```
import pandas as pd
import numpy as np
```

```
df=pd.DataFrame(np.arange(0,20).reshape(4,5),index=['row1','row2','row3','row4'],       colu
```

```
print(df)
```

```
      column1  column2  column3  column4  column5
row1        0        1        2        3        4
row2        5        6        7        8        9
row3       10       11       12       13       14
row4       15       16       17       18       19
```

```
df.head()
```

|      | column1 | column2 | column3 | column4 | column5 |
|------|---------|---------|---------|---------|---------|
| **row1** | 0 | 1 | 2 | 3 | 4 |
| **row2** | 5 | 6 | 7 | 8 | 9 |
| **row3** | 10 | 11 | 12 | 13 | 14 |
| **row4** | 15 | 16 | 17 | 18 | 19 |

```
df.tail()
```

|      | column1 | column2 | column3 | column4 | column5 |
|------|---------|---------|---------|---------|---------|
| **row1** | 0 | 1 | 2 | 3 | 4 |
| **row2** | 5 | 6 | 7 | 8 | 9 |
| **row3** | 10 | 11 | 12 | 13 | 14 |
| **row4** | 15 | 16 | 17 | 18 | 19 |

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 4 entries, row1 to row4
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   column1  4 non-null      int64
 1   column2  4 non-null      int64
 2   column3  4 non-null      int64
 3   column4  4 non-null      int64
 4   column5  4 non-null      int64
dtypes: int64(5)
memory usage: 364.0+ bytes
```

```
df.iloc[:,:]  # index location of rows    columns
```

```
df.iloc[:,:]  # index location of rows , columns
```

|  | column1 | column2 | column3 | column4 | column5 |
| --- | --- | --- | --- | --- | --- |
| **row1** | 0 | 1 | 2 | 3 | 4 |
| **row2** | 5 | 6 | 7 | 8 | 9 |
| **row3** | 10 | 11 | 12 | 13 | 14 |
| **row4** | 15 | 16 | 17 | 18 | 19 |

```
df.iloc[0:2,1:3]
```

|  | column2 | column3 |
| --- | --- | --- |
| **row1** | 1 | 2 |
| **row2** | 6 | 7 |

```
df.to_csv('pc1.csv')
```

```
import matplotlib.pyplot as plt
%matplotlib inline
# by use of this line we dont have to write plt.show while showing the plot
```

```
a=np.arange(0,20)
b=np.arange(30,50)
```

```
print(a)
```

```
    [ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19]
```

```
print(b)
```

```
    [30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49]
```

```
plt.scatter(a,b,c='r')
#c means color
plt.xlabel('age')
plt.ylabel('height')
plt.title("useless age height ratio")
# to save the graph
plt.savefig("deletefig1.png")
```
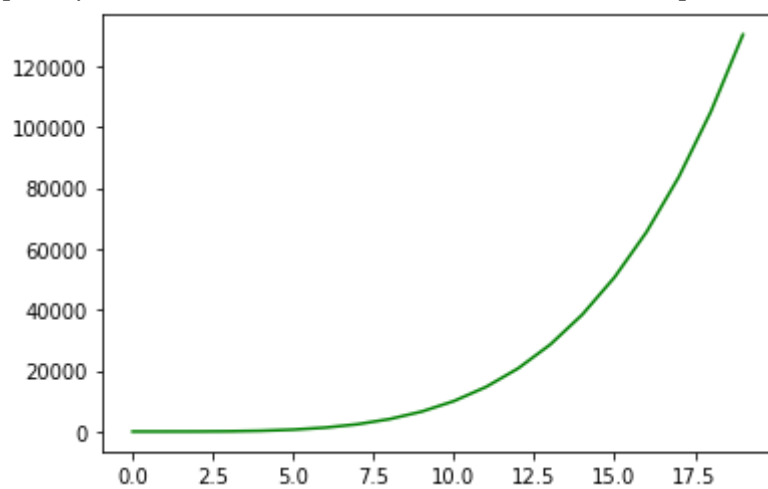
```
b=a**4
```
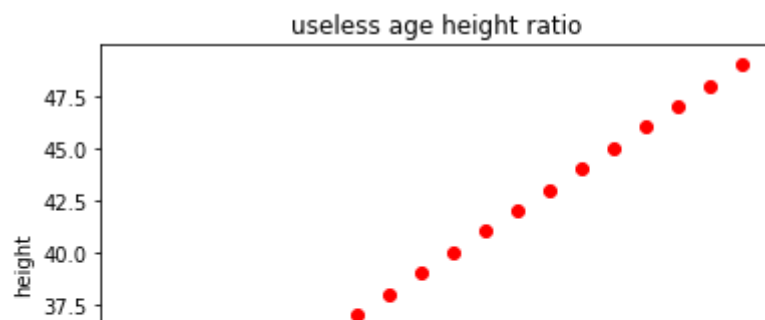
```
plt.plot(a,b,c='g')
```

[<matplotlib.lines.Line2D at 0x7f5d19668750>]



```
plt.plot(a,b,'ro')
```

[<matplotlib.lines.Line2D at 0x7f5d198365d0>]
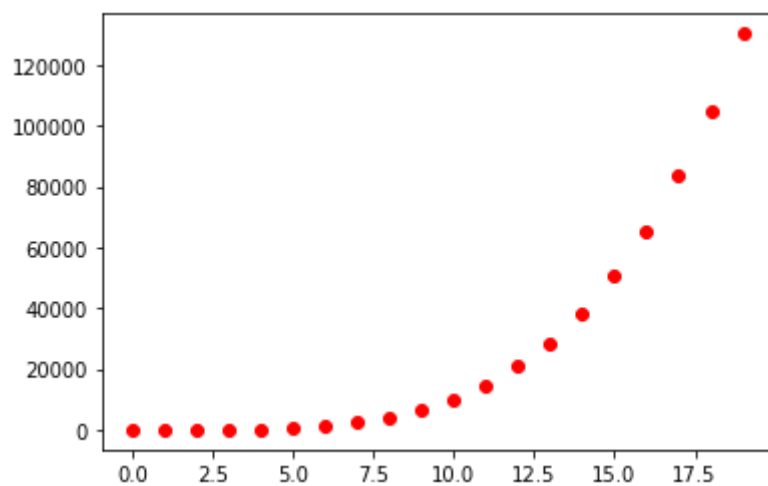


```
plt.plot(a,b,'ro-')
```
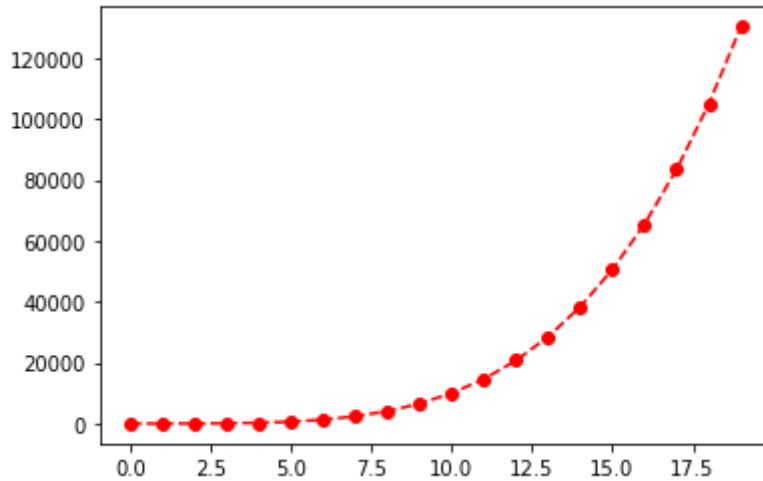
```
[<matplotlib.lines.Line2D at 0x7f5d19918bd0>]
```



```
plt.plot(a,b,'ro--')
```
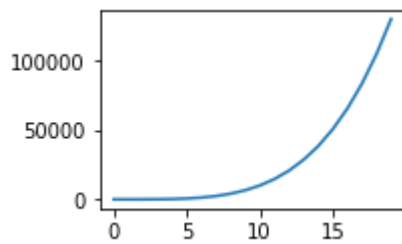
```
[<matplotlib.lines.Line2D at 0x7f5d19513b50>]
```
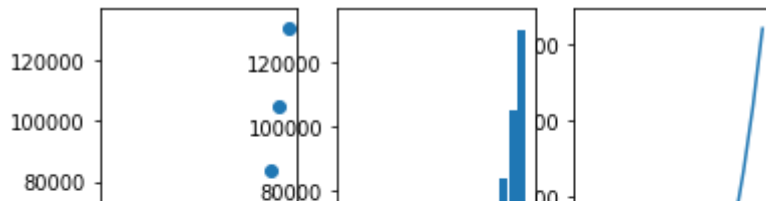


```
plt.subplot(2,2,1) #no of rows,no of columns, position
plt.plot(a,b)
```
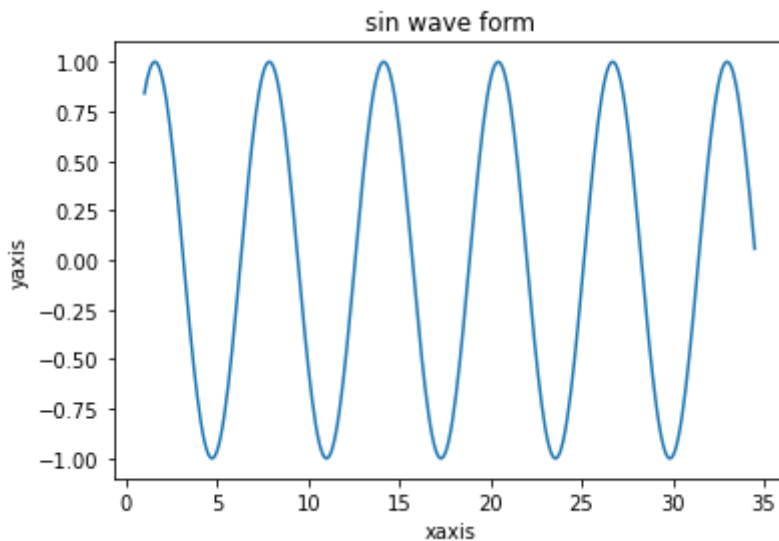
```
[<matplotlib.lines.Line2D at 0x7f5d193fc090>]
```



```
plt.subplot(1,3,1)
plt.scatter(a,b)
plt.subplot(1,3,3)
plt.plot(a,2*b)
plt.subplot(1,3,2)
plt.bar(a,b)
```
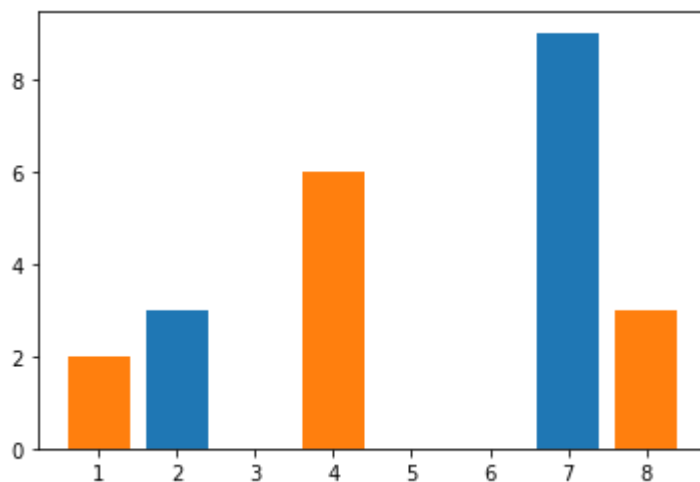
```
<BarContainer object of 20 artists>
```



```
x=np.arange(1,11*np.pi,0.1)
y=np.sin(x)
plt.plot(x,y)
plt.xlabel("xaxis")
plt.ylabel("yaxis")
plt.title("sin wave form")
```

```
Text(0.5, 1.0, 'sin wave form')
```



```
x=[2,4,7]
y=[3,6,9]
x1=[1,4,8]
y1=[2,6,3]
plt.bar(x,y)
plt.bar(x1,y1)
```
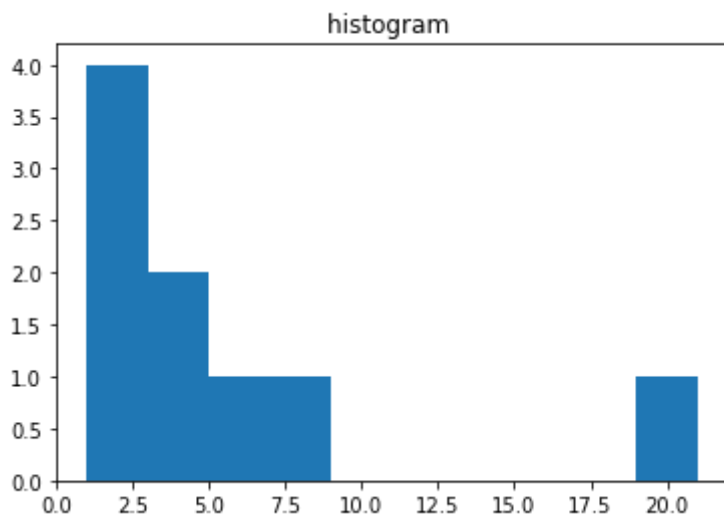
```
<BarContainer object of 3 artists>
```



```
#histograms
```

```
a=np.array([1,2,3,4,5,1,2,7,21])
plt.hist(a)
plt.title('histogram ')
```

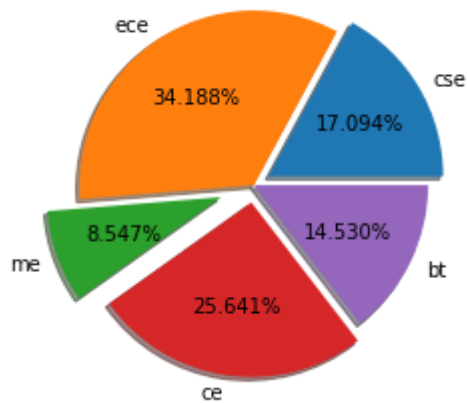        Text(0.5, 1.0, 'histogram ')



```
#pie plot
```

```
labels=['cse','ece','me','ce','bt']
size=[100,200,50,150,85]
colors=['red','blue','golden','grey','green']
explode=(0.1,0,0.2,0.1,0) #gap between slices
plt.pie(size,explode=explode,labels=labels, shadow=True,autopct='%1.3f%%')
```

    ➡

```
([<matplotlib.patches.Wedge at 0x7f5d1861d050>,
  <matplotlib.patches.Wedge at 0x7f5d1861da10>,
  <matplotlib.patches.Wedge at 0x7f5d186245d0>,
  <matplotlib.patches.Wedge at 0x7f5d185af250>,
  <matplotlib.patches.Wedge at 0x7f5d185bc0d0>],
```

```
  Text(-1.221600364236409, -0.4446263038749206, 'me'),
  Text(-0.17657438785630794, -1.186937860864321, 'ce'),
  Text(0.9873752043868569, -0.4848610169543564, 'bt')],
 [Text(0.6014643214920522, 0.3581070649569836, '17.094%'),
  Text(-0.3274587358887896, 0.5027631413400508, '34.188%'),
  Text(-0.7517540702993285, -0.2736161869999511, '8.547%'),
  Text(-0.10300172624951297, -0.6923804188375206, '25.641%'),
  Text(0.5385682933019219, -0.2644696456114671, '14.530%')])
```