Zeru-Zhou-project2

January 22, 2022

1 Project 2 – Zeru Zhou

TA Help: NA

Collaboration: NA

- Get help from Dr. Ward's videos

1.1 Question 1

[6]:
```
import pandas as pd
```

[7]:
```
df = pd.read_csv('/depot/datamine/data/noaa/2020_sample.csv',␣
↪names=["station_id", "date", "element_code", "value", "mflag", "qflag",␣
↪"sflag", "obstime"])
```

[4]:
```
df.head(10)
```

[4]:

	station_id	date	element_code	value	mflag	qflag	sflag	obstime
0	AE000041196	20200101	TMIN	168	NaN	NaN	S	NaN
1	AE000041196	20200101	PRCP	0	D	NaN	S	NaN
2	AE000041196	20200101	TAVG	211	H	NaN	S	NaN
3	AEM00041194	20200101	PRCP	0	NaN	NaN	S	NaN
4	AEM00041194	20200101	TAVG	217	H	NaN	S	NaN
5	AEM00041217	20200101	TAVG	205	H	NaN	S	NaN
6	AEM00041218	20200101	TMIN	148	NaN	NaN	S	NaN
7	AEM00041218	20200101	TAVG	199	H	NaN	S	NaN
8	AFM00040938	20200101	PRCP	23	NaN	NaN	S	NaN
9	AFM00040938	20200101	TAVG	54	H	NaN	S	NaN

The first 10 rows are provided. Obviously this is much easier than the for loop because this is only a one-line command, extremely easy to think about.

1.2 Question 2

[6]:
```
df.shape
```

[6]:
```
(15000000, 8)
```

[7]:
```
print(f'There are {df.shape[1]} columns in the DataFrame!')
```

```
There are 8 columns in the DataFrame!
```

[8]:
```
print(f'There are {df.shape[0]} rows in the DataFrame!')
```

```
There are 15000000 rows in the DataFrame!
```

There are 8 columns and 15000000 rows in the dataframe.

1.3 Question 3

[10]:
```
my_dict = {"fruits": ["apple", "orange", "pear"], "person": "John",␣
 ↪"vegetables": ["carrots", "peas"]}

# If "person" is indeed a key, they will function the same way
my_dict["person"]
```

[10]:
```
'John'
```

[11]:
```
my_dict.get("person")
```

[11]:
```
'John'
```

[14]:
```
my_dict.get("Same")
```

[15]:
```
my_dict["Same"]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-15-b2604a9aae85> in <module>
----> 1 my_dict["Same"]

KeyError: 'Same'
```

[9]:
```
station_ids = df["station_id"].dropna().tolist()
```

[3]:
```
my_dict = {}
```

[10]:
```
Unique1 = list(set(station_ids))
```

[11]:
```
for i in Unique1:
    my_dict[i] = 0
```

[12]:
```
for j in station_ids:
    my_dict[j]+=1
```

[20]:
```
print(my_dict['US1MANF0058'])
```

```
378
```

[25]:
```
print(my_dict['USW00023081'])
```

```
1290
```

[26]:
```
print(my_dict['US10sali004'])
```

```
13
```

"get" function and braskets normally works the same, but when searching for a non-exist key, brackets would show an error but get method won't. The dictionary my_dict is designed as above.

1.4 Question 4

[27]:
```
df_intruder = pd.read_csv('/depot/datamine/data/noaa/2020_sampleB.csv',␣
↪names=["station_id", "date", "element_code", "value", "mflag", "qflag",␣
↪"sflag", "obstime"])
```

[29]:
```
intruder_ids = df_intruder["station_id"].dropna().tolist()
```

[31]:
```
Unique2 = list(set(intruder_ids))
```

[34]:
```
for i in Unique2:
    if i not in Unique1:
        print(i)
```

```
USFAKEROW22
```

[39]:
```
df_intruder[df_intruder["station_id"] == "USFAKEROW22" ]
```

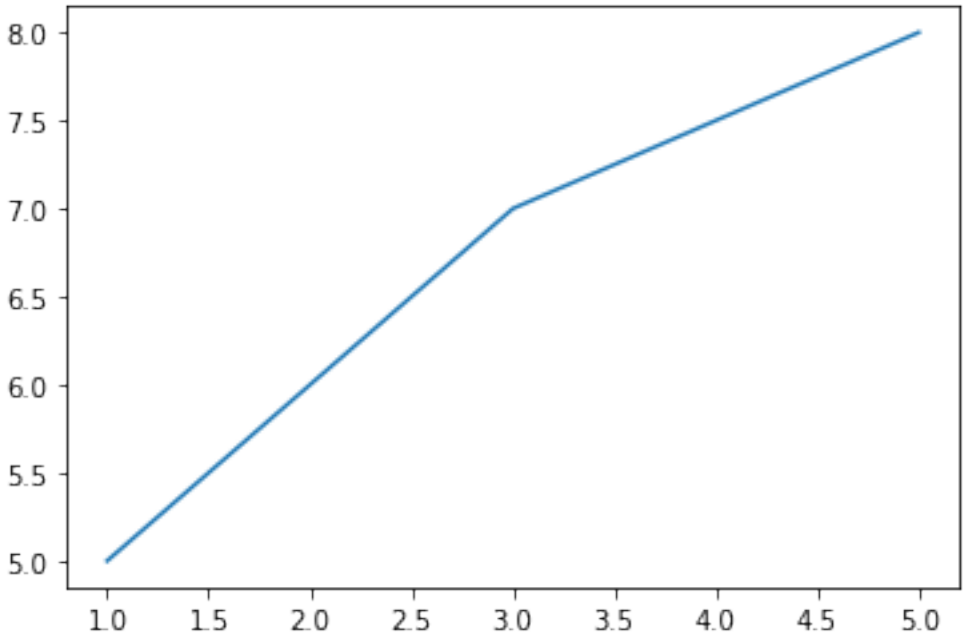
[39]:

	station_id	date	element_code	value	mflag	qflag	sflag	obstime
13002032	USFAKEROW22	20200516	PRCP	0	NaN	NaN	N	NaN

The intruder row is printed.

1.5 Question 5

[1]:
```
import matplotlib.pyplot as plt
plt.plot([1,2,3,5],[5,6,7,8])
plt.show()
plt.close()
```



[55]:
```
new_dict = {}
```

[56]:
```
q_flag = df["qflag"].dropna().tolist()
```

[57]:
```
Unique3 = list(set(q_flag))
```

[58]:
```
for i in Unique3:
    new_dict[i]=0
```

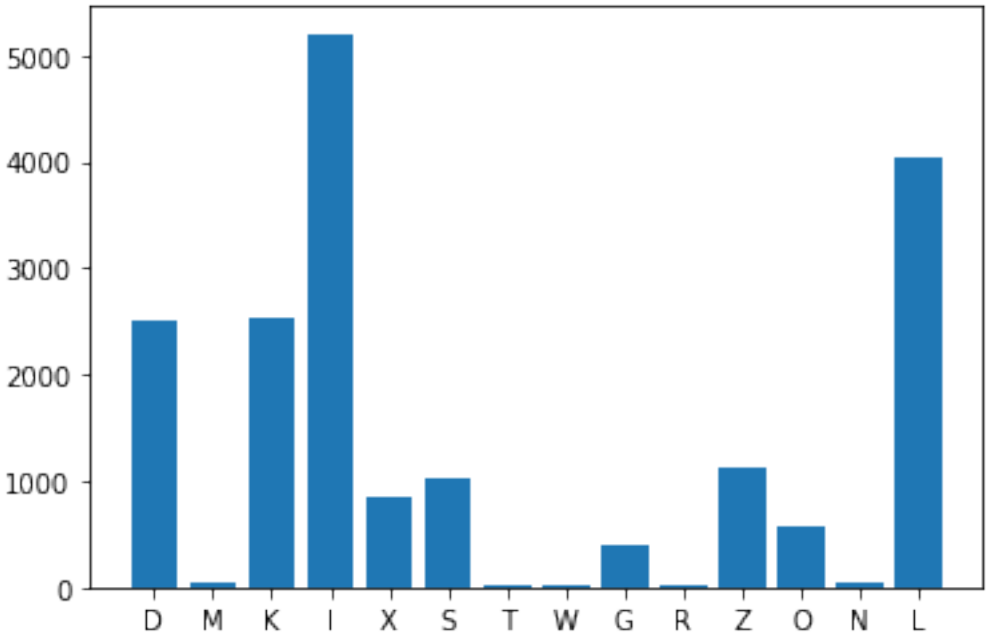
[59]:
```
for j in q_flag:
    new_dict[j]+=1
```

[16]:
```
import itertools
```

[60]:
```
Sliced = dict(itertools.islice(new_dict.items(),20))
```

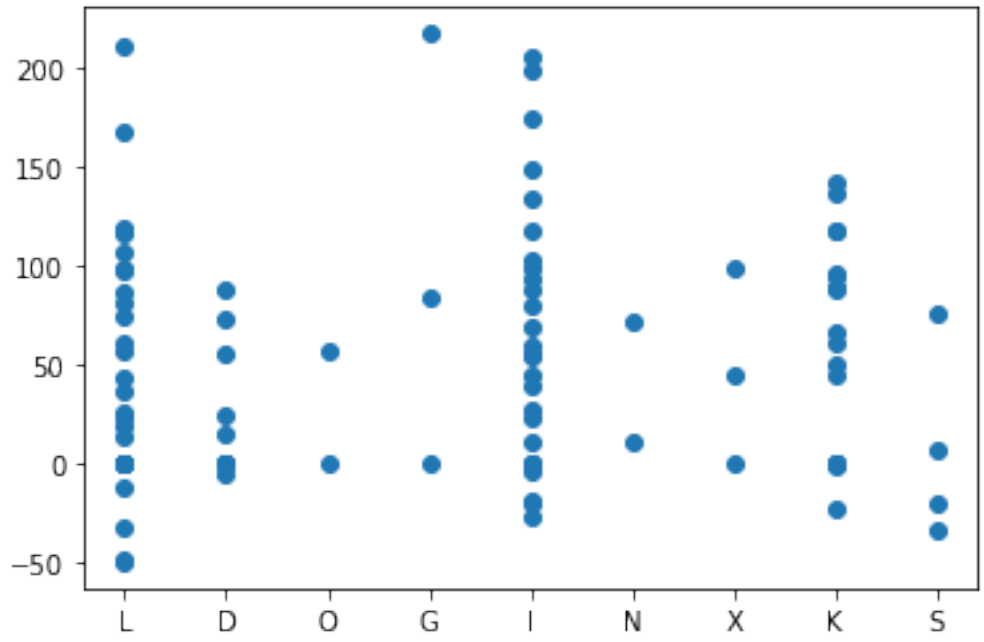
[61]:
```
plt.bar(Sliced.keys(), Sliced.values())
```

[61]: <BarContainer object of 14 artists>



[21]:
```
plt.scatter(df["qflag"].dropna().head(n=100) ,df["value"].head(n=100)) # Check
↪the relationship between qflag and value
```

[21]: <matplotlib.collections.PathCollection at 0x2b9c44216eb0>



First, I used itertools to slice the dictonary, and created a new dictonary of qflag. Then I draw the first 20 qflags and see how many times they appear in the full data set respectively. Then, I draw a scatter plot between values and qflag.

1.6 Pledge

By submitting this work I hereby pledge that this is my own, personal work. I've acknowledged in the designated place at the top of this file all sources that I used to complete said work, including but not limited to: online resources, books, and electronic communications. I've noted all collaboration with fellow students and/or TA's. I did not copy or plagiarize another's work.

> As a Boilermaker pursuing academic excellence, I pledge to be honest and true in all that I do. Accountable together – We are Purdue.