Matplot

Libraries

```
In [1]:
   ...: import matplotlib.pyplot as plt
   ...: import numpy as np
   ...: import pandas as pd
```

matplotlib. colors

- b : blue.
- **g : green**.
- r : red.
- c : cyan.
- m : magenta.
- y : yellow.
- k : black.
- w : white.

matplotlib.org › api › colors_api

colors — Matplotlib 2.0.2 documentation


black
gray
silver
whitesmoke
rosybrown
firebrick
red
darksalmon
sienna
sandybrown
bisque
tan
moccasin
floralwhite
gold
darkkhaki
lightgoldenrodyellow
olivedrab
chartreuse
palegreen
darkgreen
seagreen
mediumspringgreen
lightseagreen
paleturquoise
darkcyan
darkturquoise
deepskyblue
aliceblue
slategray
royalblue
navy
blue
mediumpurple
darkorchid
plum
m
mediumvioletred
palevioletred
k
grey
lightgray
w
lightcoral
maroon
mistyrose
coral
seashell
peachpuff
darkorange
navajowhite
orange
darkgoldenrod
lemonchiffon
ivory
olive
yellowgreen
lawngreen
lightgreen
g
mediumseagreen
mediumaquamarine
mediumturquoise
darkslategray
c
cadetblue
skyblue
dodgerblue
slategrey
ghostwhite
darkblue
slateblue
rebeccapurple
darkviolet
violet
fuchsia
deeppink
crimson
dimgray
darkgray
lightgrey
white
indianred
darkred
salmon
orangered
chocolate
peru
burlywood
blanchedalmond
wheat
goldenrod
khaki
beige
y
darkolivegreen
honeydew
forestgreen
green
springgreen
aquamarine
azure
darkslategrey
aqua
powderblue
lightskyblue
lightslategrey
lightsteelblue
lavender
mediumblue
darkslateblue
blueviolet
mediumorchid
purple
magenta
hotpink
pink
dimgrey
darkgrey
gainsboro
snow
brown
r
tomato
lightsalmon
saddlebrown
linen
antiquewhite
papayawhip
oldlace
cornsilk
palegoldenrod
lightyellow
yellow
greenyellow
darkseagreen
limegreen
lime
mintcream
turquoise
lightcyan
teal
cyan
lightblue
steelblue
lightslategrey
cornflowerblue
midnightblue
b
mediumslateblue
indigo
thistle
darkmagenta
orchid
lavenderblush
lightpink

To plot symbols rather than lines, provide an additional string argument.

symbols	- , –, -., , . , , , , o , ^ , v , < , > , s , + , x , D , d , 1 , 2 , 3 , 4 , h , H , p , | , _
colors	b, g, r, c, m, y, k, w

Color codes

Character	Color
‘b’	Blue
‘g’	Green
‘r’	Red
‘b’	Blue
‘c’	Cyan
‘m’	Magenta
‘y’	Yellow
‘k’	Black
‘b’	Blue
‘w’	White

Marker codes

Character	Description
'.'	Point marker
'o'	Circle marker
'x'	X marker
'D'	Diamond marker
'H'	Hexagon marker
's'	Square marker
'+'	Plus marker

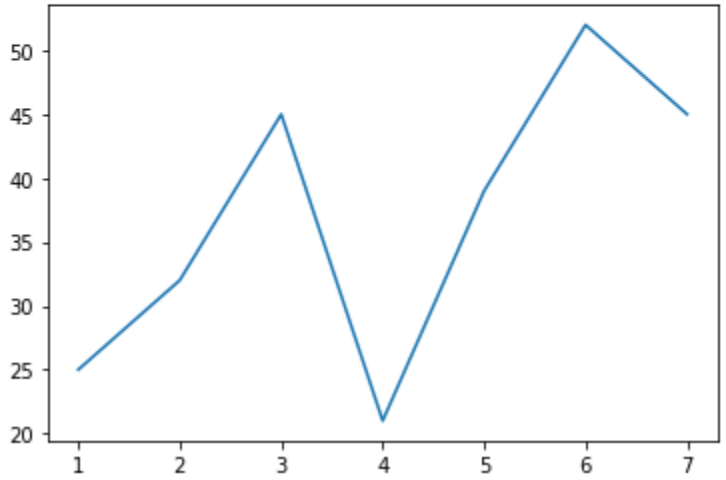
Line styles

Character	Description
'-'	Solid line
'—'	Dashed line
'-.'	Dash-dot line
':'	Dotted line
'H'	Hexagon marker

Line Chart

```
6   x = [1,2,3,4,5,6,7] # days
7   y = [25, 32, 45, 21, 39, 52, 45] # sales
8   plt.plot(x,y)
```

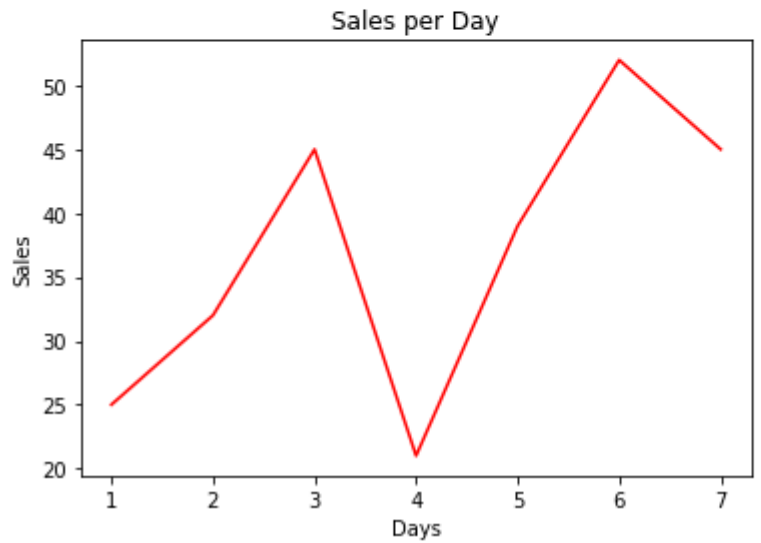
```
In [2]: x = [1,2,3,4,5,6,7] # days

In [3]: y = [25, 32, 45, 21, 39, 52, 45] # sales

In [4]: plt.plot(x,y)
Out[4]: [<matplotlib.lines.Line2D at 0x196306a9e08>]
```



```
10  #You can set the plot title, and labels for x and y axes.
11  # run as a block
12  x = [1,2,3,4,5,6,7] # days
13  y = [25, 32, 45, 21, 39, 52, 45] # sales
14  plt.plot(x,y, 'r') # red
15  plt.xlabel("Days")
16  plt.ylabel("Sales")
17  plt.title('Sales per Day')
18  plt.show()
19
```

Line Chart

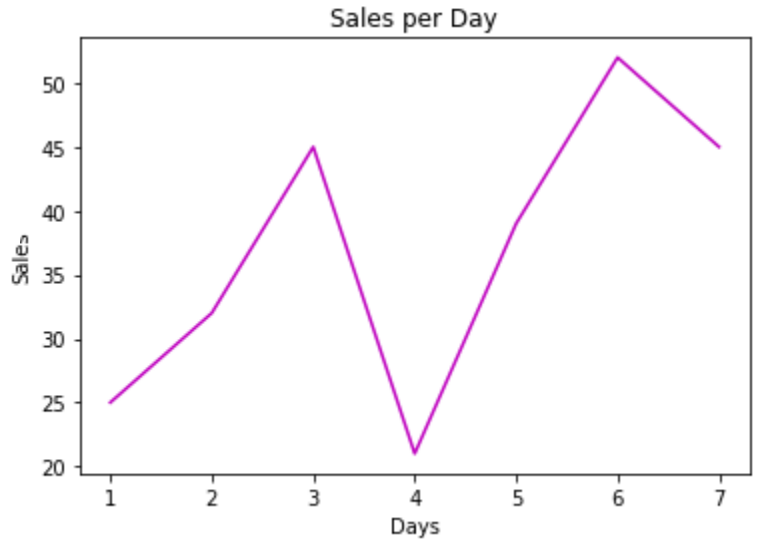
```
In [5]:
   ...:
   ...: x = [1,2,3,4,5,6,7] # days
   ...: y = [25, 32, 45, 21, 39, 52, 45] # sales
   ...: plt.plot(x,y, 'r') # red
   ...: plt.xlabel("Days")
   ...: plt.ylabel("Sales")
   ...: plt.title('Sales per Day')
   ...: plt.show()
```



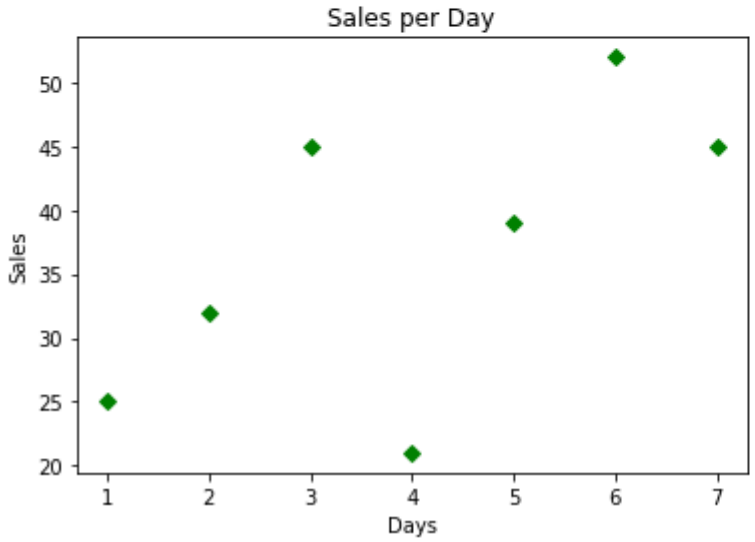
Line Chart

```
20  #__________colors
21  x = [1,2,3,4,5,6,7] # days
22  y = [25, 32, 45, 21, 39, 52, 45] # sales
23  plt.plot(x,y, 'm') # magneta
24  plt.xlabel("Days")
25  plt.ylabel("Sales")
26  plt.title('Sales per Day')
27  plt.show()
28
```

```
In [6]:
   ...: x = [1,2,3,4,5,6,7] # days
   ...: y = [25, 32, 45, 21, 39, 52, 45] # sales
   ...: plt.plot(x,y, 'm') # magneta
   ...: plt.xlabel("Days")
   ...: plt.ylabel("Sales")
   ...: plt.title('Sales per Day')
   ...: plt.show()
```



```
In [7]:
   ...: x = [1,2,3,4,5,6,7] # days
   ...: y = [25, 32, 45, 21, 39, 52, 45] # sales
   ...: plt.plot(x,y, 'gD') # green dotted --, hat ^, filled circle'o',
diamond D
   ...: plt.xlabel("Days")
   ...: plt.ylabel("Sales")
   ...: plt.title('Sales per Day')
   ...: plt.show()
```



Diamond

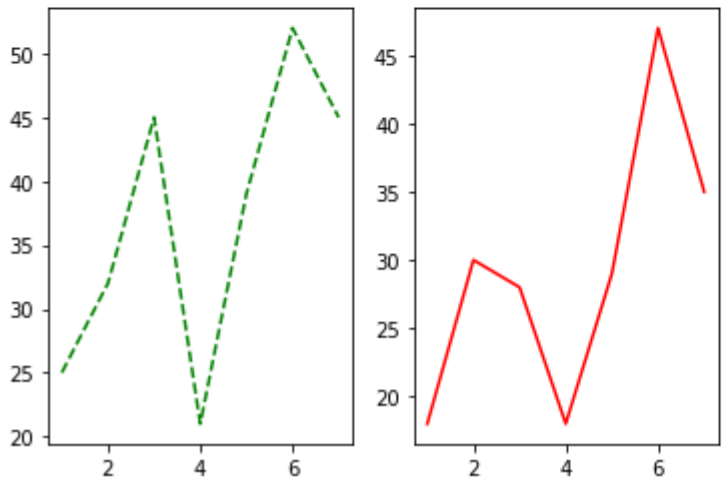
```
29  # line style
30  x = [1,2,3,4,5,6,7] # days
31  y = [25, 32, 45, 21, 39, 52, 45] # sales
32  plt.plot(x,y, 'gD') # green dotted --, hat ^, filled circle'o', diamond D
33  plt.xlabel("Days")
34  plt.ylabel("Sales")
35  plt.title('Sales per Day')
36  plt.show()
```

More Line Charts

```
In [8]:
   ...: x = [1,2,3,4,5,6,7] # days
   ...: y = [25, 32, 45, 21, 39, 52, 45] # sales
   ...: z = [18, 30, 28, 18, 29, 47, 35] # sold on credit
   ...: plt.plot(x,y, 'g--')
   ...: plt.plot(x,z, 'r')
   ...: plt.xlabel("Days")
   ...: plt.ylabel("Sales & Credit")
   ...: plt.title('Sales & Credit per Day')
   ...: plt.legend(labels = ('sales', 'credit sales'), loc = 'upper left')
   ...: plt.show()
```


```
38  #_______multilines
39  x = [1,2,3,4,5,6,7] # days
40  y = [25, 32, 45, 21, 39, 52, 45] # sales
41  z = [18, 30, 28, 18, 29, 47, 35] # sold on credit
42  plt.plot(x,y, 'g--')
43  plt.plot(x,z, 'r')
44  plt.xlabel("Days")
45  plt.ylabel("Sales & Credit")
46  plt.title('Sales & Credit per Day')
47  plt.legend(labels = ('sales', 'credit sales'), loc = 'upper left')
48  plt.show()
```

```
In [9]:
   ...: x = [1,2,3,4,5,6,7] # days
   ...: y = [25, 32, 45, 21, 39, 52, 45] # sales
   ...: z = [18, 30, 28, 18, 29, 47, 35] # sold on credit
   ...: plt.figure(figsize=(6, 4))
   ...: fig, (ax1, ax2)= plt.subplots(1, 2)
   ...: ax1.plot(x,y, 'g--')
   ...: ax2.plot(x,z, 'r')
   ...: plt.show()
<Figure size 432x288 with 0 Axes>
```



Sub Plots

```
50  # sub plots
51  x = [1,2,3,4,5,6,7] # days
52  y = [25, 32, 45, 21, 39, 52, 45] # sales
53  z = [18, 30, 28, 18, 29, 47, 35] # sold on credit
54  plt.figure(figsize=(6, 4))
55  fig, (ax1, ax2)= plt.subplots(1, 2)
56  ax1.plot(x,y, 'g--')
57  ax2.plot(x,z, 'r')
58  plt.show()
59
```

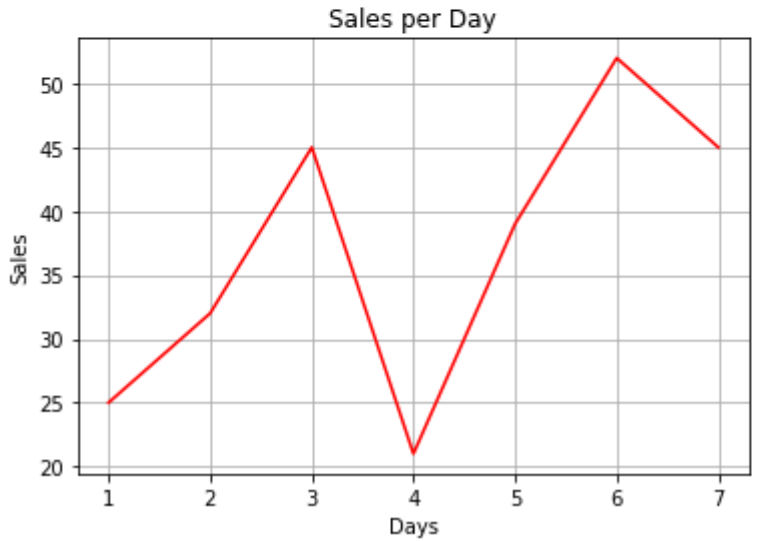
Sub Plots

```
In [10]:
   ...:
   ...: x = [1,2,3,4,5,6,7] # days
   ...: y = [25, 32, 45, 21, 39, 52, 45] # sales
   ...: z = [18, 30, 28, 18, 29, 47, 35] # sold on credit
   ...: plt.figure(figsize=(6, 4))
   ...: fig, (ax1, ax2)= plt.subplots(1, 2)
   ...: ax1.plot(x,y, 'g--')
   ...: ax1.set_title('Sales vs Days')
   ...: ax2.plot(x,z, 'r')
   ...: ax2.set_title('Credit vs Days')
   ...: plt.show()

<Figure size 432x288 with 0 Axes>
```


```
60  #_____subplot with titles
61  x = [1,2,3,4,5,6,7] # days
62  y = [25, 32, 45, 21, 39, 52, 45] # sales
63  z = [18, 30, 28, 18, 29, 47, 35] # sold on credit
64  plt.figure(figsize=(6, 4))
65  fig, (ax1, ax2)= plt.subplots(1, 2)
66  ax1.plot(x,y, 'g--')
67  ax1.set_title('Sales vs Days')
68  ax2.plot(x,z, 'r')
69  ax2.set_title('Credit vs Days')
70  plt.show()
```

```
In [11]:
    ...: x = [1,2,3,4,5,6,7] # days
    ...: y = [25, 32, 45, 21, 39, 52, 45] # sales
    ...: plt.plot(x,y, 'r') # red
    ...: plt.xlabel("Days")
    ...: plt.ylabel("Sales")
    ...: plt.title('Sales per Day')
    ...: plt.grid(True)
    ...: plt.show()
```

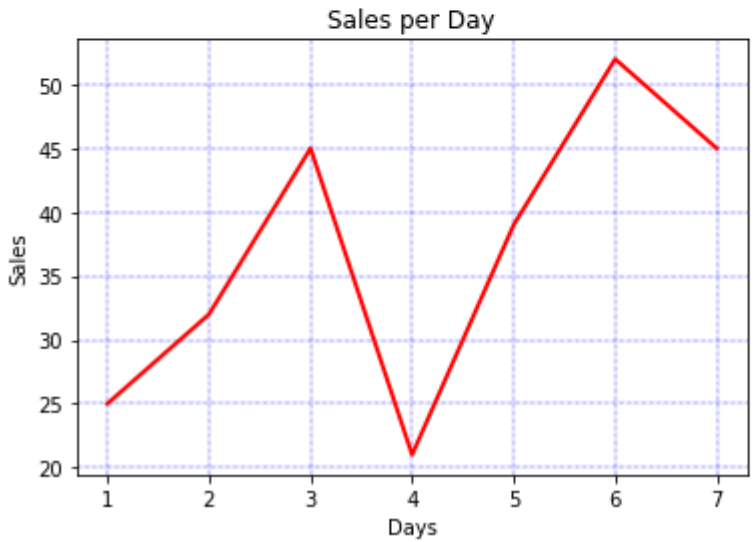
Grid




```
73  #_____plot with default grid
74  x = [1,2,3,4,5,6,7] # days
75  y = [25, 32, 45, 21, 39, 52, 45] # sales
76  plt.plot(x,y, 'r') # red
77  plt.xlabel("Days")
78  plt.ylabel("Sales")
79  plt.title('Sales per Day')
80  plt.grid(True)
81  plt.show()
```

Grid

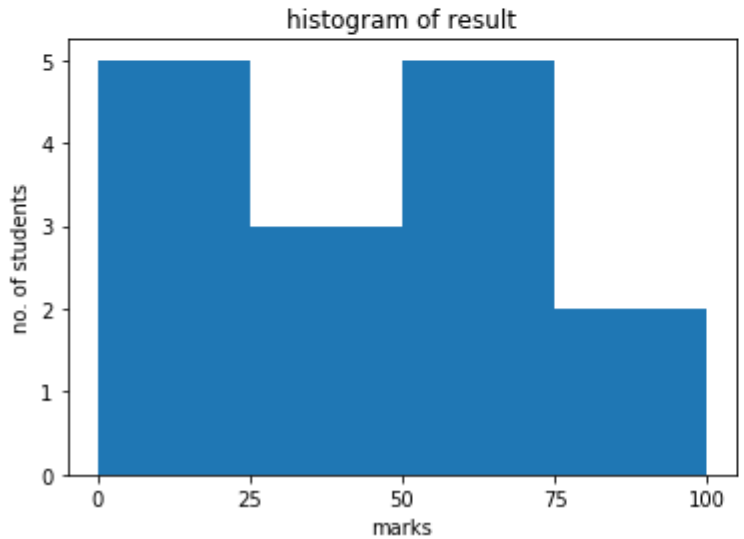
```
In [12]:
    ...: x = [1,2,3,4,5,6,7] # days
    ...: y = [25, 32, 45, 21, 39, 52, 45] # sales
    ...: plt.plot(x,y, 'r', lw = 2) # red
    ...: plt.xlabel("Days")
    ...: plt.ylabel("Sales")
    ...: plt.title('Sales per Day')
    ...: plt.grid(color='b', ls = '-.', lw = 0.35)
    ...: plt.show()
```




```
85  #_____plot with custom grid, line width = 2
86  x = [1,2,3,4,5,6,7] # days
87  y = [25, 32, 45, 21, 39, 52, 45] # sales
88  plt.plot(x,y, 'r', lw = 2) # red
89  plt.xlabel("Days")
90  plt.ylabel("Sales")
91  plt.title('Sales per Day')
92  plt.grid(color='b', ls = '-.', lw = 0.35)
93  plt.show()
94
```

```
In [15]:
   ...: fig,ax = plt.subplots(1,1)
   ...: a = np.array([22,87,5,43,56,73,55,54,11,20,51,5,79,31,27])
   ...: ax.hist(a, bins = [0,25,50,75,100])
   ...: ax.set_title("histogram of result")
   ...: ax.set_xticks([0,25,50,75,100])
   ...: ax.set_xlabel('marks')
   ...: ax.set_ylabel('no. of students')
   ...: plt.show()
```

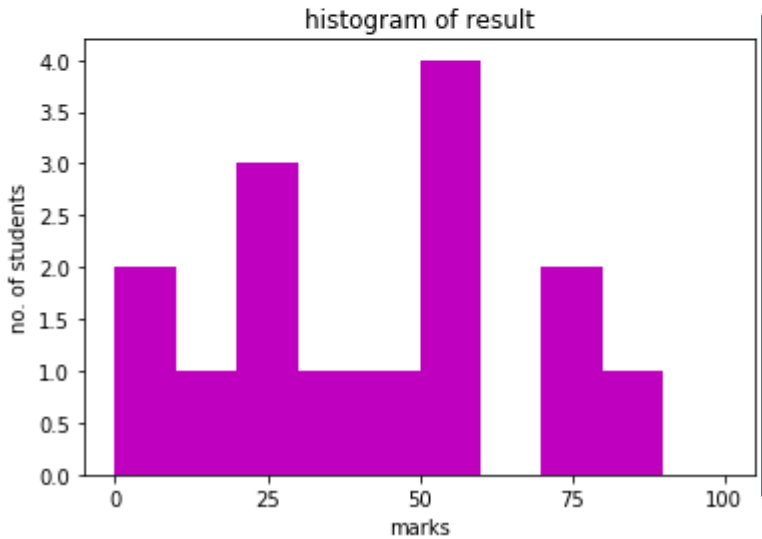
Histogram




```
 97  #_____histogram
 98  fig,ax = plt.subplots(1,1)
 99  a = np.array([22,87,5,43,56,73,55,54,11,20,51,5,79,31,27])
100  ax.hist(a, bins = [0,25,50,75,100])
101  ax.set_title("histogram of result")
102  ax.set_xticks([0,25,50,75,100])
103  ax.set_xlabel('marks')
104  ax.set_ylabel('no. of students')
105  plt.show()
106
```

```
In [16]: fig,ax = plt.subplots(1,1)
    ...: a = np.array([22,87,5,43,56,73,55,54,11,20,51,5,79,31,27])
    ...: ax.hist(a, bins = [0,10,20,30,40,50,60,70,80,90,100], facecolor =
'm')
    ...: ax.set_title("histogram of result")
    ...: ax.set_xticks([0,25,50,75,100])
    ...: ax.set_xlabel('marks')
    ...: ax.set_ylabel('no. of students')
    ...: plt.show()
```

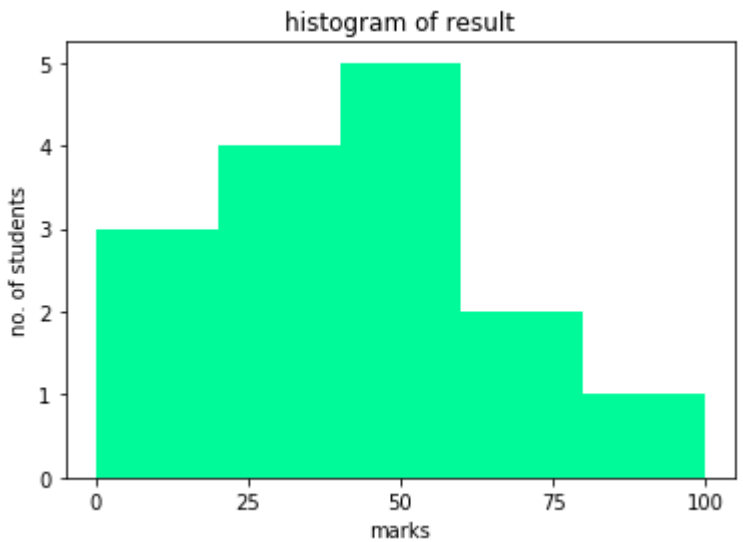
Histogram




```
107  #_____color, bin size 10
108  fig,ax = plt.subplots(1,1)
109  a = np.array([22,87,5,43,56,73,55,54,11,20,51,5,79,31,27])
110  ax.hist(a, bins = [0,10,20,30,40,50,60,70,80,90,100], facecolor = 'm')
111  ax.set_title("histogram of result")
112  ax.set_xticks([0,25,50,75,100])
113  ax.set_xlabel('marks')
114  ax.set_ylabel('no. of students')
115  plt.show()
116
```

```
In [17]:
    ...: fig,ax = plt.subplots(1,1)
    ...: a = np.array([22,87,5,43,56,73,55,54,11,20,51,5,79,31,27])
    ...: ax.hist(a, bins = [0,20,40,60,80,100], facecolor =
'mediumspringgreen')
    ...: ax.set_title("histogram of result")
    ...: ax.set_xticks([0,25,50,75,100])
    ...: ax.set_xlabel('marks')
    ...: ax.set_ylabel('no. of students')
    ...: plt.show()
```

Histogram

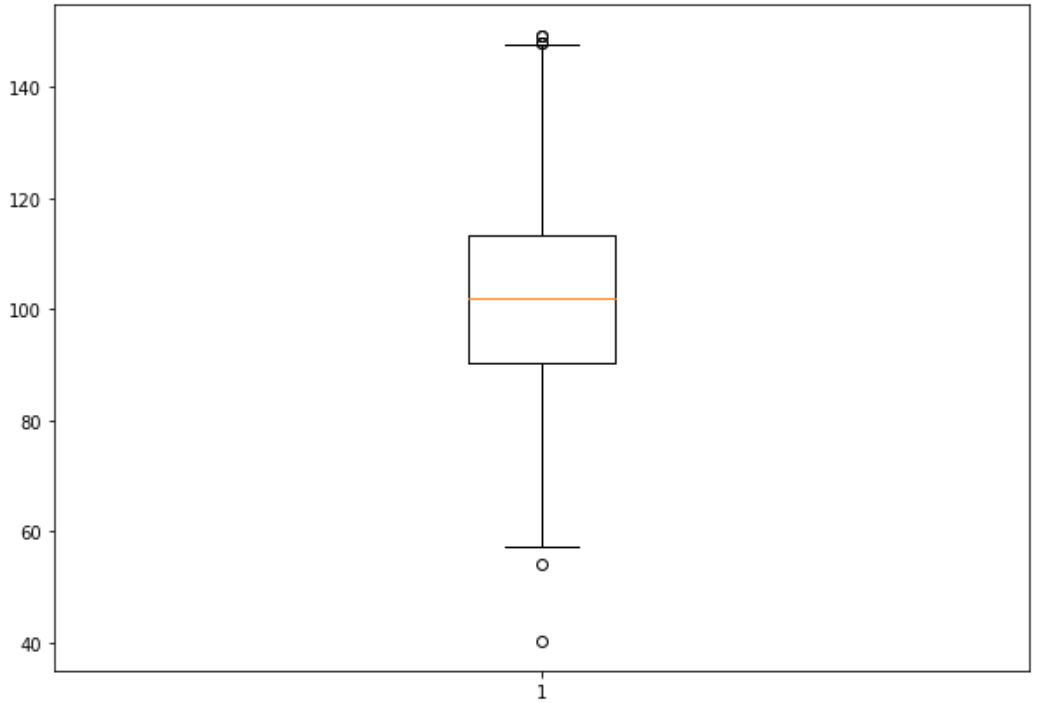



```
118  #_____bin size = 20
119  fig,ax = plt.subplots(1,1)
120  a = np.array([22,87,5,43,56,73,55,54,11,20,51,5,79,31,27])
121  ax.hist(a, bins = [0,20,40,60,80,100], facecolor = 'mediumspringgreen')
122  ax.set_title("histogram of result")
123  ax.set_xticks([0,25,50,75,100])
124  ax.set_xlabel('marks')
125  ax.set_ylabel('no. of students')
126  plt.show()
```

```
In [18]:
   ...: cs2m = pd.read_csv("C:/Users/Dr Vinod/Desktop/DataSets1/cs2m.csv")
   ...: cs2m = pd.DataFrame(cs2m)
   ...:
   ...: fig,ax = plt.subplots(1,1)
   ...: ax.hist(cs2m.Age, bins = [0,20,40,60,80,100], facecolor = 'deeppink')
   ...: ax.set_title("Histogram of Age")
   ...: ax.set_xticks([0,25,50,75,100])
   ...: ax.set_xlabel('Age')
   ...: ax.set_ylabel('Counts ladies')
   ...: plt.show()
```


```
128  #_________hist of Age file cs2m
129  cs2m = pd.read_csv("C:/Users/Dr Vinod/Desktop/DataSets1/cs2m.csv")
130  cs2m = pd.DataFrame(cs2m)
131
132  fig,ax = plt.subplots(1,1)
133  ax.hist(cs2m.Age, bins = [0,20,40,60,80,100], facecolor = 'deeppink')
134  ax.set_title("Histogram of Age")
135  ax.set_xticks([0,25,50,75,100])
136  ax.set_xlabel('Age')
137  ax.set_ylabel('Counts ladies')
138  plt.show()
139
```

```
In [19]:
   ...: import matplotlib.pyplot as plt
   ...: import numpy as np
   ...: # Creating dataset
   ...: np.random.seed(10)
   ...: data = np.random.normal(100, 20, 200)
   ...: fig = plt.figure(figsize =(10, 7))
   ...: # Creating plot
   ...: plt.boxplot(data)
   ...: # show plot
   ...: plt.show()
```

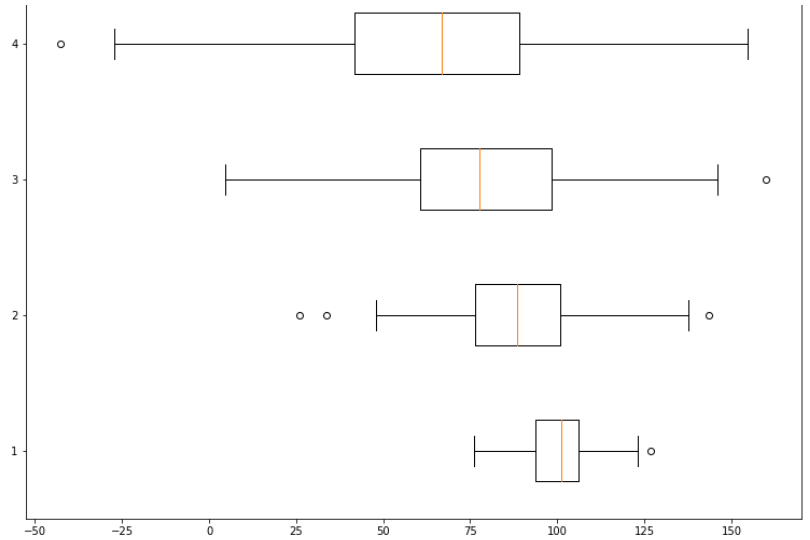


Box Plot

```
140  #_________Box Plots
141
142  #_______one boxplot
143  import matplotlib.pyplot as plt
144  import numpy as np
145  # Creating dataset
146  np.random.seed(10)
147  data = np.random.normal(100, 20, 200)
148  fig = plt.figure(figsize =(10, 7))
149  # Creating plot
150  plt.boxplot(data)
151  # show plot
152  plt.show()
```

More Box Plots

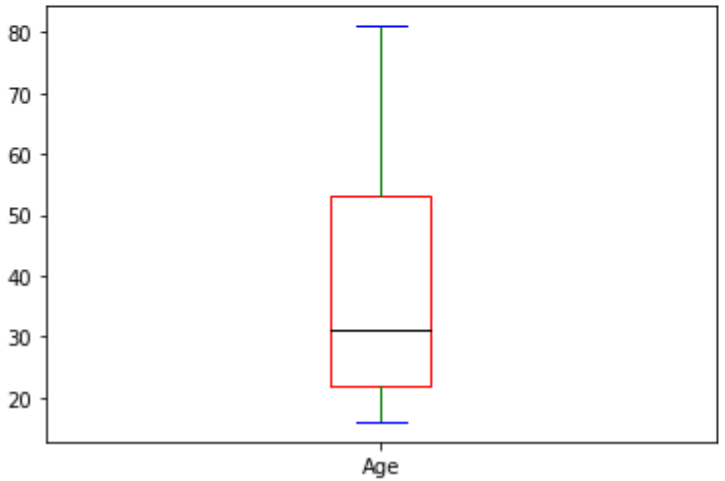
```
In [20]:
    ...: data_1 = np.random.normal(100, 10, 200)
    ...: data_2 = np.random.normal(90, 20, 200)
    ...: data_3 = np.random.normal(80, 30, 200)
    ...: data_4 = np.random.normal(70, 40, 200)
    ...: data = [data_1, data_2, data_3, data_4]
    ...: fig = plt.figure(figsize =(10, 7))
    ...: # Creating axes instance
    ...: ax = fig.add_axes([0, 0, 1, 1])
    ...: # Creating plot
    ...: bp = ax.boxplot(data, vert = False )
    ...: # show plot
    ...: plt.show()
```




```
154  #_____many boxplots
155  data_1 = np.random.normal(100, 10, 200)
156  data_2 = np.random.normal(90, 20, 200)
157  data_3 = np.random.normal(80, 30, 200)
158  data_4 = np.random.normal(70, 40, 200)
159  data = [data_1, data_2, data_3, data_4]
160  fig = plt.figure(figsize =(10, 7))
161  # Creating axes instance
162  ax = fig.add_axes([0, 0, 1, 1])
163  # Creating plot
164  bp = ax.boxplot(data, vert = False )
165  # show plot
166  plt.show()
```

```
168  #_____Age, cs2m
169  Age = cs2m['Age']
170
171  #__making colorful
172  props2 = {'boxes': 'red', 'whiskers': 'green', 'medians': 'black', 'caps': 'blue'}
173  Age.plot.box(color=props2)
```

```
In [21]: Age = cs2m['Age']

In [22]: #__making colorful

In [23]: props2 = {'boxes': 'red', 'whiskers': 'green', 'medians': 'black',
'caps': 'blue'}

In [24]: Age.plot.box(color=props2)
Out[24]: <matplotlib.axes._subplots.AxesSubplot at 0x19630890e08>
```

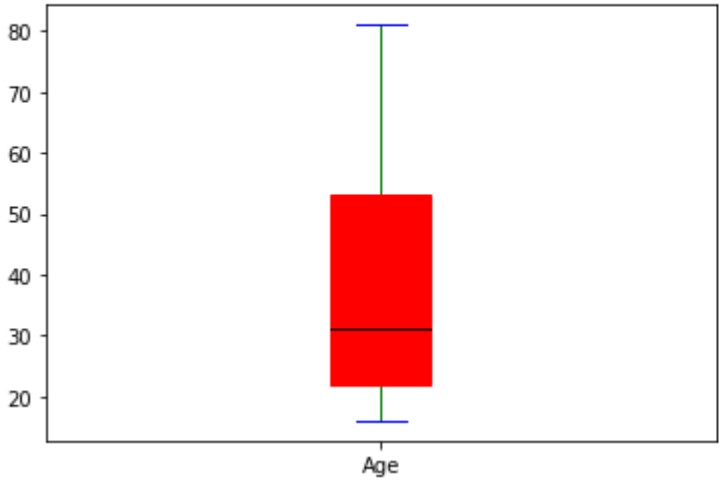


Box Plot

Box Plot

```
174 Age.plot.box(color=props2, patch_artist = True, vert = True)
175
```

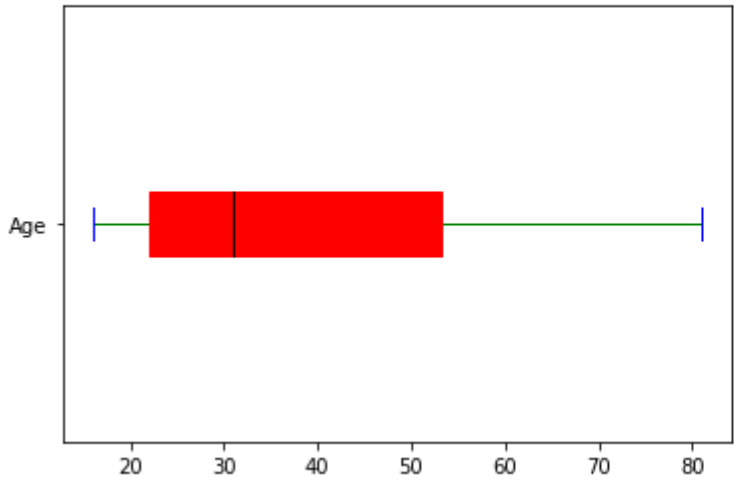
```
In [25]: Age.plot.box(color=props2, patch_artist = True, vert = True)
Out[25]: <matplotlib.axes._subplots.AxesSubplot at 0x1963080a348>
```



Box Plot

```
176  Age.plot.box(color=props2, patch_artist = True, vert = False)
177
```

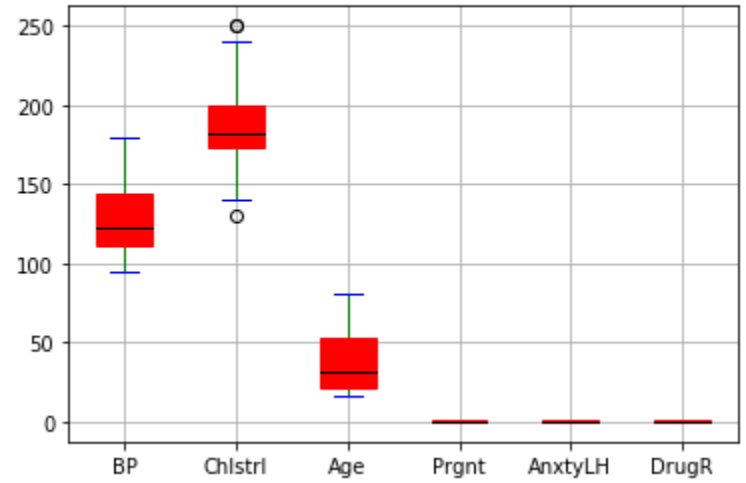
```
In [26]: Age.plot.box(color=props2, patch_artist = True, vert = False)
Out[26]: <matplotlib.axes._subplots.AxesSubplot at 0x19630703bc8>
```



Box Plot

```
178  cs2m.boxplot(color = props2, patch_artist = True)
179
```

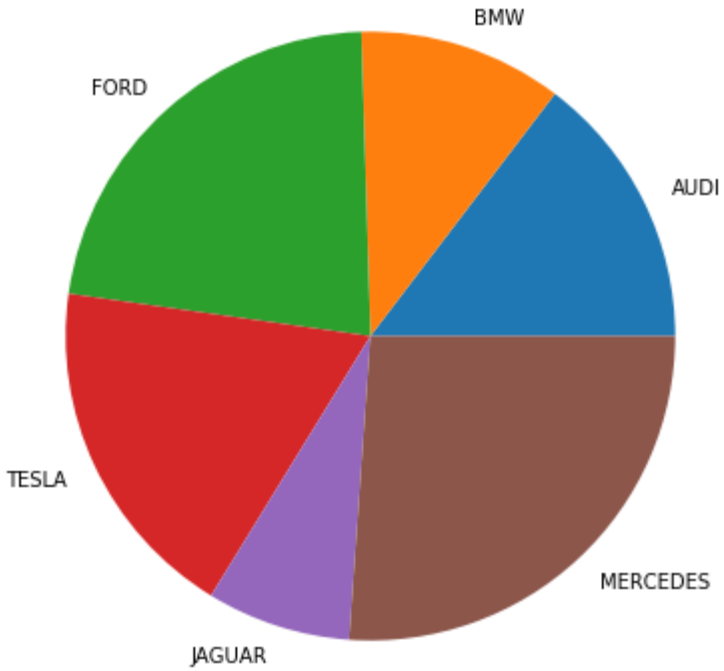
```
In [27]: cs2m.boxplot(color = props2, patch_artist = True)
Out[27]: <matplotlib.axes._subplots.AxesSubplot at 0x196308b8fc8>
```



Pie Chart

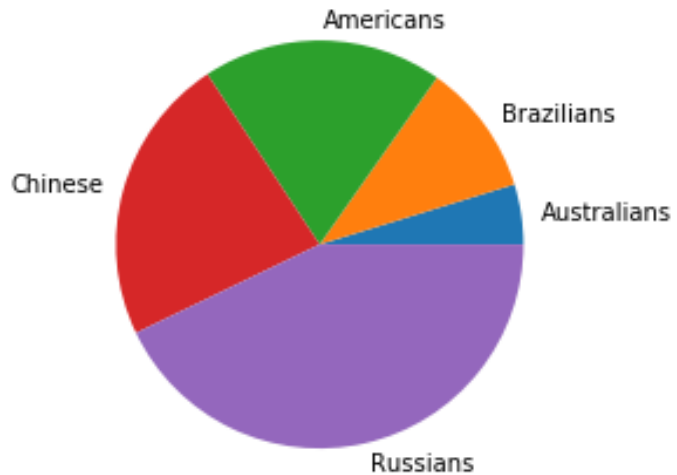
```
242  #_______pie chart
243
244  # Creating dataset
245  cars = ['AUDI', 'BMW', 'FORD',
246          'TESLA', 'JAGUAR', 'MERCEDES']
247
248  data = [23, 17, 35, 29, 12, 41]
249  # Creating plot
250  fig = plt.figure(figsize =(10, 7))
251  plt.pie(data, labels = cars)
252  # show plot
253  plt.show()
```

```
In [35]: cars = ['AUDI', 'BMW', 'FORD',
    ...:          'TESLA', 'JAGUAR', 'MERCEDES']

In [36]: data = [23, 17, 35, 29, 12, 41]
    ...: # Creating plot
    ...: fig = plt.figure(figsize =(10, 7))
    ...: plt.pie(data, labels = cars)
    ...: # show plot
    ...: plt.show()
```



```
255  #_____________pie from df grades
256  grades = pd.read_csv("C:/Users/Dr Vinod/Desktop/DataSets1/grades.csv")
257  grades = pd.DataFrame(grades)
258  grades.ethnicity.value_counts()
259  ethnicity = ['Australians', 'Brazilians', 'Americans', 'Chinese', 'Russians']
260  data = [5,11,20,24,45]
261  plt.pie(data, labels = ethnicity)
```

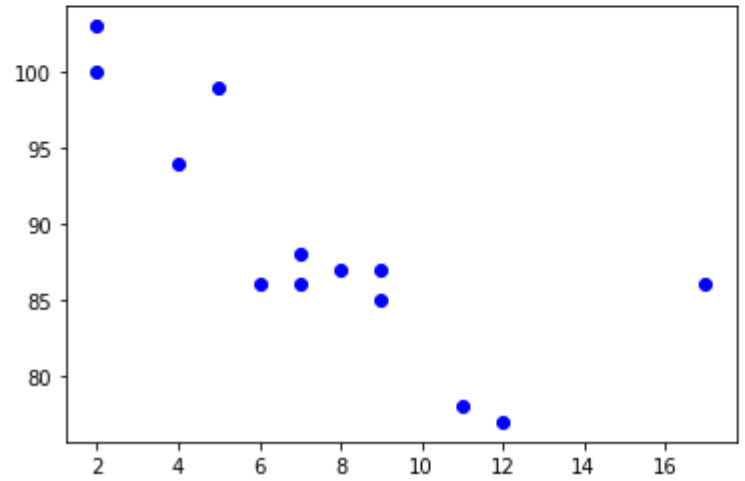
```
In [37]: #_____________pie from df grades

In [38]: grades = pd.read_csv("C:/Users/Dr Vinod/Desktop/DataSets1/grades.csv")

In [39]: grades = pd.DataFrame(grades)

In [40]: grades.ethnicity.value_counts()
Out[40]:
4    45
3    24
2    20
5    11
1     5
Name: ethnicity, dtype: int64

In [41]: ethnicity = ['Australians', 'Brazilians', 'Americans', 'Chinese', 'Russians']

In [42]: data = [5,11,20,24,45]

In [43]: plt.pie(data, labels = ethnicity)
```



Scatter Plot

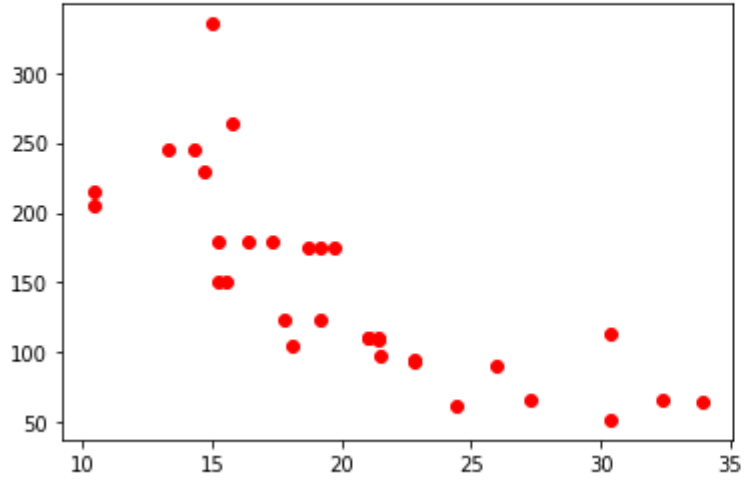
```
263  #_____________scatter plot
264  x =[5, 7, 8, 7, 2, 17, 2, 9,
265      4, 11, 12, 9, 6]
266  y =[99, 86, 87, 88, 100, 86,
267      103, 87, 94, 78, 77, 85, 86]
268  plt.scatter(x, y, c ="blue")
269  # To show the plot
270  plt.show()
271
```

```
In [44]: #_____________scatter plot

In [45]: x =[5, 7, 8, 7, 2, 17, 2, 9,
    ...:     4, 11, 12, 9, 6]

In [46]: y =[99, 86, 87, 88, 100, 86,
    ...:     103, 87, 94, 78, 77, 85, 86]

In [47]: plt.scatter(x, y, c ="blue")
    ...: # To show the plot
    ...: plt.show()
```



```
In [48]: mtcars = pd.read_csv("C:/Users/Dr Vinod/Desktop/DataSets1/
mtcars.csv")

In [49]: mtcars = pd.DataFrame(mtcars)

In [50]: mtcars.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32 entries, 0 to 31
Data columns (total 12 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Unnamed: 0  32 non-null     object
 1   mpg         32 non-null     float64
 2   cyl         32 non-null     int64
 3   disp        32 non-null     float64
 4   hp          32 non-null     int64
 5   drat        32 non-null     float64
 6   wt          32 non-null     float64
 7   qsec        32 non-null     float64
 8   vs          32 non-null     int64
 9   am          32 non-null     int64
 10  gear        32 non-null     int64
 11  carb        32 non-null     int64
dtypes: float64(5), int64(6), object(1)
memory usage: 3.1+ KB
```

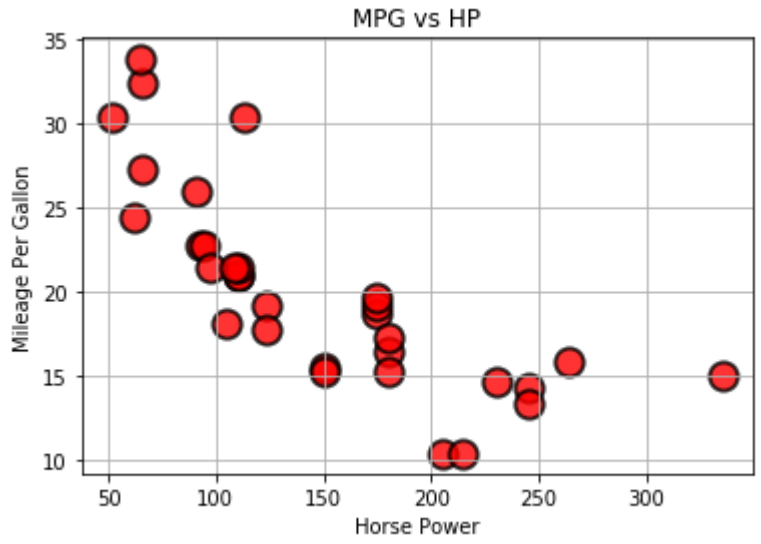
```
In [51]: plt.scatter(mtcars.mpg, mtcars.hp, c = 'r')
Out[51]: <matplotlib.collections.PathCollection at 0x19631a9e588>
```




```
273  mtcars = pd.read_csv("C:/Users/Dr Vinod/Desktop/DataSets1/mtcars.csv")
274  mtcars = pd.DataFrame(mtcars)
275  mtcars.info()
276  plt.scatter(mtcars.mpg, mtcars.hp, c = 'r')
277
```

Scatter Plot

```
In [52]: plt.scatter(mtcars.hp, mtcars.mpg, c ="red",
    ...:                linewidths = 2,
    ...:                marker ="o",
    ...:                edgecolor ="k",
    ...:                s = 200,
    ...:                alpha = 0.8)
    ...:
    ...: plt.xlabel("Horse Power")
    ...: plt.ylabel("Mileage Per Gallon")
    ...: plt.title('MPG vs HP')
    ...: plt.grid()
    ...: plt.show()
```




```
279 """_________linewidths, color, marker, edge color,
280 s = size of marker, alpha = low value, less visibility """
281
282 plt.scatter(mtcars.hp, mtcars.mpg, c ="red",
283                linewidths = 2,
284                marker ="o",
285                edgecolor ="k",
286                s = 200,
287                alpha = 0.8)
288
289 plt.xlabel("Horse Power")
290 plt.ylabel("Mileage Per Gallon")
291 plt.title('MPG vs HP')
292 plt.grid()
293 plt.show()
294
```

Scatter Plot

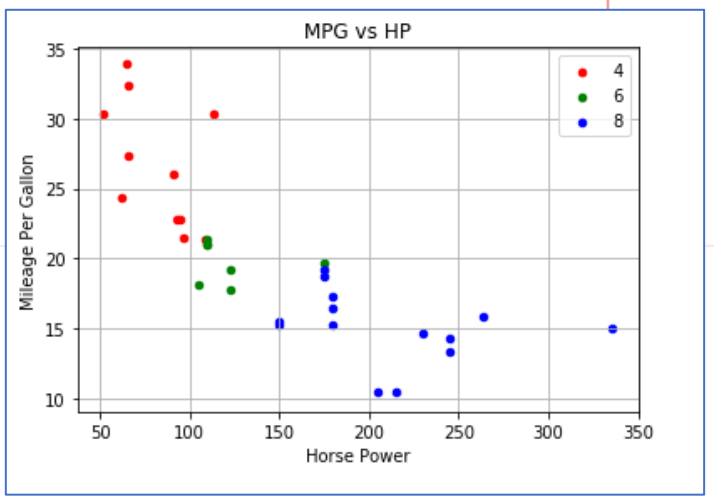
```
296  #_____---scatter plot by category
297  mtcars.info()
298  mtcars.cyl.value_counts() #4=11, 6=7, 8=14
299
```

```
In [62]: mtcars.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32 entries, 0 to 31
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   Unnamed: 0   32 non-null     object
 1   mpg          32 non-null     float64
 2   cyl          32 non-null     int64
 3   disp         32 non-null     float64
 4   hp           32 non-null     int64
 5   drat         32 non-null     float64
 6   wt           32 non-null     float64
 7   qsec         32 non-null     float64
 8   vs           32 non-null     int64
 9   am           32 non-null     int64
 10  gear         32 non-null     int64
 11  carb         32 non-null     int64
dtypes: float64(5), int64(6), object(1)
memory usage: 3.1+ KB
```

```
In [63]: mtcars.cyl.value_counts() #4=11, 6=7, 8=14
Out[63]:
8    14
4    11
6     7
Name: cyl, dtype: int64
```

Scatter Plot by Category

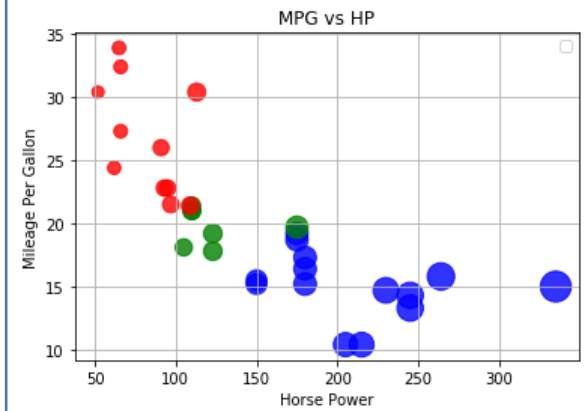
```
296  #_____---scatter plot by category
297  mtcars.info()
298  mtcars.cyl.value_counts() #4=11, 6=7, 8=14
299
300  #___________simple category wise
301  fig, ax = plt.subplots()
302  colors = {4:'red', 6:'green', 8:'blue'}
303  grouped = mtcars.groupby('cyl')
304
305  for key, group in grouped:
306      group.plot(ax=ax, kind='scatter', x= 'hp', y= 'mpg', label=key, color=colors[key])
307  plt.xlabel("Horse Power")
308  plt.ylabel("Mileage Per Gallon")
309  plt.title('MPG vs HP')
310  plt.grid()
311  plt.show()
312
```




```
In [64]:
    ...: fig, ax = plt.subplots()
    ...: colors = {4:'red', 6:'green', 8:'blue'}
    ...: grouped = mtcars.groupby('cyl')
    ...:
    ...: for key, group in grouped:
    ...:     group.plot(ax=ax, kind='scatter', x= 'hp', y= 'mpg', label=key, color=colors[key])
    ...: plt.xlabel("Horse Power")
    ...: plt.ylabel("Mileage Per Gallon")
    ...: plt.title('MPG vs HP')
    ...: plt.grid()
    ...: plt.show()
```

```
315  #_____________________________________beautiful
316  x = mtcars['hp']
317  y = mtcars['mpg']
318  colors = {4:'red', 6:'green', 8:'blue'}
319
320  plt.scatter(x, y, s=1.25*mtcars['hp'], alpha=0.8,
321              c= mtcars['cyl'].map(colors)
322              )
323  plt.xlabel("Horse Power")
324  plt.ylabel("Mileage Per Gallon")
325  plt.title('MPG vs HP')
326  plt.grid()
327  labels = {4:'red', 6:'green', 8:'blue'}
328  plt.legend(loc = 'upper right')
329
```

```
Out[65]: <matplotlib.legend.Legend at 0x1962ff67308>
```




```
In [65]:
    ...: x = mtcars['hp']
    ...: y = mtcars['mpg']
    ...: colors = {4:'red', 6:'green', 8:'blue'}
    ...:
    ...: plt.scatter(x, y, s=1.25*mtcars['hp'], alpha=0.8,
    ...:             c= mtcars['cyl'].map(colors)
    ...:             )
    ...: plt.xlabel("Horse Power")
    ...: plt.ylabel("Mileage Per Gallon")
    ...: plt.title('MPG vs HP')
    ...: plt.grid()
    ...: labels = {4:'red', 6:'green', 8:'blue'}
    ...: plt.legend(loc = 'upper right')
No handles with labels found to put in legend.
Out[65]: <matplotlib.legend.Legend at 0x1962ff67308>
```


DO
WHAT
YOU
LOVE
LOVE
WHAT
YOU
DO


Enjoy
Learning