MIDTERM

MATTHEW LANGENDORFER

PART I
FREQUENCY OF TEXT FILE

```python
import json
import matplotlib.pyplot as plt

numlist = []
dict = {}

f = open("/Users/ml/Desktop/midterm-project/numbers.txt", "r")
text = f.read()

for num in text:
    if num.isdigit():
        numlist.append(int(num))

for num in numlist:
    if num in dict.keys():
        dict[num] += 1
    else:
        dict[num] = 1

for key, value in dict.items():
    print(f"{key} : {value}")

plt.bar(list(dict.keys()), dict.values(), color='g')
plt.xlabel("Value")
plt.ylabel("Frequency")
plt.title('Frequency of Text File')
plt.show()

y = json.dumps(dict)
print(y)
with open("out.json", "w") as outfile:
    json.dump(y, outfile)
```

INPUT

```
import json
import matplotlib.pyplot as plt

numlist = []
dict = {}

f = open("/Users/ml/Desktop/midterm-project/numbers.txt", "r")
text = f.read()

for num in text:
    if num.isdigit():
        numlist.append(int(num))

for num in numlist:
    if num in dict.keys():
        dict[num] += 1
    else:
        dict[num] = 1

for key, value in dict.items():
    print(f"{key} : {value}")

plt.bar(list(dict.keys()), dict.values(), color='g')
plt.xlabel("Value")
plt.ylabel("Frequency")
plt.title('Frequency of Text File')
plt.show()

y = json.dumps(dict)
print(y)
with open("out.json", "w") as outfile:
    json.dump(y, outfile)
```

❑ First I imported the modules that I needed.
❑ Isdigit checks to see if the iterator is a number, so that it only takes the numbers out of the text file, then it appends the file to turn the string into an integer.
❑ The first loop takes all the numbers in the text and assigns them to a key, while the second loop adds to the value for each of the keys already established for each number in what is now the list of all the numbers.
❑ Json file is saved as out.json locally.
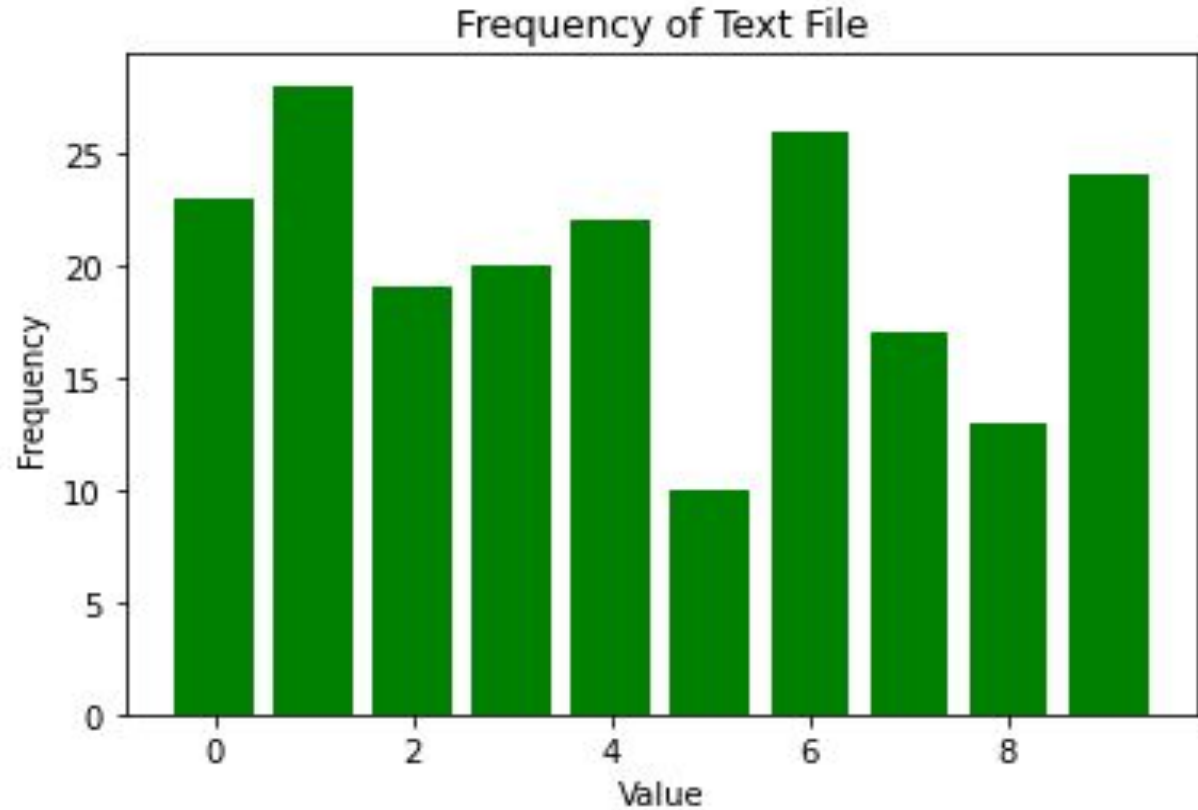
OUTPUT

2 : 19
4 : 22
6 : 26
8 : 13
5 : 10
1 : 28
9 : 24
0 : 23
7 : 17
3 : 20

{"2": 19, "4": 22, "6": 26, "8": 13,
"5": 10, "1": 28, "9": 24, "0": 23,
"7": 17, "3": 20}

Frequency of Text File

PART II

NETFLIX VIEWING HABITS

❑ **USED THE SAMPLE NETFLIX .CSV FILE**

❑ **FOCUSED ON STAR TREK VIEWING HABITS**

❑ <u>**STATS:**</u>

<u>**Total Watchtime:**</u>
Timedelta('0 days 20:00:57')
<u>**Average Watchtime**</u>
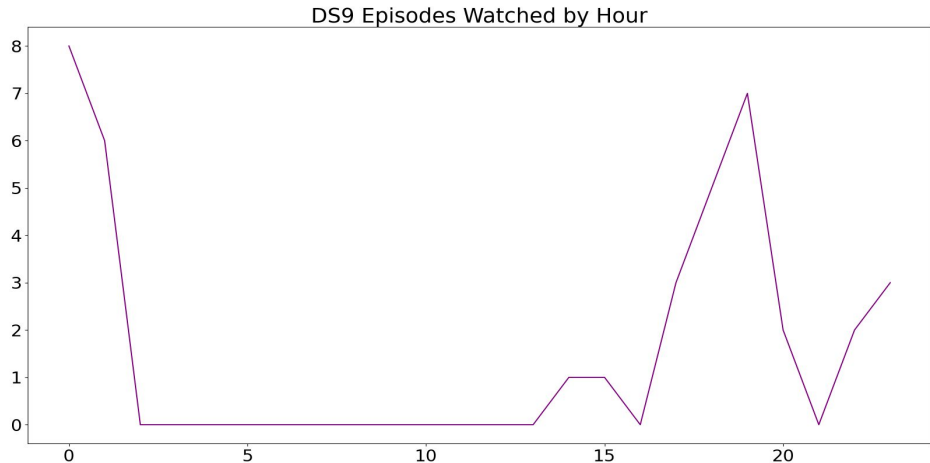**Timedelta('0 days 00:31:36.236842105')**

GRAPHS

1.

```
startrek['hour'] = pd.Categorical(startrek['hour'], categories=

[0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,
23],
    ordered=True)

startrek_by_hour = startrek['hour'].value_counts()

startrek_by_hour = startrek_by_hour.sort_index()

startrek_by_hour.plot(kind='line', color='purple',
figsize=(20,10), title='DS9 Episodes Watched by Hour')
```
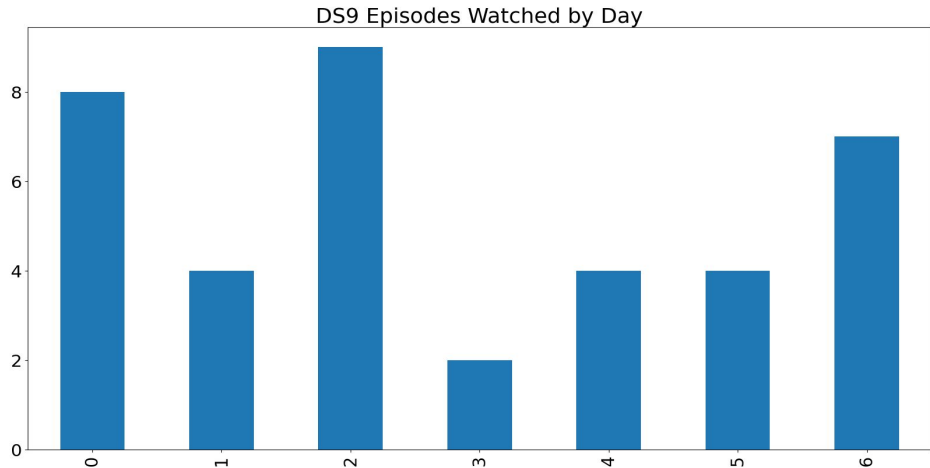
2.

```
startrek['weekday'] = pd.Categorical(startrek['weekday'],
categories=
    [0,1,2,3,4,5,6],
    ordered=True)

startrek_by_day = startrek['weekday'].value_counts()

startrek_by_day = startrek_by_day.sort_index()

matplotlib.rcParams.update({'font.size': 22})

startrek_by_day.plot(kind='bar', figsize=(20,10), title="DS9
Episodes Watched by Day')
```


DS9 Episodes Watched by Hour


DS9 Episodes Watched by Day

PIE

```
startrek['hour'] =
pd.Categorical(startrek['hour'], categories=

[0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,1
8,19,20,21,22,23],
    ordered=True)

startrek_by_hour =
startrek['hour'].value_counts()

startrek_by_hour =
startrek_by_hour.sort_index()

startrek_by_hour.plot(kind='pie',
figsize=(20,10), title='DS9 Episodes Watched
by Hour')
```


DS9 Episodes Watched by Hour
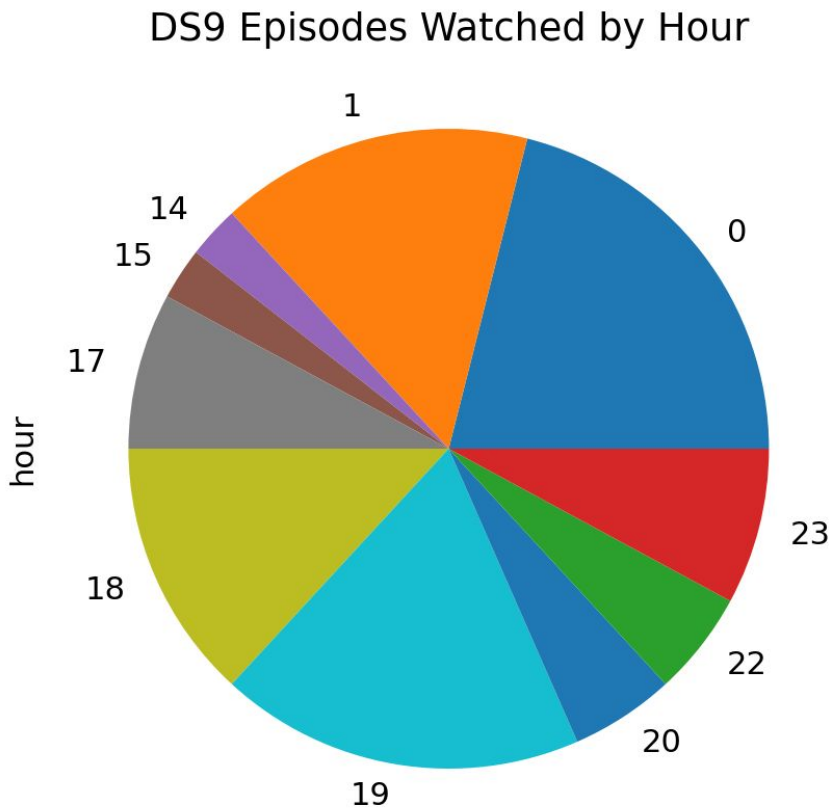
SEABORN

```
import seaborn as sns
palette = sns.color_palette("rocket",
as_cmap=True)
ax = sns.countplot(startrek['weekday'],
label='Count', palette='rocket')
ax.set_title('DS9 Episodes By Day')
ax.set_ylabel('Frequency')
```
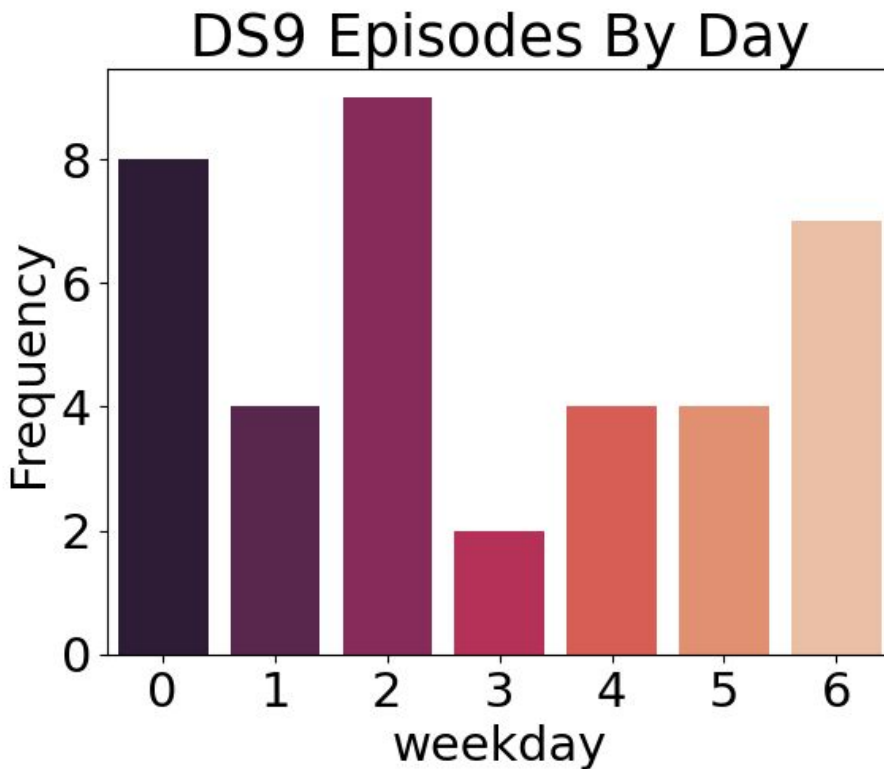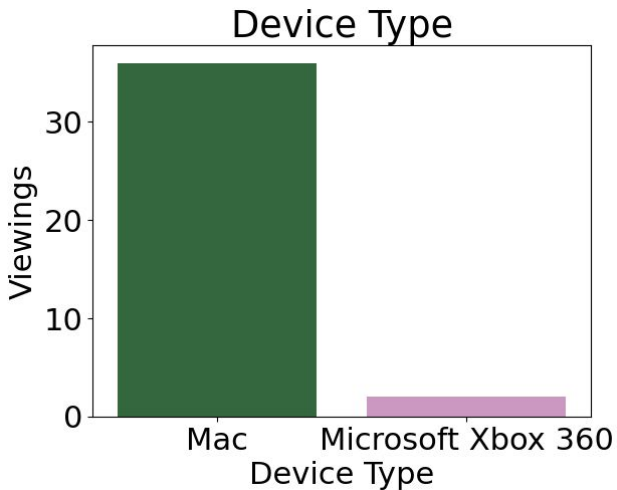
DEVICE TYPE

```
startrek_by_day = startrek['Device
Type'].value_counts()

startrek_by_day =
startrek_by_day.sort_index()
print(startrek_by_day)

sns.color_palette("cubehelix",
as_cmap=True)
ax = sns.countplot(startrek['Device Type'],
label='Count', palette='cubehelix')
ax.set_title('Device Type')
ax.set_ylabel('Viewings')
```



```
Mac                      36
Microsoft Xbox 360        2
Name: Device Type, dtype: int64
```

GITHUB LINK

https://github.com/dorf3/Midterm