

```
In [123]: import seaborn as sns  
import pandas as pd
```

```
In [124]: input_file_name_bicicletas = "bicicletas.xlsx"  
input_sheet_name_bicicletas = "bicicletas"
```

```
In [125]: input_file_name_subte = "subte.xlsx"  
input_sheet_name_subte = "subte"
```

```
In [126]: input_file_name_vehiculos = "vehiculos.xlsx"  
input_sheet_name_vehiculos = "vehiculos"
```

```
In [127]: df_bicicletas = pd.read_excel(input_file_name_bicicletas, sheet_name = input_sheet_name_bicicletas)
```

```
In [128]: df_subte = pd.read_excel(input_file_name_subte, sheet_name = input_sheet_name_subte)
```

```
In [129]: df_vehiculos = pd.read_excel(input_file_name_vehiculos, sheet_name = input_sheet_name_vehiculos)
```

```
In [130]: df = pd.concat([df_bicicletas,df_subte,df_vehiculos])
df['TOTAL_ESCALA'] = np.where((df.TIPO == 'SUBTE'),(df.TOTAL/50).astype(int),df.TOTAL)
df
```

Out[130]:

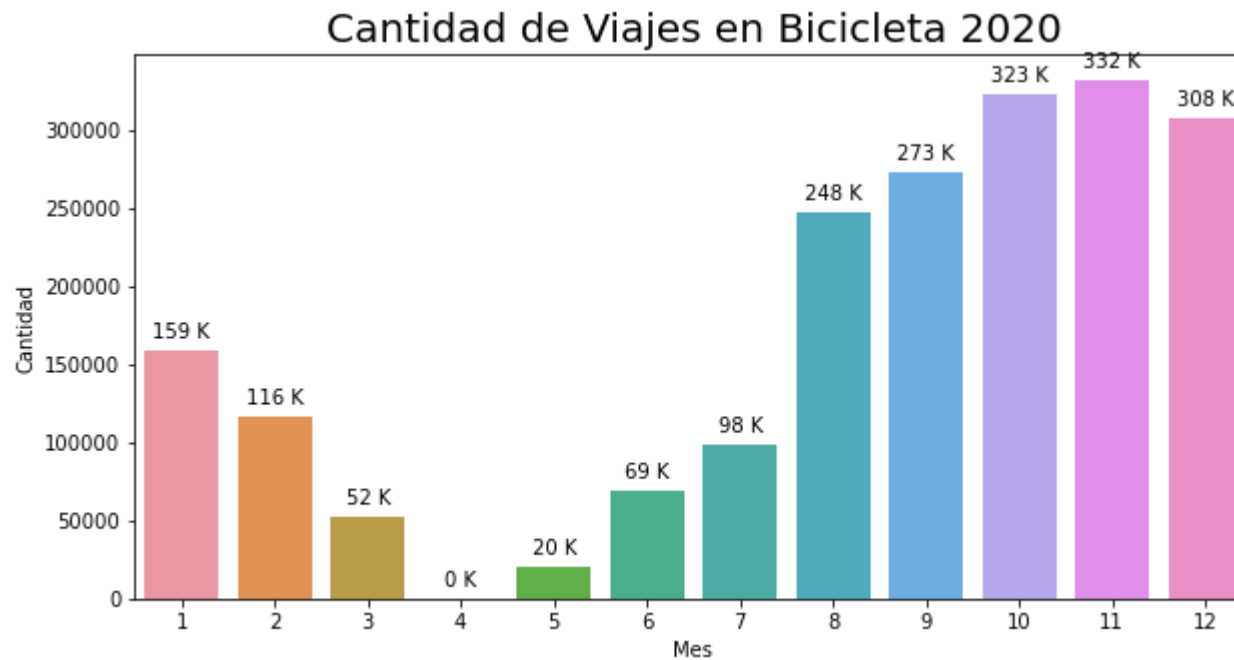
	AÑO	MES	TIPO	TOTAL	TOTAL_ESCALA
0	2020	1	BICICLETAS	159233	159233
1	2020	2	BICICLETAS	116965	116965
2	2020	3	BICICLETAS	52951	52951
3	2020	4	BICICLETAS	0	0
4	2020	5	BICICLETAS	20235	20235
5	2020	6	BICICLETAS	69707	69707
6	2020	7	BICICLETAS	98328	98328
7	2020	8	BICICLETAS	248180	248180
8	2020	9	BICICLETAS	273719	273719
9	2020	10	BICICLETAS	323130	323130
10	2020	11	BICICLETAS	332410	332410
11	2020	12	BICICLETAS	308087	308087
0	2020	1	SUBTE	14431260	288625
1	2020	2	SUBTE	14379523	287590
2	2020	3	SUBTE	16665570	333311
3	2020	4	SUBTE	2532868	50657
4	2020	5	SUBTE	2769479	55389
5	2020	6	SUBTE	3566230	71324
6	2020	7	SUBTE	3417198	68343
7	2020	8	SUBTE	3122951	62459
8	2020	9	SUBTE	3629713	72594
9	2020	10	SUBTE	4966696	99333
10	2020	11	SUBTE	3768015	75360
11	2020	12	SUBTE	4255468	85109
0	2020	1	VEHICULOS	248546	248546

	AÑO	MES	TIPO	TOTAL	TOTAL_ESCALA
1	2020	2	VEHICULOS	232209	232209
2	2020	3	VEHICULOS	197980	197980
3	2020	4	VEHICULOS	143191	143191
4	2020	5	VEHICULOS	171215	171215
5	2020	6	VEHICULOS	212227	212227
6	2020	7	VEHICULOS	209202	209202
7	2020	8	VEHICULOS	213511	213511
8	2020	9	VEHICULOS	219852	219852
9	2020	10	VEHICULOS	226641	226641
10	2020	11	VEHICULOS	222866	222866
11	2020	12	VEHICULOS	236303	236303

```

In [135]: plt.figure(figsize=(10,5))
s = sns.barplot(data = df_bicicletas,
                x = 'MES',
                y = 'TOTAL')
for p in s.patches:
    s.annotate(str(int(p.get_height()/1000)) + ' K',
               (p.get_x() + p.get_width() / 2., p.get_height()),
               ha = 'center', va = 'center',
               size=10,
               xytext = (0, 9),
               textcoords = 'offset points')
plt.title("Cantidad de Viajes en Bicicleta 2020", fontsize = 20)
plt.xlabel('Mes')
plt.ylabel('Cantidad')
plt.show()

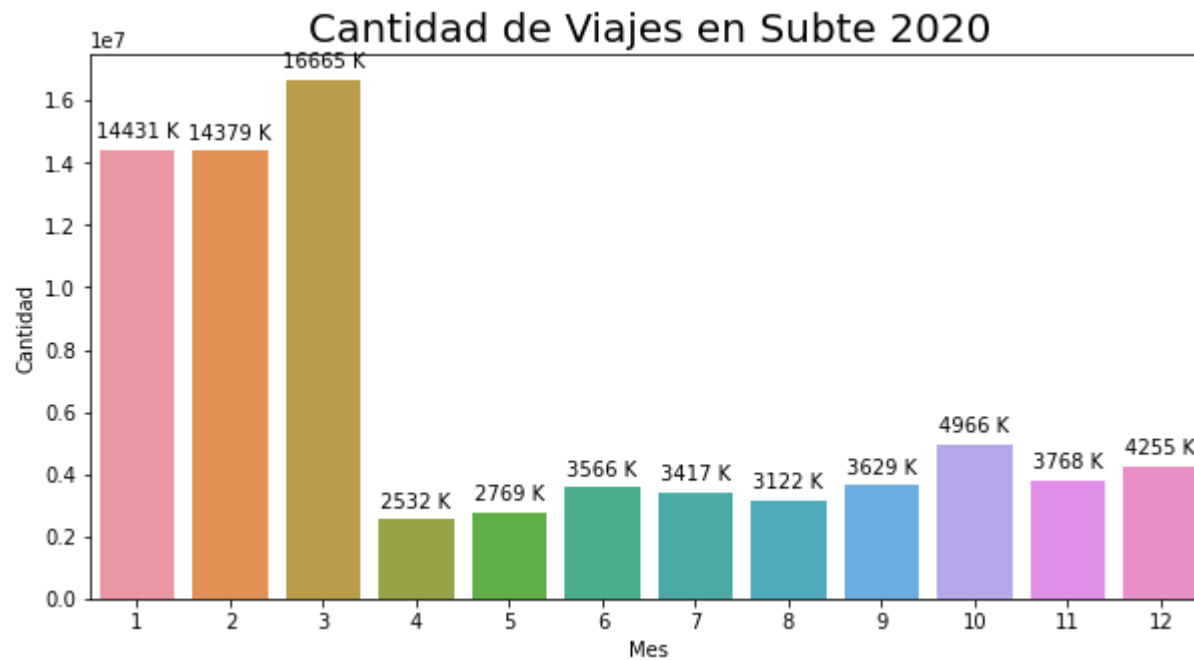
```



```

In [136]: plt.figure(figsize=(10,5))
s = sns.barplot(data = df_subte,
                x = 'MES',
                y = 'TOTAL')
for p in s.patches:
    s.annotate(str(int(p.get_height()/1000)) + ' K',
               (p.get_x() + p.get_width() / 2., p.get_height()),
               ha = 'center', va = 'center',
               size=10,
               xytext = (0, 9),
               textcoords = 'offset points')
plt.title("Cantidad de Viajes en Subte 2020", fontsize = 20)
plt.xlabel('Mes')
plt.ylabel('Cantidad')
plt.show()

```



plt.figure(figsize=(10,5))

```
s = sns.barplot(data = df_vehiculos,
                x = 'MES',
                y = 'TOTAL')
for p in s.patches:
    s.annotate(str(int(p.get_height()/1000)) + ' K',
               (p.get_x() + p.get_width() / 2., p.get_height()),
               ha = 'center', va = 'center',
               size=10,
               xytext = (0, 9),
               textcoords = 'offset points')
plt.title("Cantidad de Viajes en Vehiculos 2020", fontsize = 20)
plt.xlabel('Mes')
plt.ylabel('Cantidad')
plt.show()
```

```

In [138]: plt.figure(figsize=(10,5))
b = sns.lineplot(data = df,
                  x = 'MES',
                  y = 'TOTAL_ESCALA',
                  hue = 'TIPO',
                  style = 'TIPO',
                  markers=True,
                  dashes=True
                )
plt.title("Cantidad de Viajes 2020", fontsize = 20)
plt.xlabel('Mes')
plt.ylabel('Cantidad')
plt.show()

```

