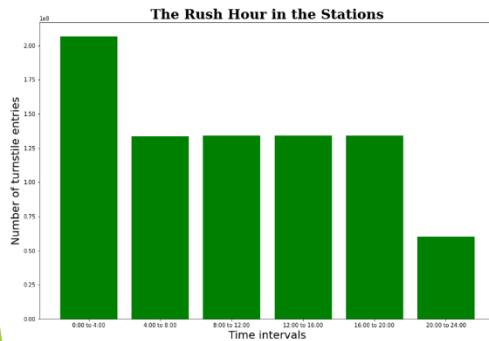


Observation:

We note here the five most requested stations by passengers, which are 34 ST-PENN STATION, FULTON ST, GRD CTRL-42 ST, 23 ST, 86 ST which means that the most passengers frequent these stations.

The Graphs

This bar chart represents the relationship between the number of passengers per time.

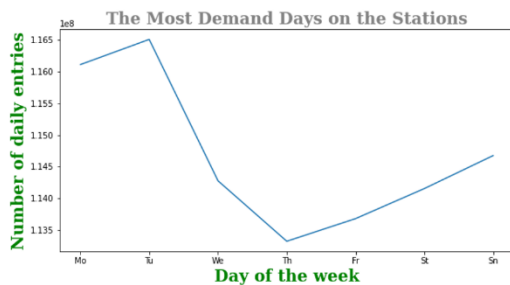


Observation:

We note here the most time interval requested by passengers, which are from 0:00 to 4:00 then from 4:00 to 8:00. which represent the metro's rush hours.

The Graphs

This chart represents the relationship between the number of passengers per Day of the week.



Observation:

We note here the 3 most frequent days by passengers, which are in order Tuesday, Monday, then Sunday

Challenges



قطار الرياض
riyadh metro



- Unclear columns.
- Large number of rows.
- Limited time.

Conclusion



قطار الرياض
riyadh metro

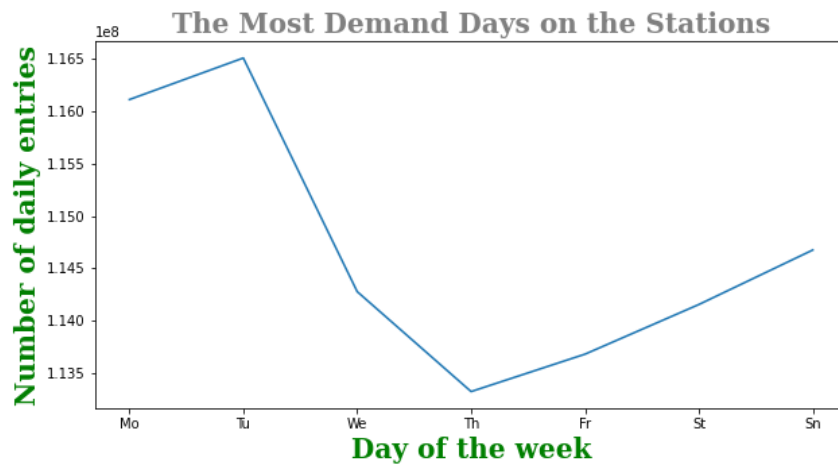
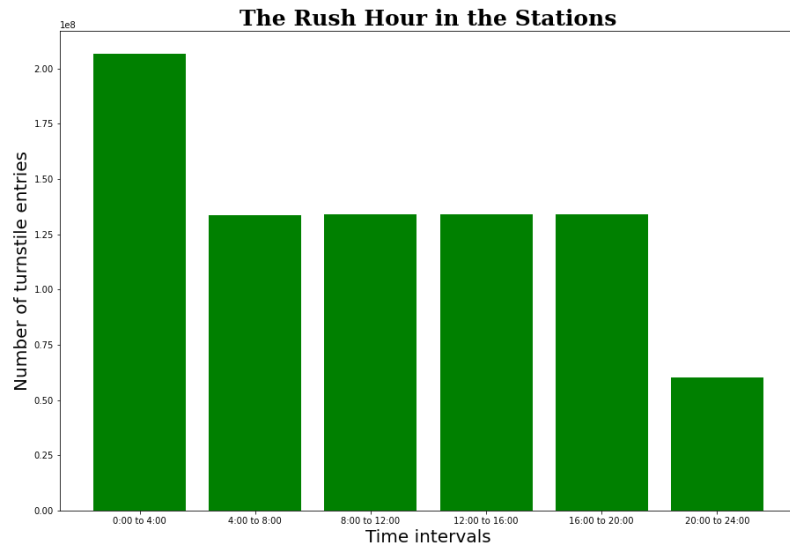


To avoid the crowd and increase the productivity of Riyadh metro, in addition, improve the quality to gain customer satisfaction.

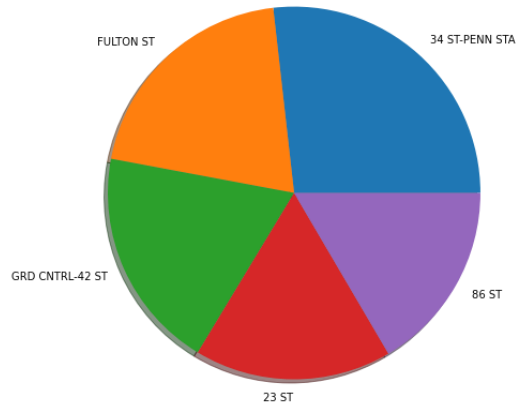
We have to take into account the following criteria when creating the new schedule for Riyadh Metro:

- Increase the number of lines to the following stations (34 ST-PENN STATION, FULTON ST, GRD CTRL-42 ST, 23 ST, 86 ST).
- Increasing trips at the following times from 0:00 to 4:00 and from 4:00 to 8:00. especially on Tuesday, Monday, then Sunday.

The Graphs:



The Most Popular Stations



The SQL Database commands:

DB Browser for SQLite - C:\Users\noodly\Downloads\EDA_project\MTA_NEW.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragma Execute SQL

SQL 1

```
1 SELECT * from MTA_Table;
```

	C/A	UNIT	SCP	STATION	LINENAME	DIVISION	DATE	TIME	DESC	ENTRIES	
1	A002	R051	02-00-00	59 ST	NQR456W	BMT	08/21/2021	00:00:00	REGULAR	7622548	2607
2	A002	R051	02-00-00	59 ST	NQR456W	BMT	08/21/2021	04:00:00	REGULAR	7622561	2607
3	A002	R051	02-00-00	59 ST	NQR456W	BMT	08/21/2021	08:00:00	REGULAR	7622573	2607
4	A002	R051	02-00-00	59 ST	NQR456W	BMT	08/21/2021	12:00:00	REGULAR	7622604	2607
5	A002	R051	02-00-00	59 ST	NQR456W	BMT	08/21/2021	16:00:00	REGULAR	7622715	2607
6	A002	R051	02-00-00	59 ST	NQR456W	BMT	08/21/2021	20:00:00	REGULAR	7622861	2607

Execution finished without errors.
Result: 2722610 rows returned in 402ms
At line 1:
SELECT * from MTA_Table;

DB Browser for SQLite - C:\Users\noodly\Downloads\EDA_project\MTA_NEW.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragma Execute SQL

SQL 1

```
1 SELECT DATE,time, count(TIME) AS E FROM MTA_Table
2 GROUP BY DATE, Time
3 ORDER BY E DESC;
```

	DATE	TIME	E
1	07/12/2021	16:00:00	2574
2	07/13/2021	00:00:00	2574
3	07/13/2021	12:00:00	2574
4	07/11/2021	16:00:00	2573
5	07/13/2021	04:00:00	2573
6	07/13/2021	08:00:00	2573
7	07/13/2021	16:00:00	2573

Execution finished without errors.
Result: 171618 rows returned in 17895ms
At line 1:
SELECT DATE,time, count(TIME) AS E FROM MTA_Table
GROUP BY DATE, Time
ORDER BY E DESC;

DB Browser for SQLite - C:\Users\nood\Downloads\EDA_project\MTA_NEW.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragma Execute SQL

SQL 1

```
1 SELECT STATION ,count(ENTRIES) AS E FROM MTA_Table
2 GROUP BY STATION
3 ORDER BY E DESC;
```

	STATION	E
1	34 ST-PENN STA	55373
2	FULTON ST	52348
3	GRD CNTRL-42 ST	48523
4	23 ST	39840
5	86 ST	35391
6	CANAL ST	31650
7	59 ST	30677

Execution finished without errors.
Result: 379 rows returned in 11349ms
At line 1:
SELECT STATION ,count(ENTRIES) AS E FROM MTA_Table
GROUP BY STATION
ORDER BY E DESC;