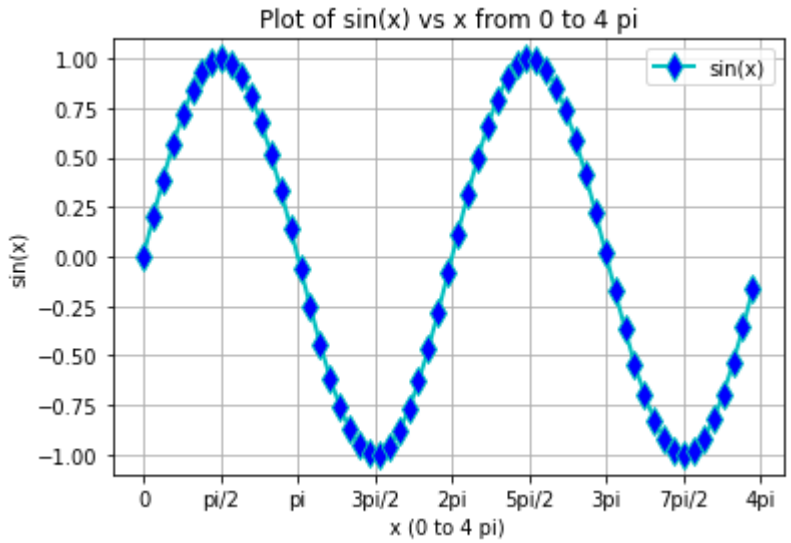
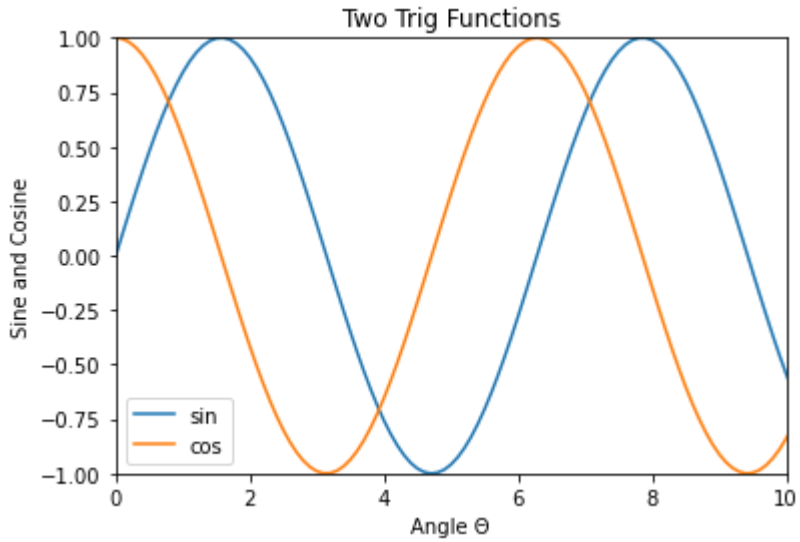
```
import numpy as np
import matplotlib.pyplot as plt

x = np.arange(0, 4 * np.pi, 0.2)
y = np.sin(x)


# 3. Plot data including options
plt.plot(x, y,linewidth=2,linestyle='-',color='c',marker='d',markersize=10,markerfacecolor=


# 4. Add plot details
plt.title('Plot of sin(x) vs x from 0 to 4 pi')
plt.xlabel('x (0 to 4 pi)')
plt.ylabel('sin(x)')

plt.legend(['sin(x)']) # list containing one string
plt.xticks(np.arange(0, 4*np.pi + np.pi/2,np.pi/2 ), ['0','pi/2','pi','3pi/2','2pi','5pi/2'
plt.grid(True)


# 5. Show the plot
plt.show()
```



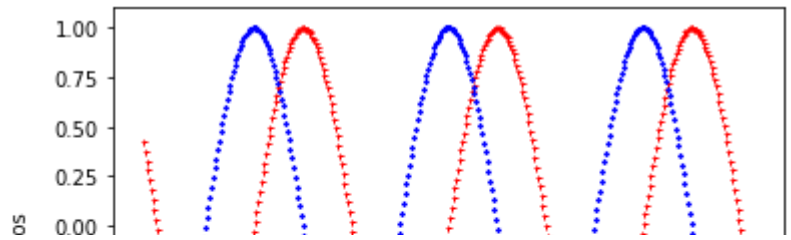

```
#multi-line plot using oo interface
x = np.arange(0,4*np.pi,0.1)
y = np.sin(x)
z = np.cos(x)


fig, ax = plt.subplots()  #create fig,ax object
```

```
ax.plot(x,y)
ax.plot(x,z)# plot


ax.set_title('Two Trig Functions')
ax.legend(['sin','cos'])
ax.xaxis.set_label_text('Angle θ')
ax.yaxis.set_label_text('Sine and Cosine')

ax.set_xlim(0, 10)
ax.set_ylim(-1, 1)

plt.savefig('plot.png', dpi=300) # save figure
plt.show()
```




```
#multi-line using plt
x1=np.arange(-3.14*3.14,3.14*3.14,.05)
y2=np.sin(x1)     # 1 waveform on x1

y1=np.cos(x1)     # 2nd waveform on x2

plt.plot(x1,y2,'r+',ms='3')
plt.xlabel("Sine")
plt.plot(x1,y1,'b*',ms='2')
plt.ylabel("cos")
```

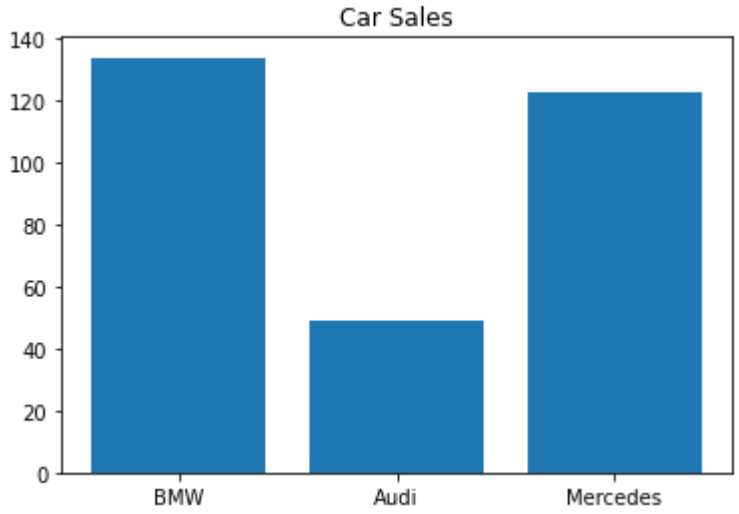
```
Text(0, 0.5, 'cos')
```



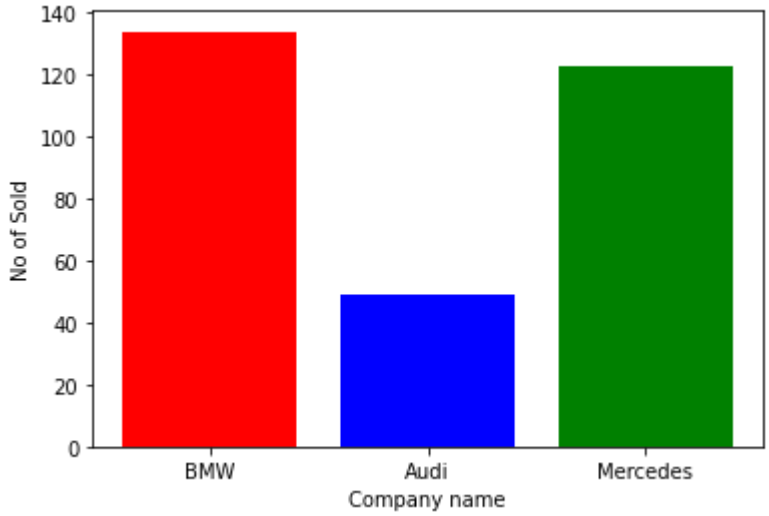

```
import pandas as pd
df=pd.DataFrame(data=[134,49,123],index=['BMW','Audi','Mercedes'],columns=['Number of units
df
```

	Number of units sold
BMW	134
Audi	49
Mercedes	123

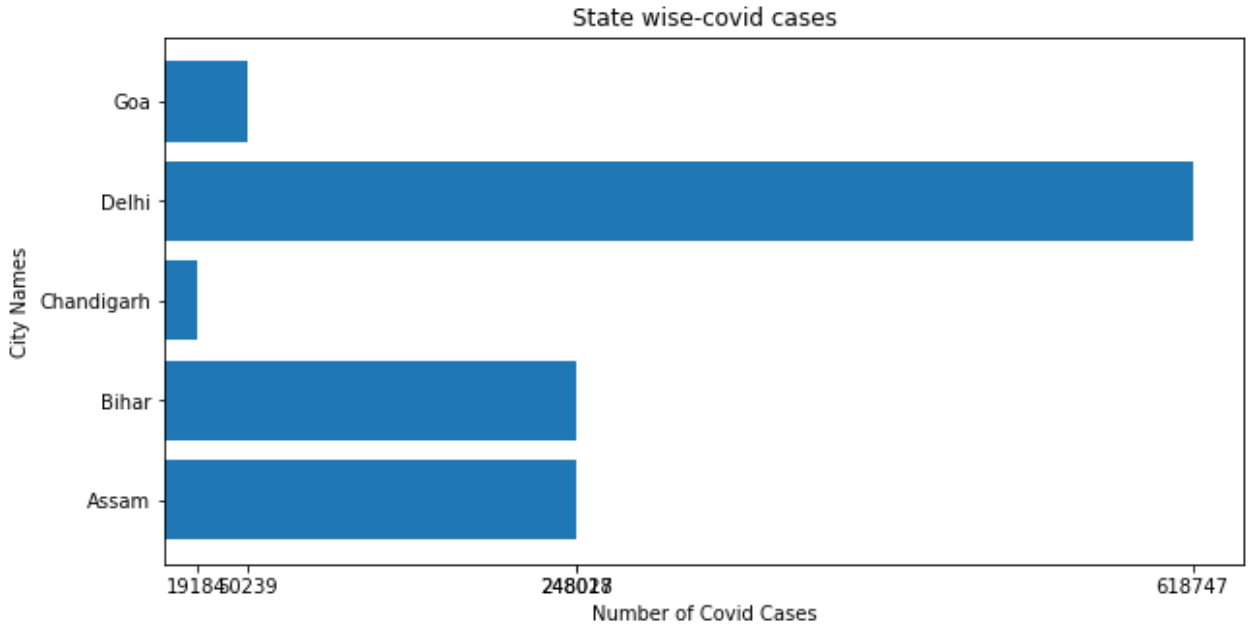
```
import matplotlib.pyplot as plt
plt.bar(df.index,df['Number of units sold'])
plt.title('Car Sales')
plt.show()
```




```
#fig=plt.figure()      #  with o-o interface   fig,ax=plt.subplots()

#axes=fig.add_axes([0,0,1,1])

plt.bar(df.index,df['Number of units sold'],color=['r','b','g'])
plt.xlabel("Company name")
plt.ylabel("No of Sold")
plt.show()
```




```
#Horizontal bar plot

states=['Assam','Bihar','Chandigarh','Delhi','Goa']
cases=[248028,248017,19184,618747,50239]

plt.figure(figsize=(10,5))

plt.barh(states,cases)  #horizontal bar plot

plt.xlabel('Number of Covid Cases')
plt.ylabel('City Names')
plt.title('State wise-covid cases')
plt.xticks(cases)
plt.show()
```

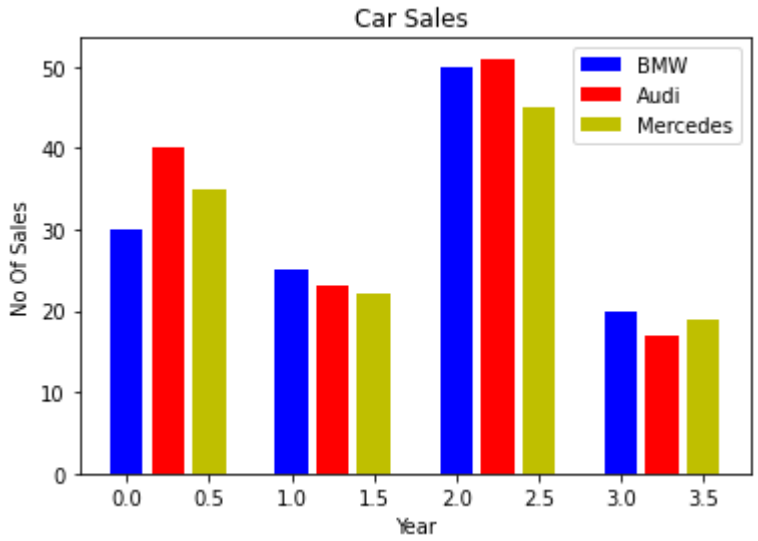



```
df_car=pd.DataFrame([[30,25,50,20],[40,23,51,17],[35,22,45,19]],index=['BMW','Audi','Mercede
df_car
```

	2001	2005	2010	2020
BMW	30	25	50	20
Audi	40	23	51	17
Mercedes	35	22	45	19

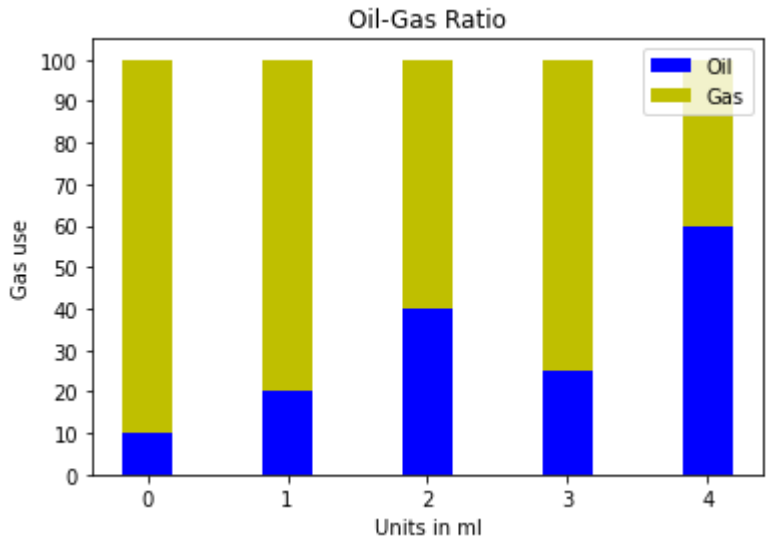
```
#Grouped bar chart
#fig=plt.figure()
#ax=fig.add_axes([0,0,1,1])    #axes object

x=np.arange(4)
print(x)
plt.bar(x+0.00,np.array(df_car.iloc[0:1]).ravel(),color='b', width=0.20)

plt.bar(x+0.25,np.array(df_car.iloc[1:2]).ravel(),color='r', width=0.20)

plt.bar(x+0.50,np.array(df_car.iloc[2:]).ravel(),color='y', width=0.20)

plt.title('Car Sales ')
plt.xlabel('Year')
plt.ylabel('No Of Sales')

plt.legend(labels=['BMW','Audi','Mercedes'])
plt.show()
```

```
[0 1 2 3]
```

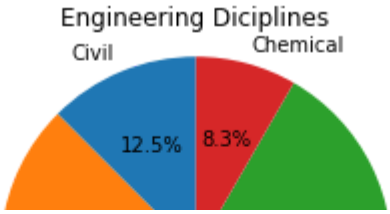



```
#Stacked bar chart
x=np.arange(5)

oilration=(10,20,40,25,60)
gasration=(90,80,60,75,40)

#fig=plt.figure()
```

```
#fig=plt.figure()

#axes=fig.add_axes([0,0,1,1])    # o-o interface
plt.title('Oil-Gas Ratio')
plt.xlabel('Units in ml')
plt.ylabel('Gas use')
plt.yticks(np.arange(0,101,10))

plt.bar(x,oilration,width=0.35,color='b')
plt.bar(x,gasration,width=0.35,color='y',bottom=oilration)

plt.legend(['Oil','Gas'])
plt.show()
```

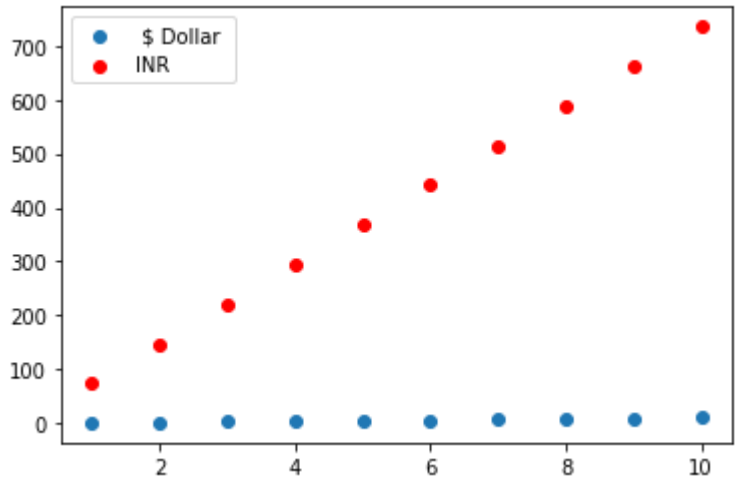



```
labels = ['Civil', 'Electrical', 'Mechanical', 'Chemical']
sizes = [15, 50, 45, 10]


#fig, ax = plt.subplots()
explode = (0, 0, 0, 0)

plt.pie(sizes, labels=labels, autopct='%1.1f%%',explode=explode,startangle=90)
plt.axis('equal')  # Equal aspect ratio ensures the pie chart is circular.
plt.title('Engineering Diciplines')


plt.show()
```

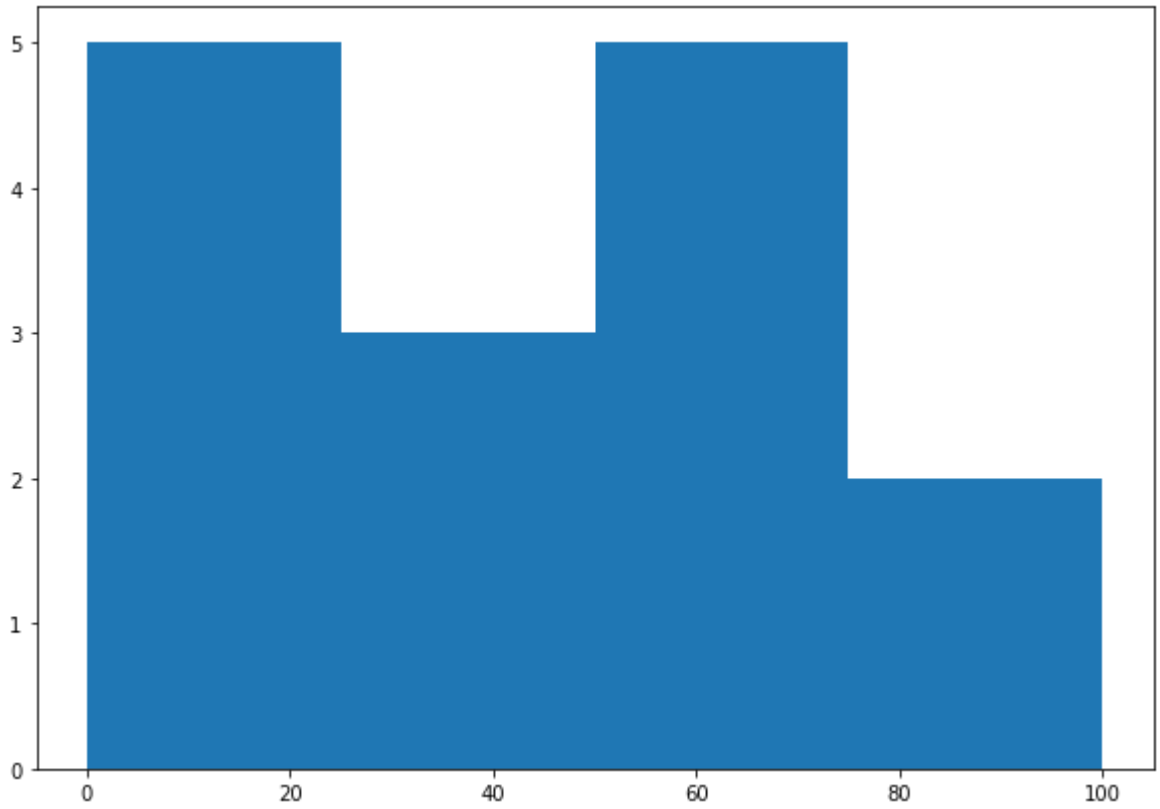



```
#scatter plot
import pandas as pd
df_curr=pd.DataFrame(np.arange(1,11),columns=['Price in $'])
df_curr['Price in INR']=df_curr['Price in $'] * 73.55
df_curr
plt.scatter(np.arange(1,11),df_curr['Price in $'],label=' $ Dollar ')
plt.scatter(np.arange(1,11),df_curr['Price in INR'],color='r',label='INR')
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7ff0a662bdd0>
```




```
a = np.array([22, 87, 5, 43, 56,
              73, 55, 54, 11,
              20, 51, 5, 79, 31,
              27])

# Creating histogram
fig, ax = plt.subplots(figsize =(10, 7))
ax.hist(a, bins = [0, 25, 50, 75, 100])

# Show plot
plt.show()
```


5
4
3
2
1
0
0
20
40
60
80
100