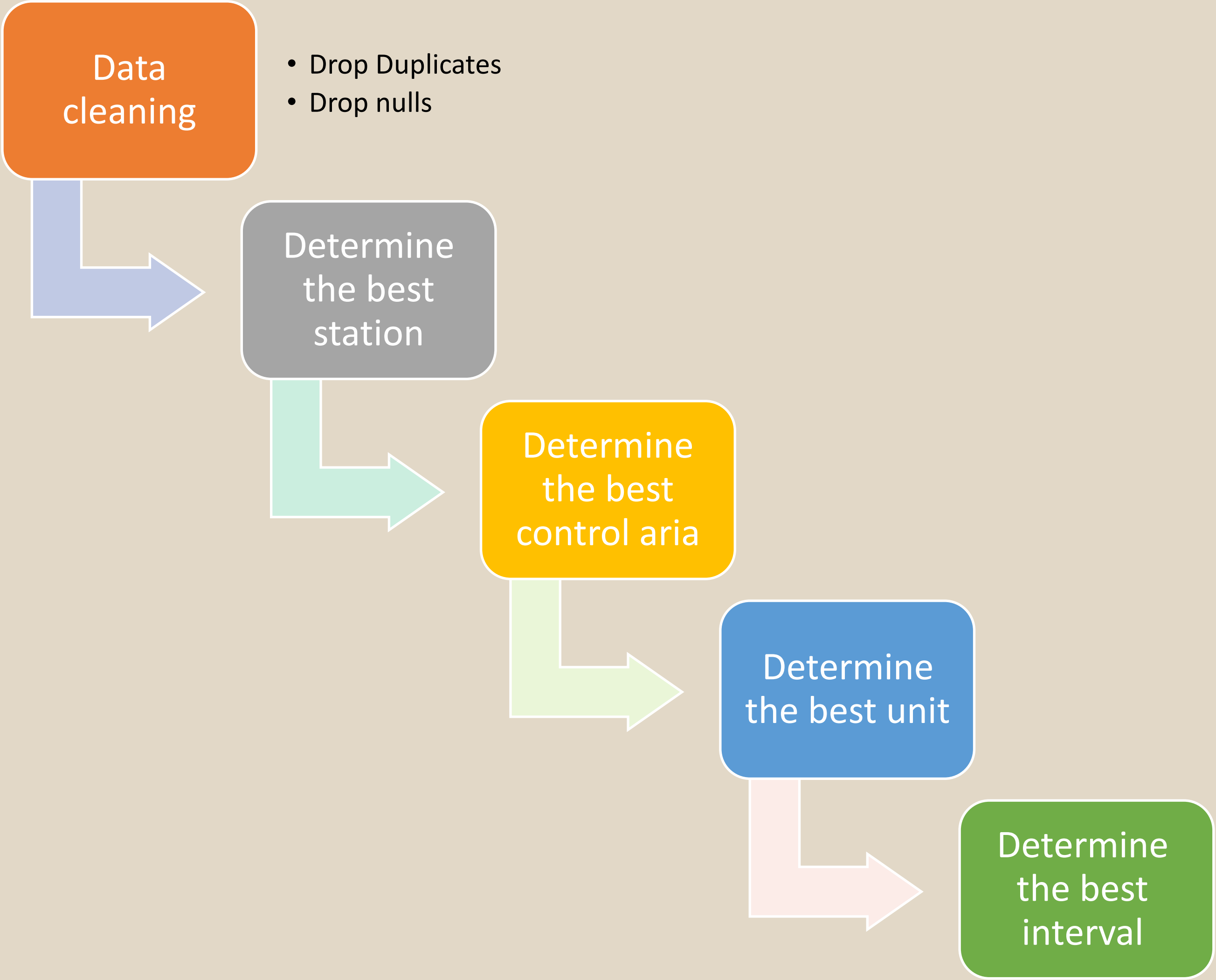
MTA Dataset

Dimah Abdulrahman Albunayyih

Backstory :

Adel is my client. He has a Food Truck, He wants to place it beside the best station and control aria. Also, he wants to put an announcement beside the best unit in that control Aria. He hired me to so.

My planning



Data cleaning

- Drop Duplicates
- Drop nulls

Determine the best station

Determine the best control aria

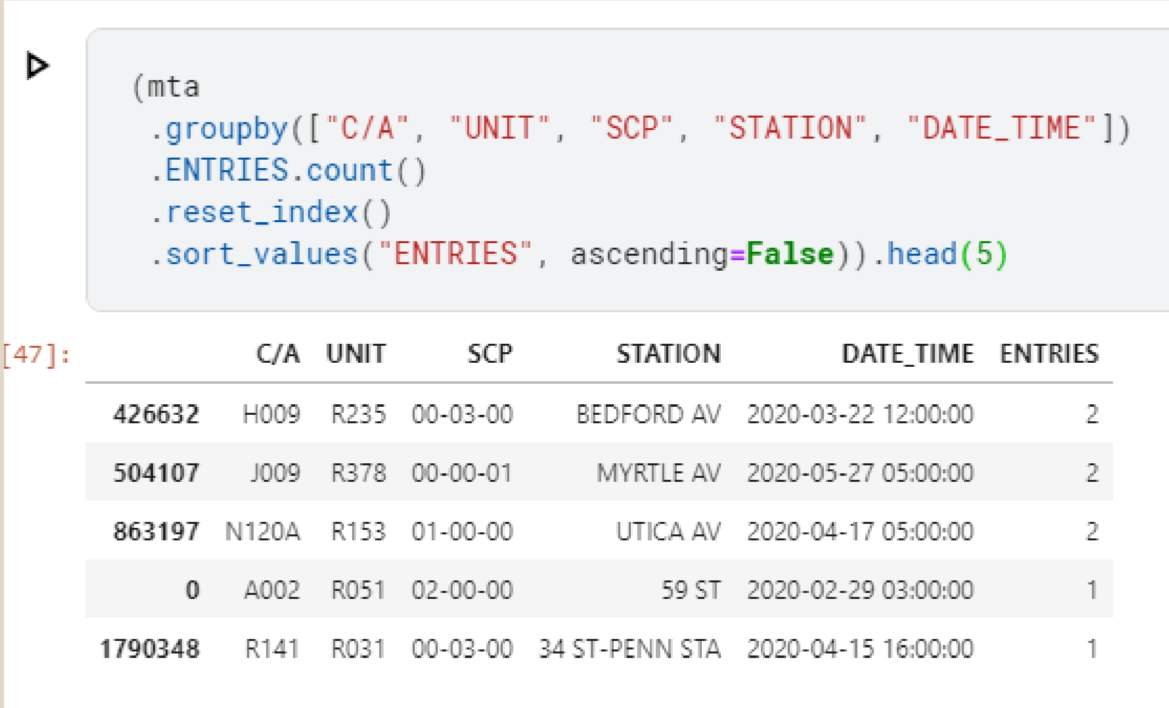
Determine the best unit

Determine the best interval

- The Morning or the evening

Data Cleaning

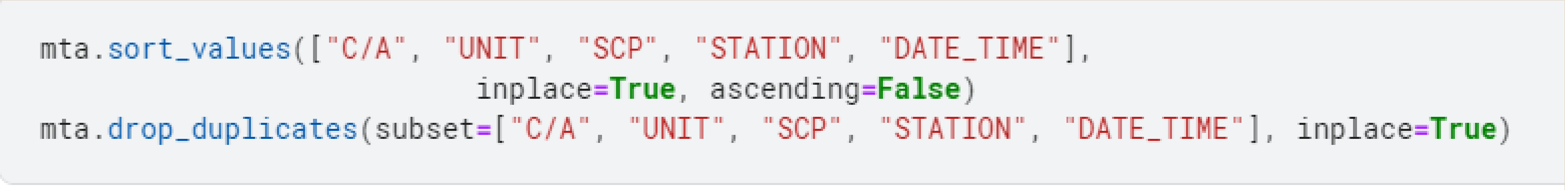
- Drop Duplicates



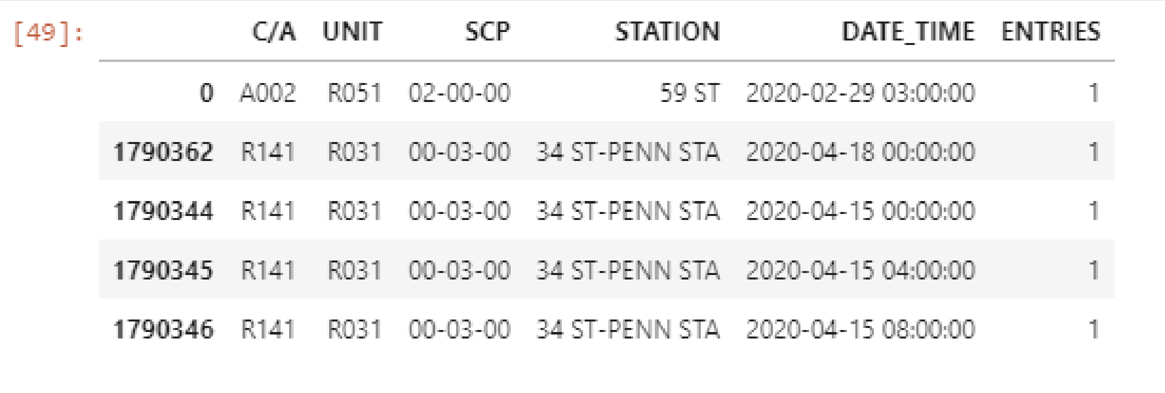
```
(mta
 .groupby(["C/A", "UNIT", "SCP", "STATION", "DATE_TIME"])
 .ENTRIES.count()
 .reset_index()
 .sort_values("ENTRIES", ascending=False)).head(5)
```

[47]:

	C/A	UNIT	SCP	STATION	DATE_TIME	ENTRIES
426632	H009	R235	00-03-00	BEDFORD AV	2020-03-22 12:00:00	2
504107	J009	R378	00-00-01	MYRTLE AV	2020-05-27 05:00:00	2
863197	N120A	R153	01-00-00	UTICA AV	2020-04-17 05:00:00	2
0	A002	R051	02-00-00	59 ST	2020-02-29 03:00:00	1
1790348	R141	R031	00-03-00	34 ST-PENN STA	2020-04-15 16:00:00	1



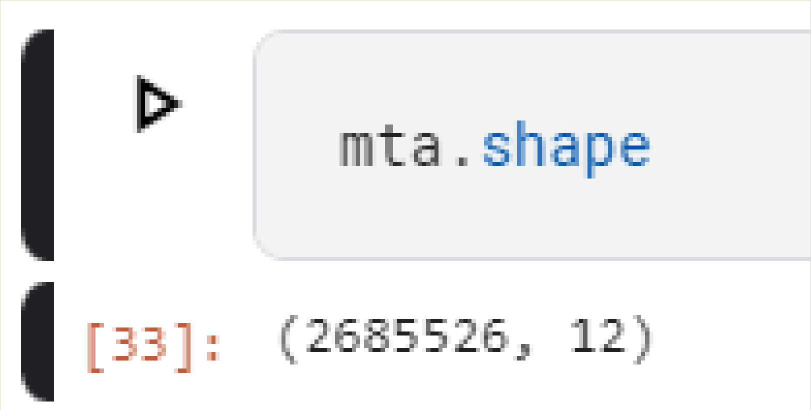
```
mta.sort_values(["C/A", "UNIT", "SCP", "STATION", "DATE_TIME"],
                inplace=True, ascending=False)
mta.drop_duplicates(subset=["C/A", "UNIT", "SCP", "STATION", "DATE_TIME"], inplace=True)
```



[49]:

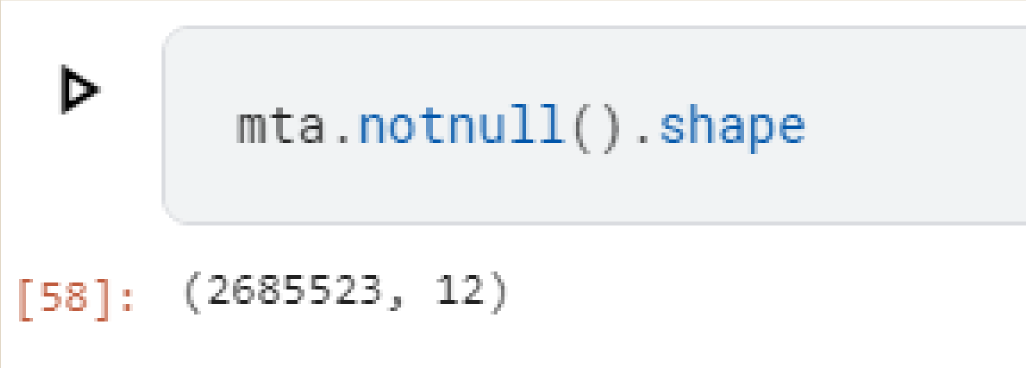
	C/A	UNIT	SCP	STATION	DATE_TIME	ENTRIES
0	A002	R051	02-00-00	59 ST	2020-02-29 03:00:00	1
1790362	R141	R031	00-03-00	34 ST-PENN STA	2020-04-18 00:00:00	1
1790344	R141	R031	00-03-00	34 ST-PENN STA	2020-04-15 00:00:00	1
1790345	R141	R031	00-03-00	34 ST-PENN STA	2020-04-15 04:00:00	1
1790346	R141	R031	00-03-00	34 ST-PENN STA	2020-04-15 08:00:00	1

- Drop Nulls



```
mta.shape
```

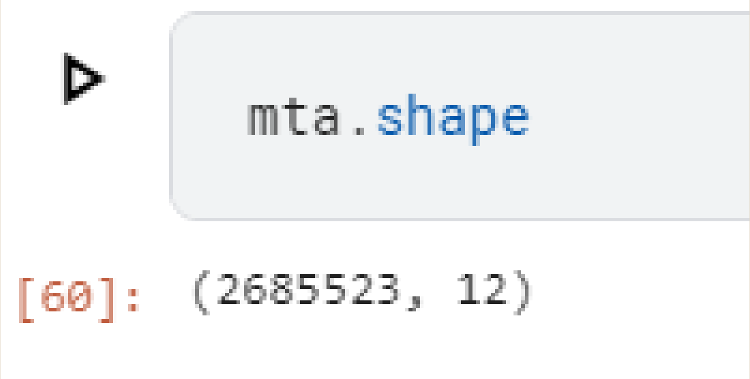
```
[33]: (2685526, 12)
```



```
mta.notnull().shape
```

```
[58]: (2685523, 12)
```

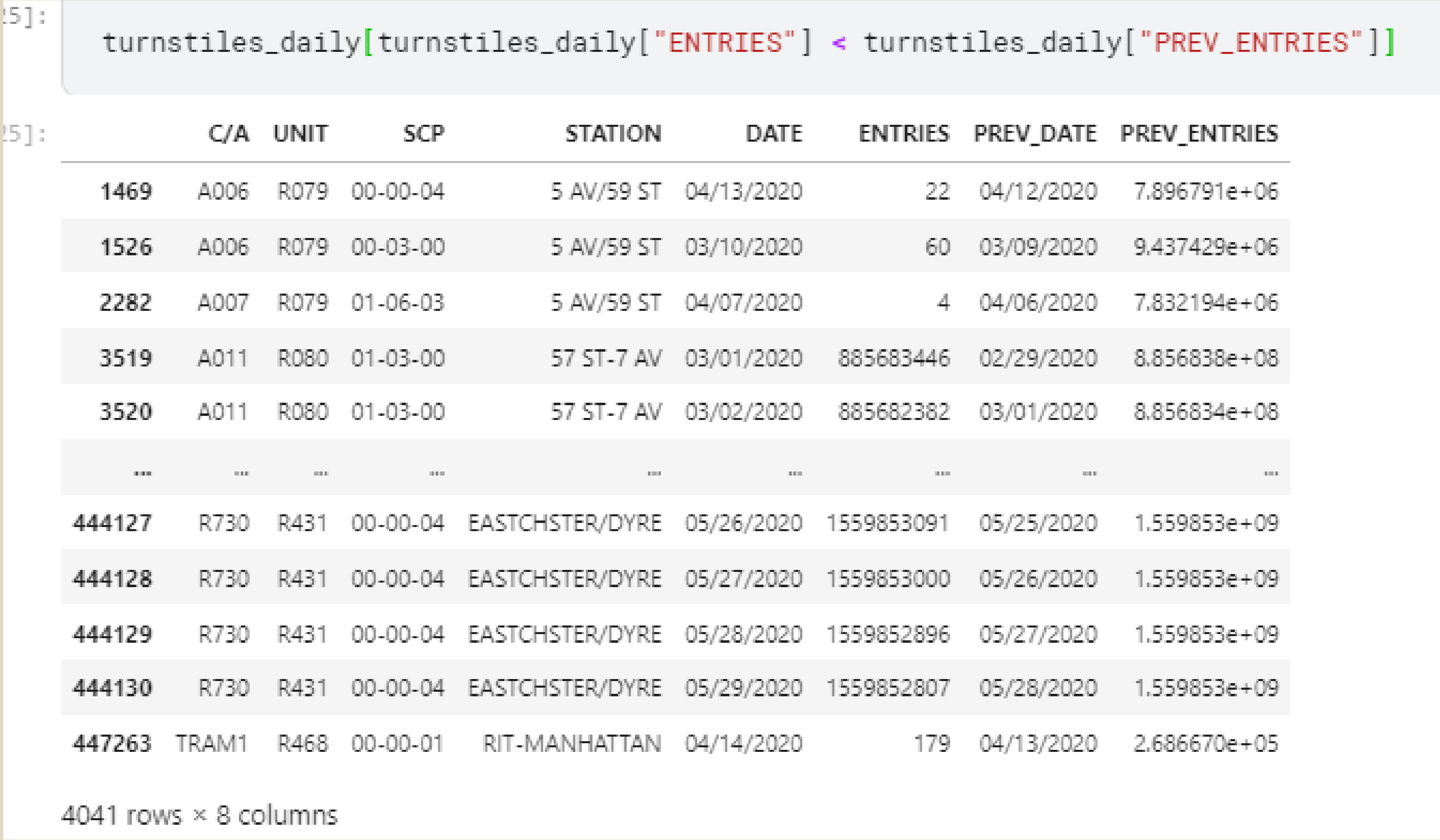
Use .dropna() Function



```
mta.shape
```

```
[60]: (2685523, 12)
```

- Check if Entries < Previous Entries

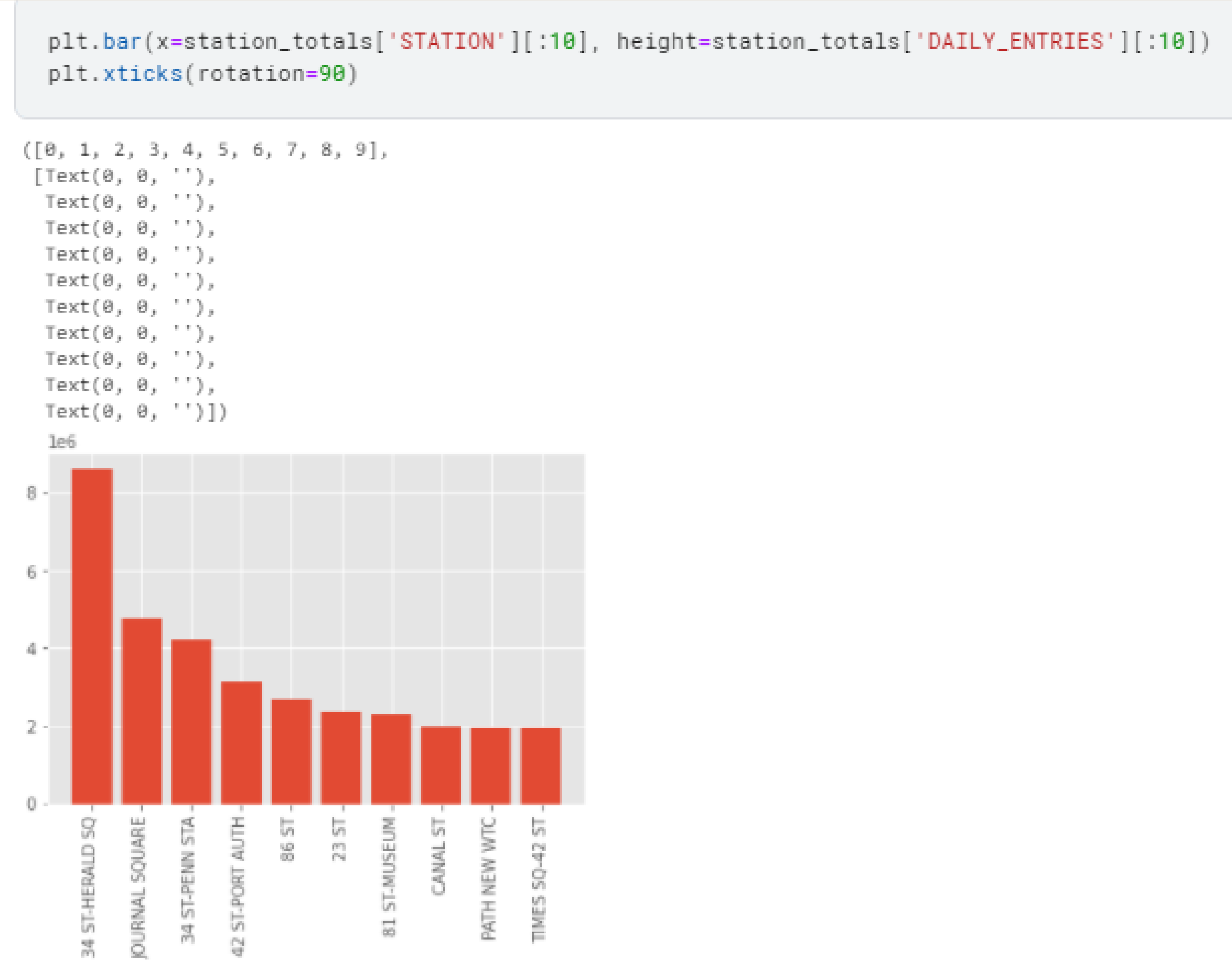


```
turnstiles_daily[turnstiles_daily["ENTRIES"] < turnstiles_daily["PREV_ENTRIES"]]
```

	C/A	UNIT	SCP	STATION	DATE	ENTRIES	PREV_DATE	PREV_ENTRIES
1469	A006	R079	00-00-04	5 AV/59 ST	04/13/2020	22	04/12/2020	7.896791e+06
1526	A006	R079	00-03-00	5 AV/59 ST	03/10/2020	60	03/09/2020	9.437429e+06
2282	A007	R079	01-06-03	5 AV/59 ST	04/07/2020	4	04/06/2020	7.832194e+06
3519	A011	R080	01-03-00	57 ST-7 AV	03/01/2020	885683446	02/29/2020	8.856838e+08
3520	A011	R080	01-03-00	57 ST-7 AV	03/02/2020	885682382	03/01/2020	8.856834e+08
...	...	...	...	...	...	...	...	...
444127	R730	R431	00-00-04	EASTCHSTER/DYRE	05/26/2020	1559853091	05/25/2020	1.559853e+09
444128	R730	R431	00-00-04	EASTCHSTER/DYRE	05/27/2020	1559853000	05/26/2020	1.559853e+09
444129	R730	R431	00-00-04	EASTCHSTER/DYRE	05/28/2020	1559852896	05/27/2020	1.559853e+09
444130	R730	R431	00-00-04	EASTCHSTER/DYRE	05/29/2020	1559852807	05/28/2020	1.559853e+09
447263	TRAM1	R468	00-00-01	RIT-MANHATTAN	04/14/2020	179	04/13/2020	2.686670e+05

4041 rows × 8 columns

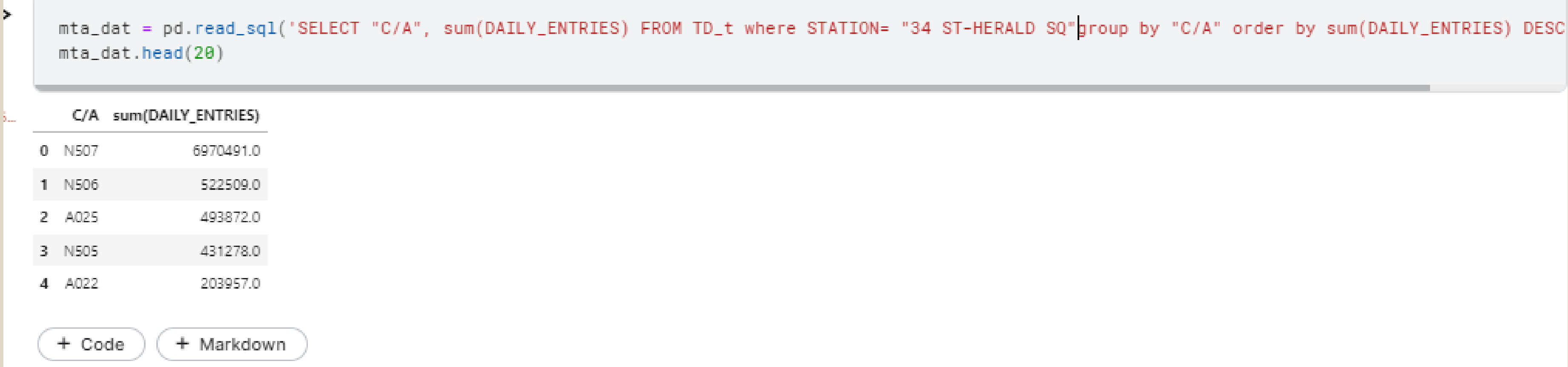
The busiest station in MTA:



```
plt.bar(x=station_totals['STATION'][:10], height=station_totals['DAILY_ENTRIES'][:10])
plt.xticks(rotation=90)
```

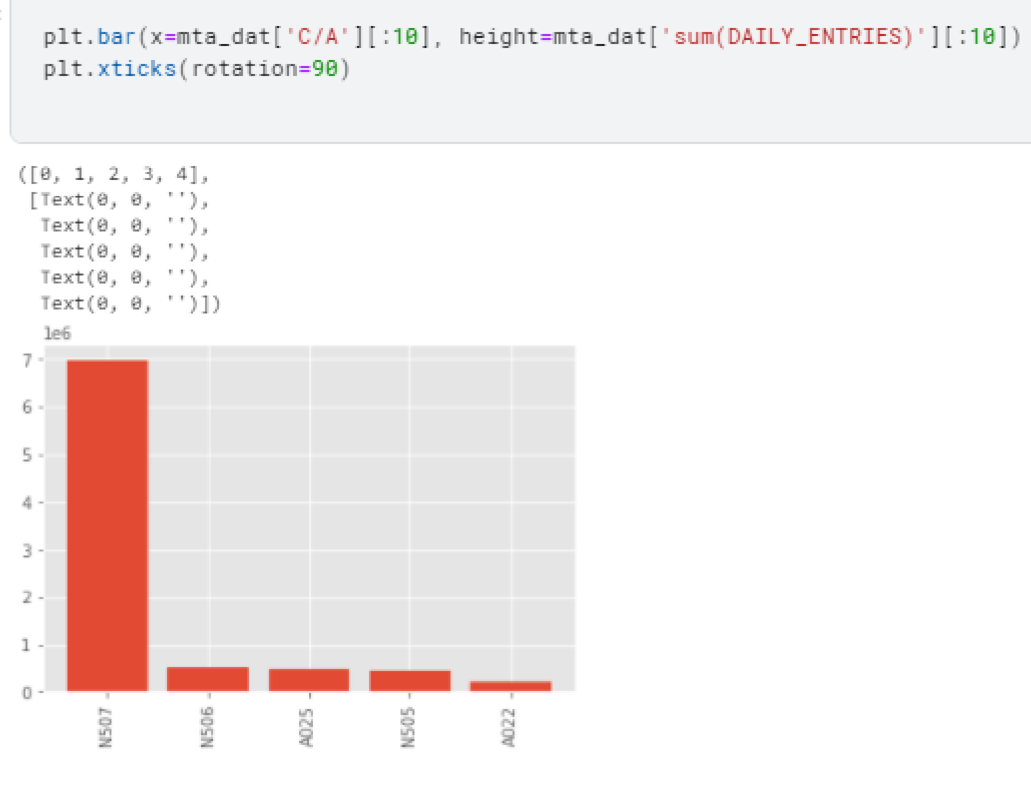
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9],
 [Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, '')])
```

The busiest C\A in MTA where STATION= 34 ST-HERALD SQ:



```
mta_dat = pd.read_sql('SELECT "C/A", sum(DAILY_ENTRIES) FROM TD_t where STATION= "34 ST-HERALD SQ"group by "C/A" order by sum(DAILY_ENTRIES) DESC
mta_dat.head(20)
```

	C/A	sum(DAILY_ENTRIES)
0	N507	6970491.0
1	N506	522509.0
2	A025	493872.0
3	N505	431278.0
4	A022	203957.0

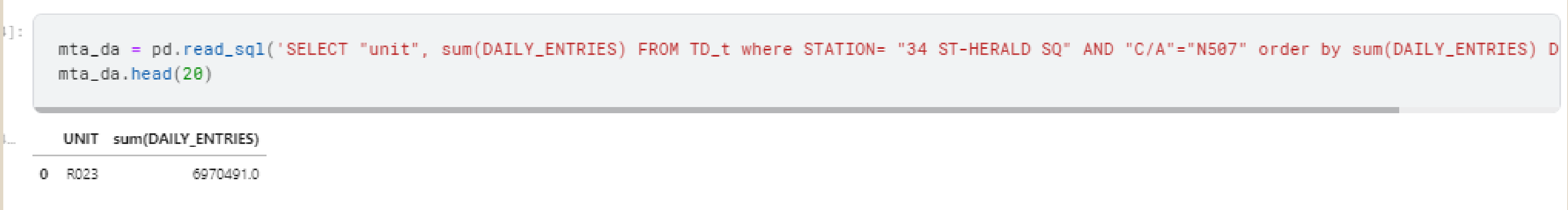


```
plt.bar(x=mta_dat['C/A'][:10], height=mta_dat['sum(DAILY_ENTRIES)'][:10])
plt.xticks(rotation=90)
```

```
([0, 1, 2, 3, 4],
 [Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, '')])
```

The busiest UNIT Where C\A = N507:

R023

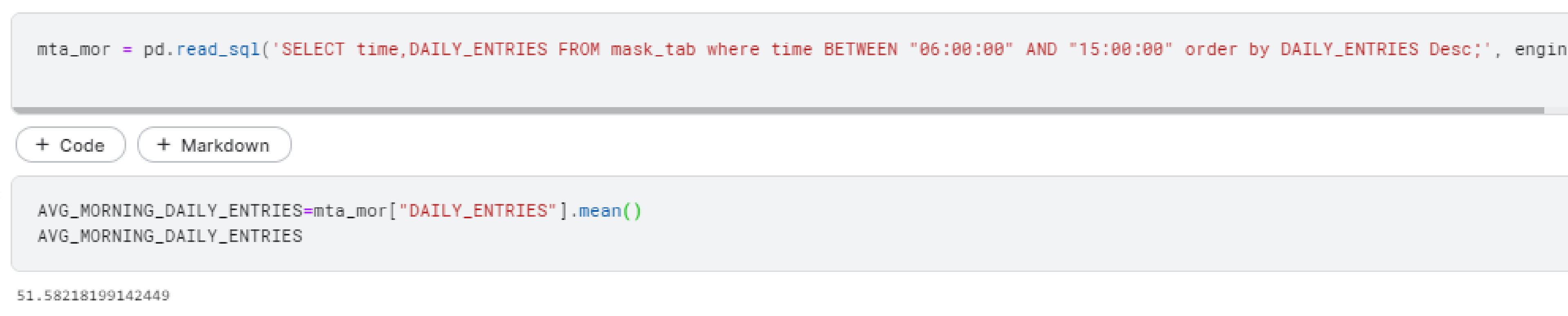


```
mta_da = pd.read_sql('SELECT "unit", sum(DAILY_ENTRIES) FROM TD_t where STATION= "34 ST-HERALD SQ" AND "C/A"="N507" order by sum(DAILY_ENTRIES) D
mta_da.head(20)
```

```
   UNIT  sum(DAILY_ENTRIES)
0  R023           6970491.0
```

Exit
Hope St &
Powers St
Food Truck
SR 17
SR 32
45
شاهي
جمر
2
SR

The Average of Daily Entries in morning interval:

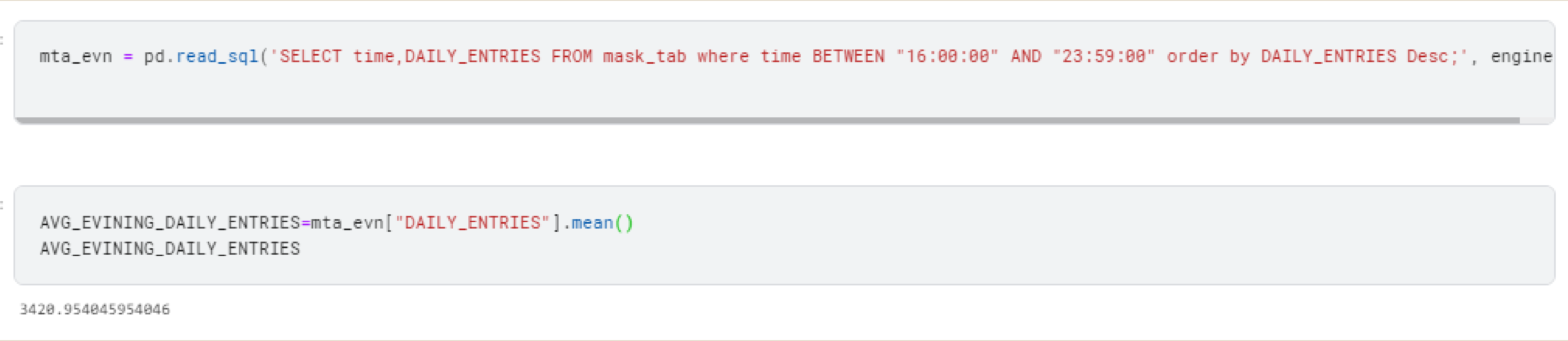


```
mta_mor = pd.read_sql('SELECT time,DAILY_ENTRIES FROM mask_tab where time BETWEEN "06:00:00" AND "15:00:00" order by DAILY_ENTRIES Desc;', engin
```

```
AVG_MORNING_DAILY_ENTRIES=mta_mor["DAILY_ENTRIES"].mean()
AVG_MORNING_DAILY_ENTRIES
```

```
51.58218199142449
```

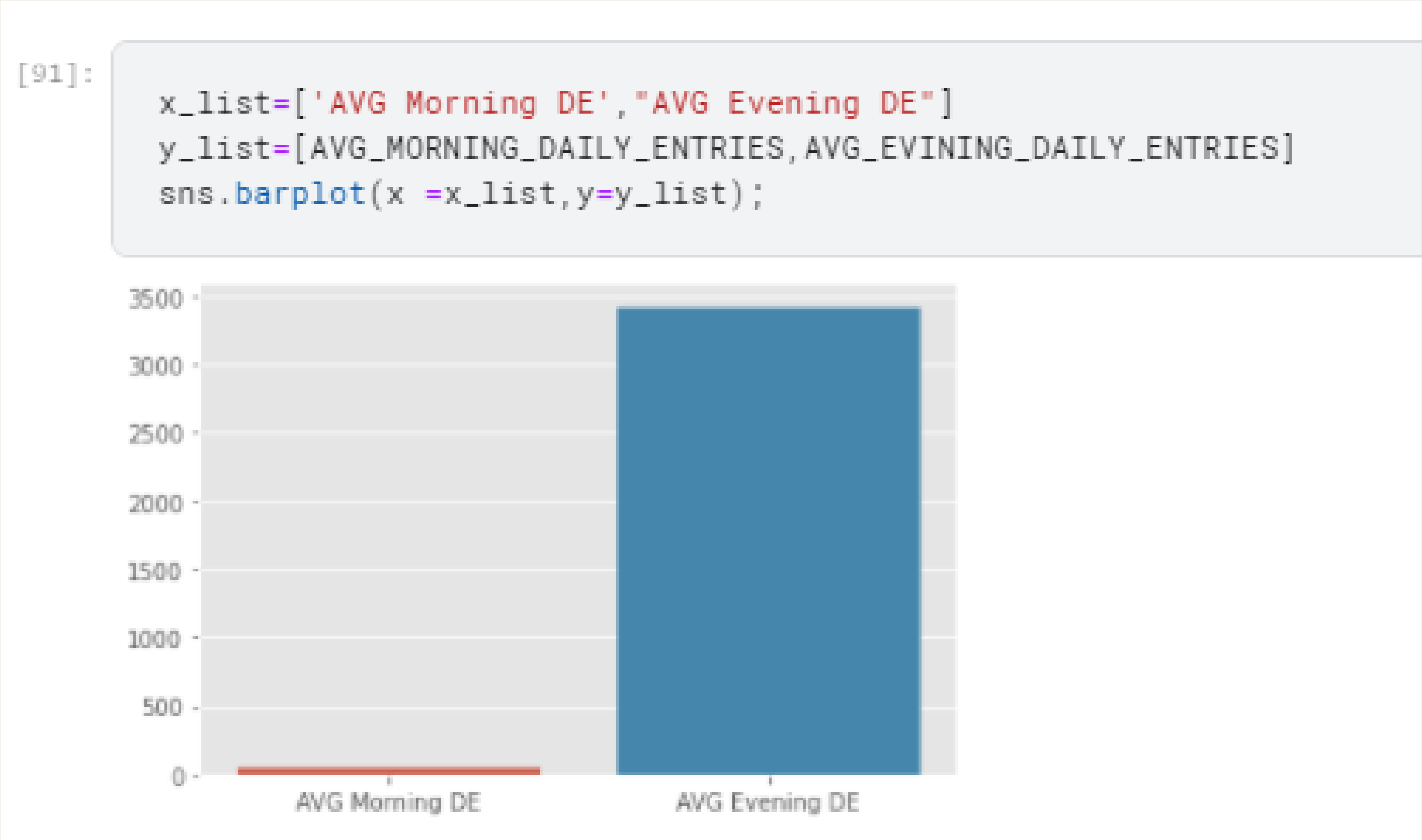
The Average of Daily Entries in Evening interval:



```
mta_evn = pd.read_sql('SELECT time,DAILY_ENTRIES FROM mask_tab where time BETWEEN "16:00:00" AND "23:59:00" order by DAILY_ENTRIES Desc;', engine
```

```
AVG_EVINING_DAILY_ENTRIES=mta_evn["DAILY_ENTRIES"].mean()
AVG_EVINING_DAILY_ENTRIES
```

```
3420.954045954046
```



[91]:

```
x_list=['AVG Morning DE',"AVG Evening DE"]
y_list=[AVG_MORNING_DAILY_ENTRIES,AVG_EVINING_DAILY_ENTRIES]
sns.barplot(x =x_list,y=y_list);
```

Thank you