

```
In [2]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set_style('whitegrid')
%matplotlib inline
```

```
In [3]: df = pd.read_csv('911.csv')
```

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

Out[4]:

	lat	lng	desc	zip	title	timeStamp	
0	40.297876	-75.581294	REINDEER CT & DEAD END; NEW HANOVER; Station ...	19525.0	EMS: BACK PAINS/INJURY	2015-12-10 17:40:00	NEW HANOVER
1	40.258061	-75.264680	BRIAR PATH & WHITEMARSH LN; HATFIELD TOWNSHIP...	19446.0	EMS: DIABETIC EMERGENCY	2015-12-10 17:40:00	HATFIELD TOWNSHIP
2	40.121182	-75.351975	HAWS AVE; NORRISTOWN; 2015-12-10 @ 14:39:21-St...	19401.0	Fire: GAS- ODOR/LEAK	2015-12-10 17:40:00	NORRISTO
3	40.116153	-75.343513	AIRY ST & SWEDE ST; NORRISTOWN; Station 308A;...	19401.0	EMS: CARDIAC EMERGENCY	2015-12-10 17:40:01	NORRISTO

	lat	lng	desc	zip	title	timeStamp	
4	40.251492	-75.603350	CHERRYWOOD CT & DEAD END; LOWER POTTSGROVE; S...	NaN	EMS: DIZZINESS	2015-12-10 17:40:01	LOWER POTTSGROVE



```
In [5]: df['Reason'] = df['title'].apply(lambda title: title.split(':')[0])
```

```
In [16]: df = df[['lat', 'lng', 'desc', 'zip', 'Reason', 'title', 'timeStamp', 'twp', 'address', 'e']]
```

```
In [17]: #First 5 rows of 911
```

```
In [23]: df.head()
```

Out[23]:

	lat	lng	desc	zip	Reason	title	timeStamp	
0	40.297876	-75.581294	REINDEER CT & DEAD END; NEW HANOVER; Station ...	19525.0	EMS	EMS: BACK PAINS/INJURY	2015-12-10 17:40:00	NI H...
1	40.258061	-75.264680	BRIAR PATH & WHITEMARSH LN; HATFIELD TOWNSHIP...	19446.0	EMS	EMS: DIABETIC EMERGENCY	2015-12-10 17:40:00	H/ TC

	lat	lng	desc	zip	Reason	title	timeStamp	
2	40.121182	-75.351975	HAWS AVE; NORRISTOWN; 2015-12-10 @ 14:39:21-St...	19401.0	Fire	Fire: GAS- ODOR/LEAK	2015-12-10 17:40:00	No
3	40.116153	-75.343513	AIRY ST & SWEDE ST; NORRISTOWN; Station 308A;...	19401.0	EMS	EMS: CARDIAC EMERGENCY	2015-12-10 17:40:01	No
4	40.251492	-75.603350	CHERRYWOOD CT & DEAD END; LOWER POTTSGROVE; S...	NaN	EMS	EMS: DIZZINESS	2015-12-10 17:40:01	LC PC

In [24]: *#Top 5 Zipcodes*

In [26]: `df['zip'].value_counts().head(5)`

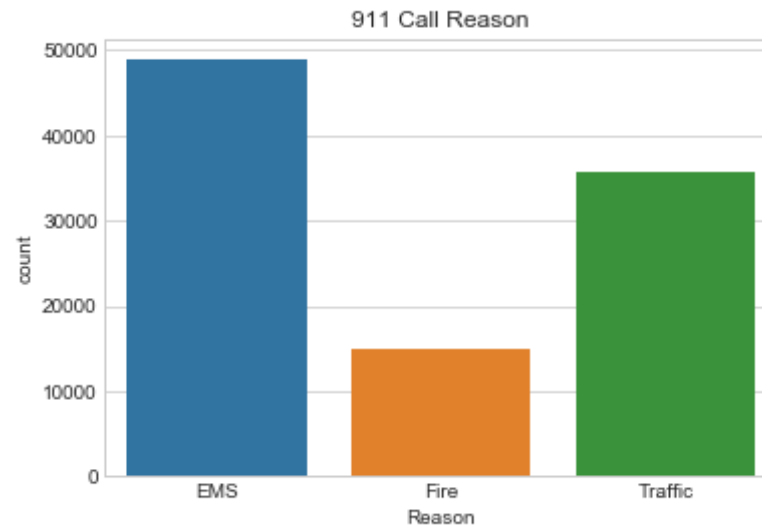
Out[26]:

19401.0	6979
19464.0	6643
19403.0	4854
19446.0	4748
19406.0	3174

Name: zip, dtype: int64

In [28]: `sns.countplot(x='Reason',data=df)`
`plt.title('911 Call Reason')`

Out[28]: `Text(0.5,1,'911 Call Reason')`



```
In [13]: #Most common reason for 911 calls
```

```
In [11]: df['Reason'].value_counts()
```

```
Out[11]: EMS      48877  
Traffic   35695  
Fire      14920  
Name: Reason, dtype: int64
```

```
In [35]: df['timeStamp'] = pd.to_datetime(df['timeStamp'])
```

```
In [74]: time = df['timeStamp'][0]  
time
```

```
Out[74]: Timestamp('2015-12-10 17:40:00')
```

```
In [75]: #Convert timestamp to objects
```

```
In [76]: df['Hour'] = df['timeStamp'].apply(lambda time: time.hour)  
df['Month'] = df['timeStamp'].apply(lambda time: time.month)
```

```
df['Day of Week'] = df['timeStamp'].apply(lambda time: time.dayofweek)
```

```
In [77]: #turn day integer to string names
```

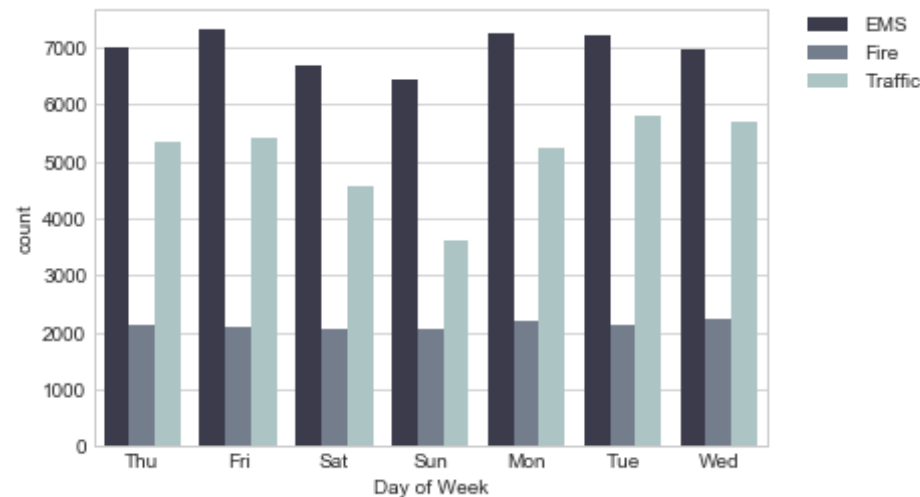
```
In [78]: dmap = {0: 'Mon', 1: 'Tue', 2: 'Wed', 3: 'Thu', 4: 'Fri', 5: 'Sat', 6: 'Sun'}
```

```
In [80]: df['Day of Week'] = df['Day of Week'].map(dmap)
```

```
In [81]: #countplot day of week
```

```
In [90]: sns.countplot(x='Day of Week', data=df, hue='Reason', palette='bone')  
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

```
Out[90]: <matplotlib.legend.Legend at 0x255ca4412b0>
```



```
In [91]: #Month
```

```
In [89]: sns.countplot(x='Month', data=df, hue='Reason', palette='bone')  
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

```
Out[89]: <matplotlib.legend.Legend at 0x255c9dbaa20>
```

