Assignment 6

Name: [Mukesh Kumar Mahto]

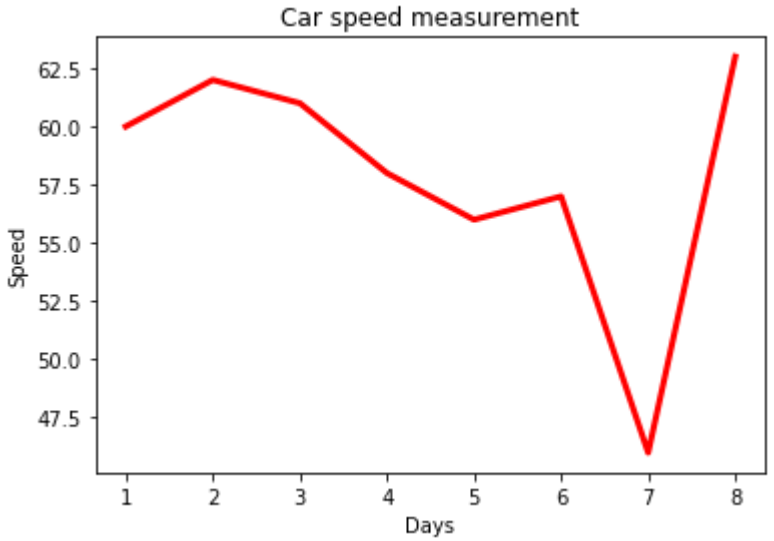
Reg_ID: GO_STP_9639

visualization using Python Matplotlib

1. Load the necessary package for plotting using pyplot from matplotlib. Example - Days(x-axis) represents 8 days and Speed represents a car's speed. Plot a Basic line plot between days and car speed, put x axis label as days and y axis label as car speed and put title Car Speed Measurement.

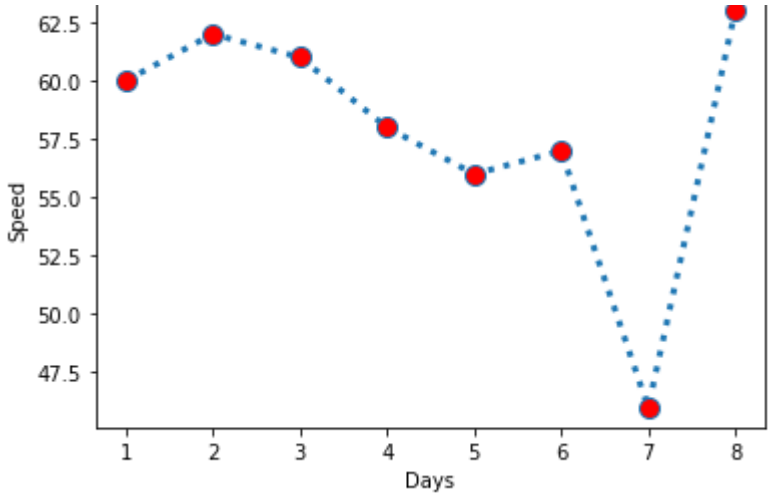
```
Days=[1,2,3,4,5,6,7,8]
Speed=[60,62,61,58,56,57,46,63]
```

```
import matplotlib.pyplot as plt
Days=[1,2,3,4,5,6,7,8]
Speed=[60,62,61,58,56,57,46,63]
plt.plot(Days, Speed, 'r', linewidth=3)
plt.xlabel("Days")
plt.ylabel("Speed")
plt.title("Car speed measurement")
plt.show()
```

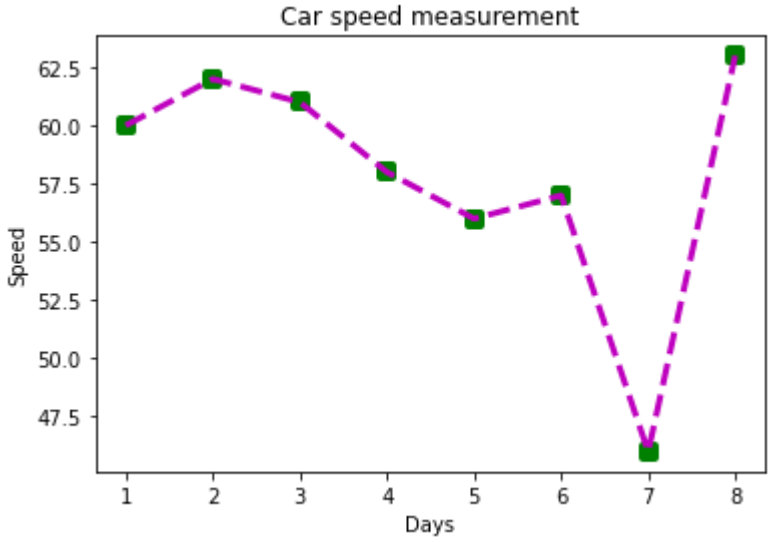


2. Now to above car data apply some string formats like line style example green dotted line, marker shape like +, change markersize, markerface color etc.

✓ 0s completed at 2:35 PM




```
plt.plot(Days, Speed, linestyle='--', color='m', linewidth=3)
plt.scatter(Days, Speed, marker="s", color='g', linewidth=4)
plt.xlabel("Days")
plt.ylabel("Speed")
plt.title("Car speed measurement")
plt.show()
```



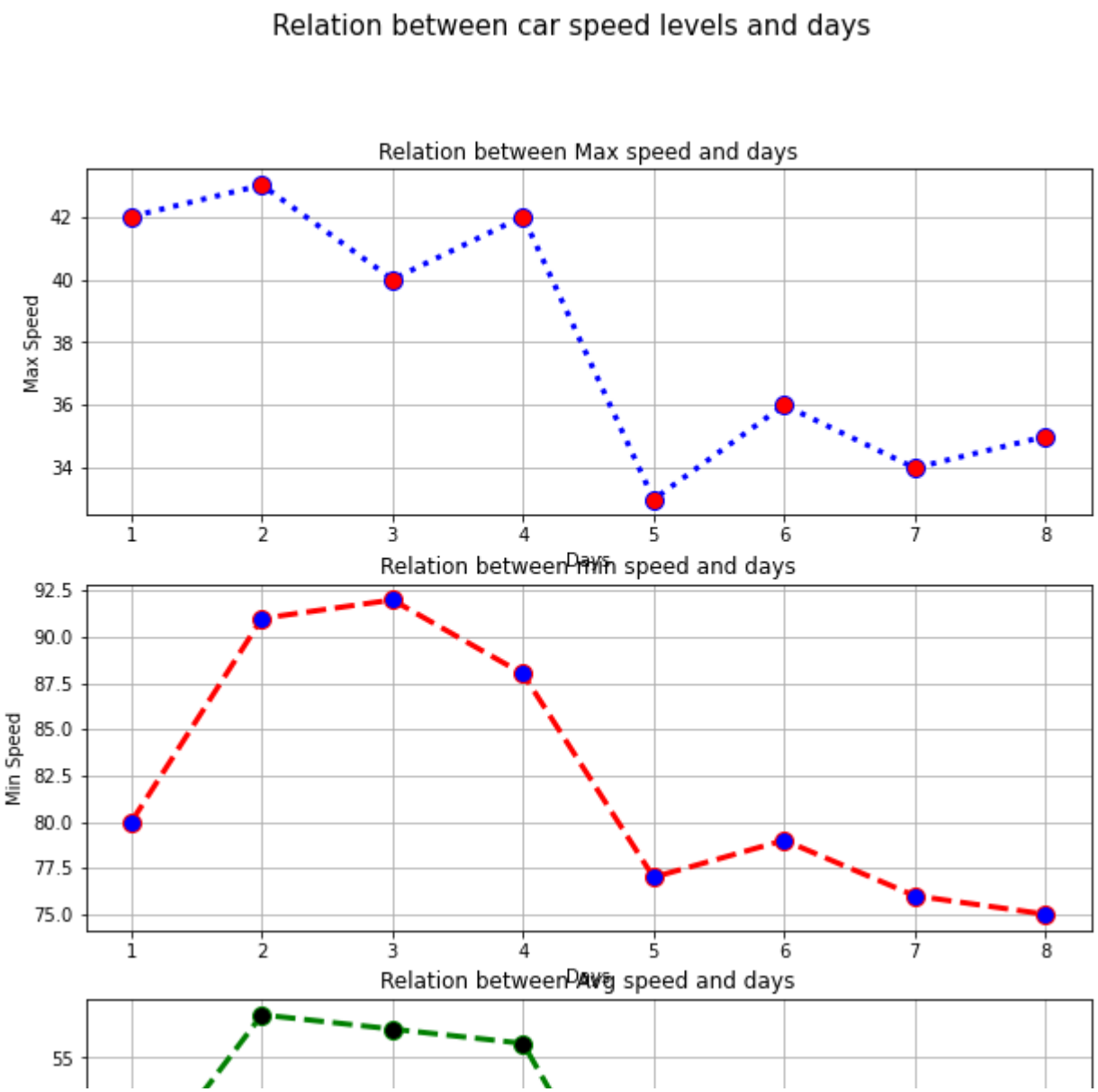
3. Plot Axes Labels, Chart title, Legend, Grid in Car minimum, Maximum and average speed in 8 days.

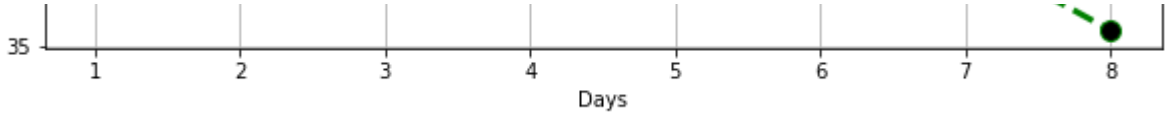
```
days=[1,2,3,4,5,6,7,8]
max_speed=[80,91,92,88,77,79,76,75]
min_speed=[42,43,40,42,33,36,34,35]
avg_speed=[46,58,57,56,40,42,41,36]
```

```
#minimum speed vs days
ax1.plot(days, min_speed, 'b:', linewidth=3, marker="o", markerfacecolor='r', ma
ax1.set_xlabel("Days")
ax1.set_ylabel("Max Speed")
ax1.set_title("Relation between Max speed and days")
ax1.grid()

#maximum speed vs days
ax2.plot(days, max_speed, 'r--', linewidth=3,marker="o",markerfacecolor='b',mark
ax2.set_xlabel("Days")
ax2.set_ylabel("Min Speed")
ax2.set_title("Relation between min speed and days")
ax2.grid()

#average speed vs days
ax3.plot(days, avg_speed, 'g--', linewidth=3,marker="o",markerfacecolor='k',mark
ax3.set_xlabel("Days")
ax3.set_ylabel("Avg Speed")
ax3.set_title("Relation between Avg speed and days")
ax3.grid()
```



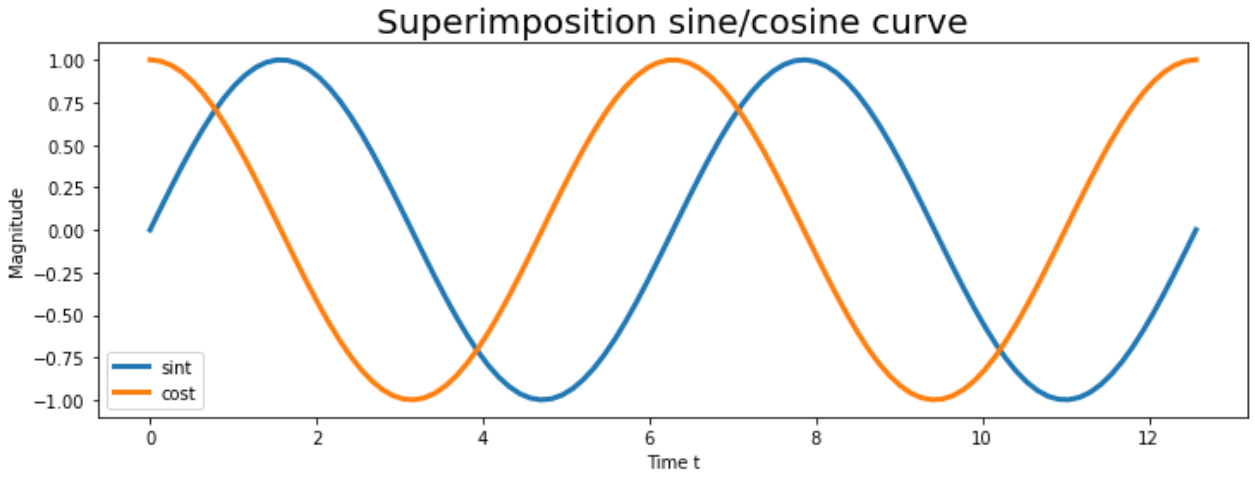


4. Plotting a basic sine graph by adding more features. Adding Multiple plots by Superimposition like cosine wave.

```
import numpy as np
t = np.linspace(0,4*np.pi,100)
sint = np.sin(t)
cost = np.cos(t)

plt.figure(figsize=(12,4))
plt.plot(t,sint,label="sint",linewidth=3)
plt.plot(t,cost,label="cost",linewidth=3)
plt.legend()
plt.xlabel("Time t")
plt.ylabel("Magnitude")
plt.title("Superimposition sine/cosine curve", fontsize=20)
```

```
Text(0.5, 1.0, 'Superimposition sine/cosine curve')
```

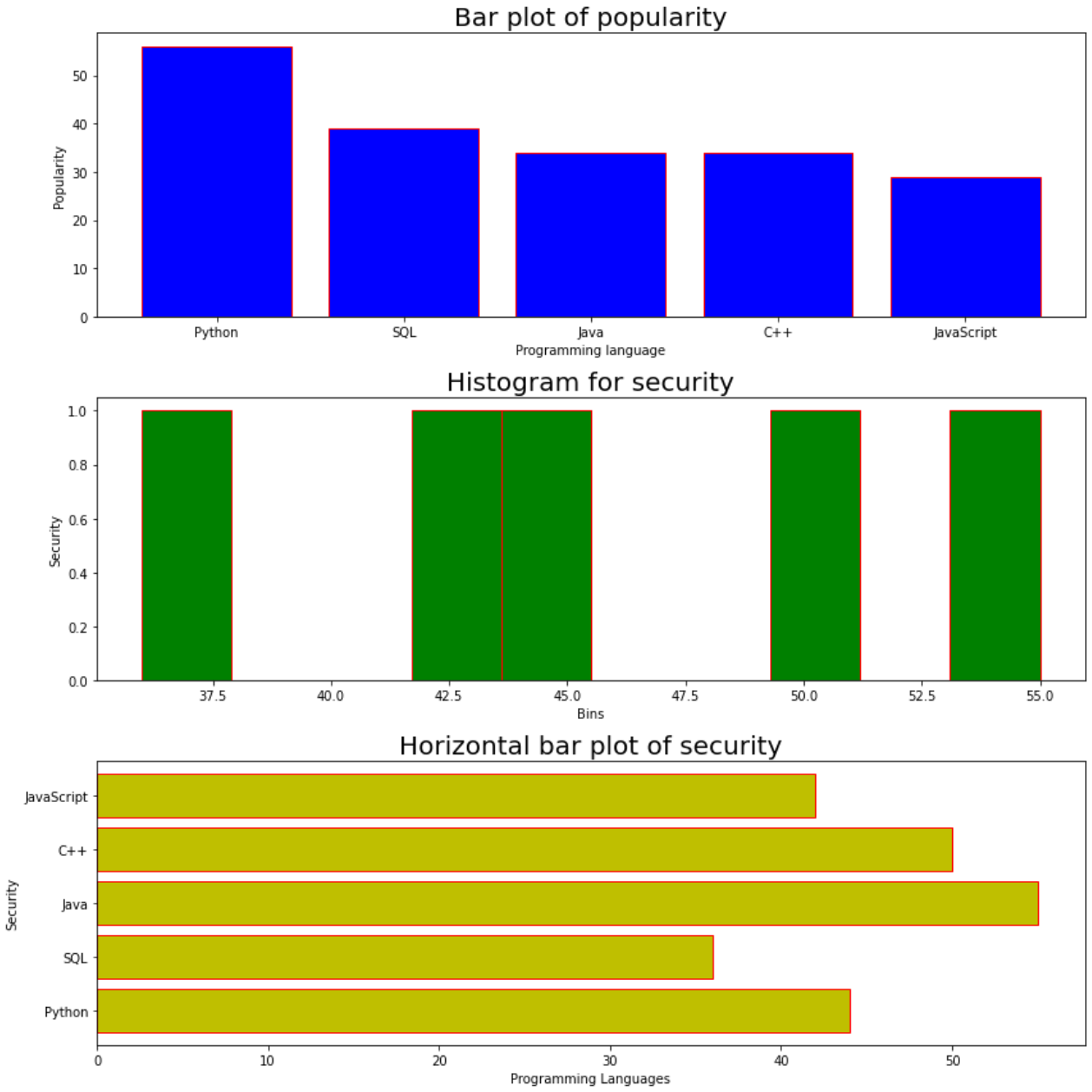


```
ax1.set_title("Bar plot of popularity",fontsize=20)
ax1.set_xlabel("Programming language")
ax1.set_ylabel("Popularity")


ax2.hist(Security, color="g", edgecolor="r")
ax2.set_title("Histogram for security",fontsize=20)
ax2.set_xlabel("Bins")
ax2.set_ylabel("Security")

ax3.barh(Languages, Security, color="y", edgecolor="r")
ax3.set_title("Horizontal bar plot of security",fontsize=20)
ax3.set_xlabel("Programming Languages")
ax3.set_ylabel("Security")

# fig.suptitle("Analysis of programming languages",fontsize=25)
fig.tight_layout()
```



6. Plot Histogram, We have a sample data of Students marks of various Students, we will try to plot number of Students by marks range and try to figure out how many Students are average, below-average and Excellent.

```
Marks = [ 61,86,42,46,73,95,65,78,53,92,55,69,70,49,72,86,64]
```

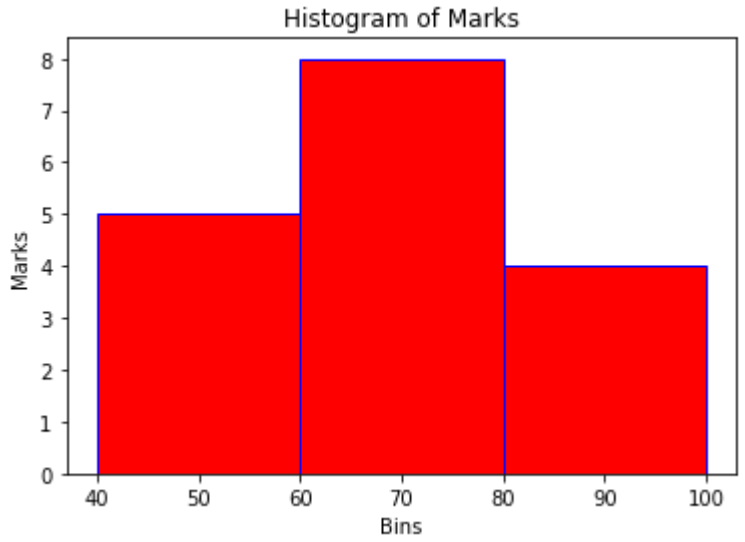
Histogram showing Below Average, Average and Execellent distribution

40-60: Below Average

60-80: Average

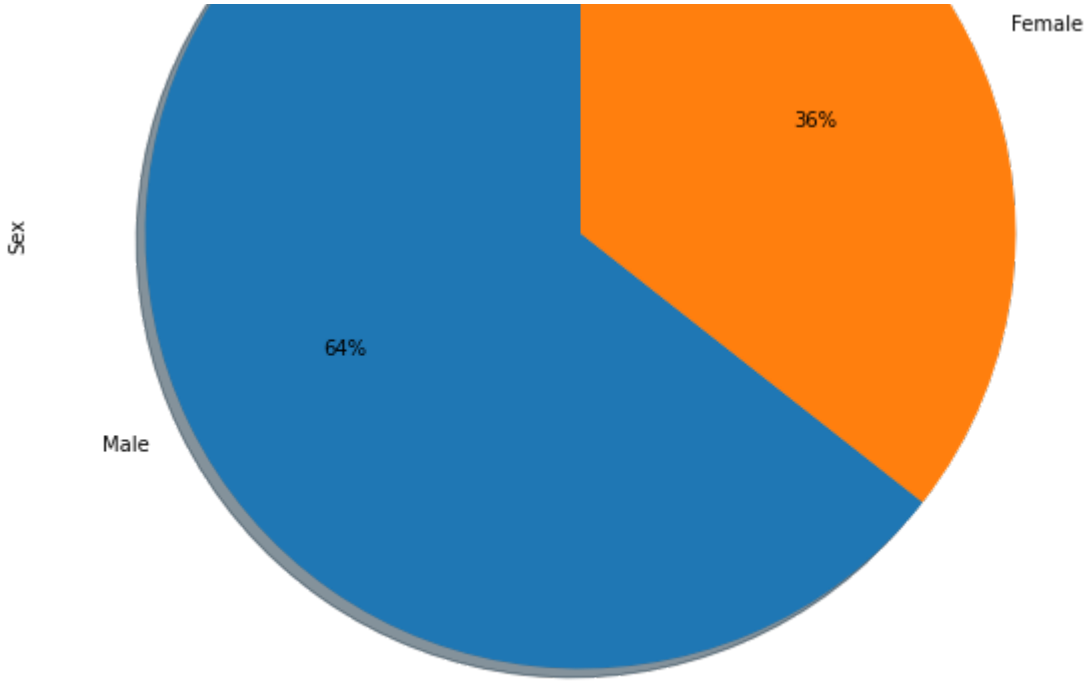
80-100: Excellent

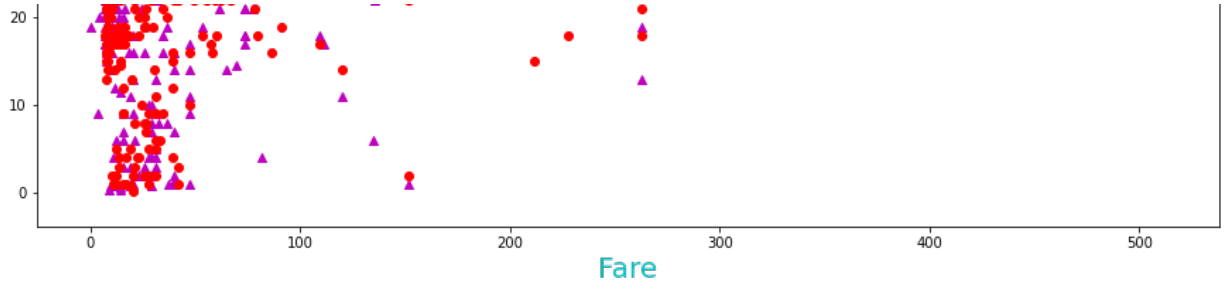
```
plt.show()
```



4	1.0	0.0	(Bessie Waldo Daniels)	female	25.0000	1.0	2.0	113781	15
5	1.0	1.0	Anderson, Mr. Harry	male	48.0000	0.0	0.0	19952	2
6	1.0	1.0	Andrews, Miss. Kornelia Theodosia	female	63.0000	1.0	0.0	13502	7

Andrews, Mr.


Female
36%
Sex
64%
Male


20
10
0
0
100
200
300
400
500
Fare