series

```
import pandas as pd
mydata1=["kavya","bhavya","navya"]
ser1=pd.Series(mydata1)
print(ser1)
```

```
0     kavya
1    bhavya
2     navya
dtype: object
```

```
import pandas as pd
mydata1=["kavya","bhavya","navya"]
roll=[1,2,3]
ser1=pd.Series(mydata1)
print(ser1)
```

```
0     kavya
1    bhavya
2     navya
dtype: object
```

```
ser1=pd.Series(mydata1,index=roll)
print(ser1)
```

```
1     kavya
2    bhavya
3     navya
dtype: object
```

```
ser1.to_csv(r"C:\Users\DELL\Downloads\mydata1.csv")
```

```
ser1.to_csv(r"C:\my pythonfiles\mydata1.csv")
```

dataframes

```
import pandas as pd
d={"Names":["kavya","bhavya","navya"],
   "Age":[20,19,21],
   "city":["tumkur","mysore","banglore"]}
print(d)
```

```
{'Names': ['kavya', 'bhavya', 'navya'], 'Age': [20, 19, 21], 'city':
['tumkur', 'mysore', 'banglore']}
```

```
d_df=pd.DataFrame(d)
print(d_df)
```

```
    Names  Age      city
0   kavya   20    tumkur
```

```
1  bhavya   19    mysore
2   navya   21  banglore
```

```
d_df.to_csv(r"C:\my pythonfiles\dataframe.csv")
```

```
df1=pd.read_csv(r"C:\my pythonfiles\sampledata.csv")
```

```
df1.head()
```

```
      name dept  sem1  sem2  sem3
0      sam   IS   9.0   8.9   7.7
1  monisha   IS   7.0   7.6   8.1
2    navya  ISE   9.0   NaN   9.0
3    Aishu  ISE   9.5  10.0   9.1
4   nayana  ISE   NaN   8.9   6.8
```

load large file

```
diabetes_df=pd.read_csv(r"C:\my pythonfiles\diabetcsvsmall.csv")
```

```
diabetes_df.head()
```

```
   preg  plas  pres  skin  insu  mass   pedi  age            class
0   6.0   148  72.0  35.0     0  33.6  0.627   50  tested_positive
1   1.0    85  66.0  29.0     0  26.6  0.351   31  tested_negative
2   8.0   183  64.0   0.0     0  23.3  0.672   32  tested_positive
3   1.0    89  66.0  23.0    94  28.1  0.167   21  tested_negative
4   0.0   137  40.0  35.0   168  43.1  2.288   33  tested_positive
```

```
diabetes_df.tail()
```

```
     preg  plas  pres  skin  insu  mass   pedi  age            class
97    1.0    71  48.0   NaN    76  20.4  0.323   22  tested_negative
98    6.0    93  50.0  30.0    64  28.7  0.356   23  tested_negative
99    NaN   122  90.0  51.0   220  49.7  0.325   31  tested_positive
100   1.0   163  72.0   0.0     0  39.0  1.222   33  tested_positive
101   1.0   151  60.0   0.0     0  26.1  0.179   22  tested_negative
```

```
diabetes_df.loc[12:19,"age"]
```

```
12     57
13     59
14     51
15     32
16     31
17     31
18     33
19     32
Name: age, dtype: int64
```

```
diabetes_df.loc[12:19]
```

```
    preg  plas  pres  skin  insu  mass   pedi  age            class
12  10.0   139  80.0   0.0     0  27.1  1.441   57  tested_negative
13   1.0   189  60.0  23.0   846  30.1  0.398   59  tested_positive
14   5.0   166  72.0  19.0   175  25.8  0.587   51  tested_positive
15   7.0   100   0.0   0.0     0  30.0  0.484   32  tested_positive
16   0.0   118  84.0  47.0   230  45.8  0.551   31  tested_positive
17   7.0   107  74.0   0.0     0  29.6  0.254   31  tested_positive
18   1.0   103  30.0  38.0    83  43.3  0.183   33  tested_negative
19   1.0   115  70.0  30.0    96  34.6  0.529   32  tested_positive

diabetes_df.iloc[12:19,3:8]          #[row range,column range] ,

    skin  insu  mass   pedi  age
12   0.0     0  27.1  1.441   57
13  23.0   846  30.1  0.398   59
14  19.0   175  25.8  0.587   51
15   0.0     0  30.0  0.484   32
16  47.0   230  45.8  0.551   31
17   0.0     0  29.6  0.254   31
18  38.0    83  43.3  0.183   33
```

feature eng (rename and create new column)

preg plas pres skin insu mass pedi age==> independent(Feature) class==> dependent==> target(depends on features)

```
diabetes_df.rename(columns={"plas":"glucose"},inplace=True)

diabetes_df.head()

   preg  glucose  pres  skin  insu  mass   pedi  age            class
\
0   6.0      148  72.0  35.0     0  33.6  0.627   50  tested_positive

1   1.0       85  66.0  29.0     0  26.6  0.351   31  tested_negative

2   8.0      183  64.0   0.0     0  23.3  0.672   32  tested_positive

3   1.0       89  66.0  23.0    94  28.1  0.167   21  tested_negative

4   0.0      137  40.0  35.0   168  43.1  2.288   33  tested_positive


   Glucose_in_mmol
0         8.214008
1         4.717505
2        10.156510
3         4.939505
4         7.603508

diabetes_df.loc[12:19,"glucose"]
```

```
12    139
13    189
14    166
15    100
16    118
17    107
18    103
19    115
Name: glucose, dtype: int64
```

```
diabetes_df['Glucose_in_mmol']=diabetes_df['glucose']/18.018
#convert glucose from mm to mmol and create new column
```

```
diabetes_df.head()
```

```
   preg  glucose  pres  skin  insu  mass   pedi  age            class
\
0   6.0      148  72.0  35.0     0  33.6  0.627   50  tested_positive

1   1.0       85  66.0  29.0     0  26.6  0.351   31  tested_negative

2   8.0      183  64.0   0.0     0  23.3  0.672   32  tested_positive

3   1.0       89  66.0  23.0    94  28.1  0.167   21  tested_negative

4   0.0      137  40.0  35.0   168  43.1  2.288   33  tested_positive


   Glucose_in_mmol
0         8.214008
1         4.717505
2        10.156510
3         4.939505
4         7.603508
```

```
filt_age=diabetes_df[diabetes_df['age']<30]
```

```
filt_age.head()
```

```
    preg  glucose  pres  skin  insu  mass   pedi  age            class
\
3    1.0       89  66.0  23.0    94  28.1  0.167   21  tested_negative

6    3.0       78  50.0  32.0    88  31.0  0.248   26  tested_positive

7   10.0      115   0.0   0.0     0  35.3  0.134   29  tested_negative

20   3.0      126  88.0  41.0   235  39.3  0.704   27  tested_negative

23   9.0      119  80.0  35.0     0  29.0  0.263   29  tested_positive
```

```
    Glucose_in_mmol
3          4.939505
6          4.329004
7          6.382506
20         6.993007
23         6.604507
```

```
glucose_below_100=diabetes_df[diabetes_df['glucose']<100]
glucose_below_100.head()
```

```
    preg  glucose  pres  skin  insu  mass   pedi  age            class
\
1    1.0       85  66.0  29.0     0  26.6  0.351   31  tested_negative

3    1.0       89  66.0  23.0    94  28.1  0.167   21  tested_negative

6    3.0       78  50.0  32.0    88  31.0  0.248   26  tested_positive

21   8.0       99  84.0   0.0     0  35.4  0.388   50  tested_negative

27   1.0       97  66.0  15.0   140  23.2  0.487   22  tested_negative


    Glucose_in_mmol
1          4.717505
3          4.939505
6          4.329004
21         5.494505
27         5.383505
```

```
glucose_above_100=diabetes_df[diabetes_df['glucose']>100]
glucose_above_100.head()
```

```
   preg  glucose  pres  skin  insu  mass   pedi  age            class
\
0   6.0      148  72.0  35.0     0  33.6  0.627   50  tested_positive

2   8.0      183  64.0   0.0     0  23.3  0.672   32  tested_positive

4   0.0      137  40.0  35.0   168  43.1  2.288   33  tested_positive

5   5.0      116  74.0   0.0     0  25.6  0.201   30  tested_negative

7  10.0      115   0.0   0.0     0  35.3  0.134   29  tested_negative


   Glucose_in_mmol
0         8.214008
2        10.156510
4         7.603508
```

```
5         6.438006
7         6.382506
```

create a filtered data set which has only the rows with age between 20 and 30

```
filt_age_20_to_30=diabetes_df[(diabetes_df['age']>20) &
(diabetes_df['age']<30)]
filt_age_20_to_30.head()
```

```
    preg  glucose  pres  skin  insu  mass   pedi  age            class  
\
3    1.0       89  66.0  23.0    94  28.1  0.167   21  tested_negative   

6    3.0       78  50.0  32.0    88  31.0  0.248   26  tested_positive   

7   10.0      115   0.0   0.0     0  35.3  0.134   29  tested_negative   

20   3.0      126  88.0  41.0   235  39.3  0.704   27  tested_negative   

23   9.0      119  80.0  35.0     0  29.0  0.263   29  tested_positive   


    Glucose_in_mmol  
3          4.939505  
6          4.329004  
7          6.382506  
20         6.993007  
23         6.604507  
```

grouping and deriving results

```
grouped_by_class_age=diabetes_df.groupby('class')['age'].mean()
grouped_by_class_age
```

```
class
tested_negative    31.238095
tested_positive    40.589744
Name: age, dtype: float64
```

```
grouped_by_class_insu=diabetes_df.groupby('class')['insu'].mean()
grouped_by_class_insu
```

```
class
tested_negative     52.571429
tested_positive    114.692308
Name: insu, dtype: float64
```

```
grouped_by_class_age_min=diabetes_df.groupby('class')['age'].min()
grouped_by_class_age_min
```

```
class
tested_negative    21
tested_positive    25
Name: age, dtype: int64
```

```
grouped_by_class_age_max=diabetes_df.groupby('class')['age'].max()
grouped_by_class_age_max
```

```
class
tested_negative    60
tested_positive    60
Name: age, dtype: int64
```

cleaning data handelling mulls

```
diabetes_df.isnull().sum()  #tell how many null values in a column
```

```
preg                   1
glucose                0
pres                   1
skin                   1
insu                   0
mass                   1
pedi                   1
age                    0
class                  0
Glucose_in_mmol    0
dtype: int64
```

```
diabetes_df.isnull()   #if null present shows true
```

```
      preg  glucose  pres  skin  insu  mass  pedi   age
class  \
0     False    False  False  False  False  False  False  False  False

1     False    False  False  False  False  False  False  False  False

2     False    False  False  False  False  False  False  False  False

3     False    False  False  False  False  False  False  False  False

4     False    False  False  False  False  False  False  False  False

..      ...      ...    ...    ...    ...    ...    ...    ...    ...

97    False    False  False   True  False  False  False  False  False

98    False    False  False  False  False  False  False  False  False

99     True    False  False  False  False  False  False  False  False
```

```
100  False    False  False  False  False  False  False  False  False

101  False    False  False  False  False  False  False  False  False


     Glucose_in_mmol
0              False
1              False
2              False
3              False
4              False
..               ...
97             False
98             False
99             False
100            False
101            False

[102 rows x 10 columns]
```

```
diabetes_df.info
```

```
<bound method DataFrame.info of      preg  glucose  pres  skin  insu
mass   pedi  age            class  \
0     6.0      148  72.0  35.0     0  33.6  0.627   50
tested_positive
1     1.0       85  66.0  29.0     0  26.6  0.351   31
tested_negative
2     8.0      183  64.0   0.0     0  23.3  0.672   32
tested_positive
3     1.0       89  66.0  23.0    94  28.1  0.167   21
tested_negative
4     0.0      137  40.0  35.0   168  43.1  2.288   33
tested_positive
..    ...      ...   ...   ...   ...   ...    ...  ...              ..
.
97    1.0       71  48.0   NaN    76  20.4  0.323   22
tested_negative
98    6.0       93  50.0  30.0    64  28.7  0.356   23
tested_negative
99    NaN      122  90.0  51.0   220  49.7  0.325   31
tested_positive
100   1.0      163  72.0   0.0     0  39.0  1.222   33
tested_positive
101   1.0      151  60.0   0.0     0  26.1  0.179   22
tested_negative

     Glucose_in_mmol
0           8.214008
1           4.717505
```

```
2          10.156510
3           4.939505
4           7.603508
..               ...
97          3.940504
98          5.161505
99          6.771007
100         9.046509
101         8.380508

[102 rows x 10 columns]>
```

```
diabetes_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 102 entries, 0 to 101
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   preg             101 non-null    float64
 1   glucose          102 non-null    int64
 2   pres             101 non-null    float64
 3   skin             101 non-null    float64
 4   insu             102 non-null    int64
 5   mass             101 non-null    float64
 6   pedi             101 non-null    float64
 7   age              102 non-null    int64
 8   class            102 non-null    object
 9   Glucose_in_mmol  102 non-null    float64
dtypes: float64(6), int64(3), object(1)
memory usage: 8.1+ KB
```

```
diabetes_df.dropna(inplace=True)   #inplace saves
diabetes_df
```

```
     preg  glucose  pres  skin  insu  mass   pedi  age
class  \
0     6.0      148  72.0  35.0     0  33.6  0.627   50
tested_positive
1     1.0       85  66.0  29.0     0  26.6  0.351   31
tested_negative
2     8.0      183  64.0   0.0     0  23.3  0.672   32
tested_positive
3     1.0       89  66.0  23.0    94  28.1  0.167   21
tested_negative
4     0.0      137  40.0  35.0   168  43.1  2.288   33
tested_positive
..    ...      ...   ...   ...   ...   ...     ...  ...                ..
.
95    6.0      144  72.0  27.0   228  33.9  0.255   40
```

```
tested_negative
96    2.0       92  62.0  28.0     0  31.6  0.130   24
tested_negative
98    6.0       93  50.0  30.0    64  28.7  0.356   23
tested_negative
100   1.0      163  72.0   0.0     0  39.0  1.222   33
tested_positive
101   1.0      151  60.0   0.0     0  26.1  0.179   22
tested_negative

     Glucose_in_mmol
0           8.214008
1           4.717505
2          10.156510
3           4.939505
4           7.603508
..               ...
95          7.992008
96          5.106005
98          5.161505
100         9.046509
101         8.380508

[98 rows x 10 columns]
```

```
diabetes_df.isnull()
```

```
      preg  glucose   pres   skin   insu   mass   pedi    age
class  \
0    False    False  False  False  False  False  False  False  False

1    False    False  False  False  False  False  False  False  False

2    False    False  False  False  False  False  False  False  False

3    False    False  False  False  False  False  False  False  False

4    False    False  False  False  False  False  False  False  False

..     ...      ...    ...    ...    ...    ...    ...    ...    ...

95   False    False  False  False  False  False  False  False  False

96   False    False  False  False  False  False  False  False  False

98   False    False  False  False  False  False  False  False  False

100  False    False  False  False  False  False  False  False  False

101  False    False  False  False  False  False  False  False  False
```

```
     Glucose_in_mmol
0              False
1              False
2              False
3              False
4              False
..               ...
95             False
96             False
98             False
100            False
101            False

[98 rows x 10 columns]
```

```
diabetes_df.isnull().sum()
```

```
preg               0
glucose            0
pres               0
skin               0
insu               0
mass               0
pedi               0
age                0
class              0
Glucose_in_mmol    0
dtype: int64
```

handelling duplicates

```
diabetes_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 98 entries, 0 to 101
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   preg             98 non-null     float64
 1   glucose          98 non-null     int64
 2   pres             98 non-null     float64
 3   skin             98 non-null     float64
 4   insu             98 non-null     int64
 5   mass             98 non-null     float64
 6   pedi             98 non-null     float64
 7   age              98 non-null     int64
 8   class            98 non-null     object
 9   Glucose_in_mmol  98 non-null     float64
```

```
dtypes: float64(6), int64(3), object(1)
memory usage: 8.4+ KB
```

```
diabetes_df.drop_duplicates(inplace=True)
diabetes_df
```

```
     preg  glucose  pres  skin  insu  mass   pedi  age
class  \
0     6.0      148  72.0  35.0     0  33.6  0.627   50
tested_positive
1     1.0       85  66.0  29.0     0  26.6  0.351   31
tested_negative
2     8.0      183  64.0   0.0     0  23.3  0.672   32
tested_positive
3     1.0       89  66.0  23.0    94  28.1  0.167   21
tested_negative
4     0.0      137  40.0  35.0   168  43.1  2.288   33
tested_positive
..    ...      ...   ...   ...   ...   ...    ...  ...                ..
.
95    6.0      144  72.0  27.0   228  33.9  0.255   40
tested_negative
96    2.0       92  62.0  28.0     0  31.6  0.130   24
tested_negative
98    6.0       93  50.0  30.0    64  28.7  0.356   23
tested_negative
100   1.0      163  72.0   0.0     0  39.0  1.222   33
tested_positive
101   1.0      151  60.0   0.0     0  26.1  0.179   22
tested_negative

     Glucose_in_mmol
0           8.214008
1           4.717505
2          10.156510
3           4.939505
4           7.603508
..               ...
95          7.992008
96          5.106005
98          5.161505
100         9.046509
101         8.380508

[96 rows x 10 columns]
```

```
diabetes_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 96 entries, 0 to 101
```

```
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   preg             96 non-null     float64
 1   glucose          96 non-null     int64  
 2   pres             96 non-null     float64
 3   skin             96 non-null     float64
 4   insu             96 non-null     int64  
 5   mass             96 non-null     float64
 6   pedi             96 non-null     float64
 7   age              96 non-null     int64  
 8   class            96 non-null     object 
 9   Glucose_in_mmol  96 non-null     float64
dtypes: float64(6), int64(3), object(1)
memory usage: 8.2+ KB
```

reading other formates

```
dia_ex=pd.read_excel(r"C:\my pythonfiles\diabetes.xlsx")
dia_ex.head()
```

```
   preg  plas  pres  skin  insu  mass   pedi  age            class
0     6   148    72    35     0  33.6  0.627   50  tested_positive
1     1    85    66    29     0  26.6  0.351   31  tested_negative
2     8   183    64     0     0  23.3  0.672   32  tested_positive
3     1    89    66    23    94  28.1  0.167   21  tested_negative
4     0   137    40    35   168  43.1  2.288   33  tested_positive
```

```
dia_ex2=pd.read_excel(r"C:\my pythonfiles\
diabetes.xlsx",sheet_name='dora')
dia_ex2
```

```
  Dead Alive
0  yes    no
1  yes    no
2  yes    no
3  yes    no
4  yes    no
```

```
dia_ex3=pd.read_excel(r"C:\my pythonfiles\
diabetes.xlsx",sheet_name='Hello')
dia_ex3
```

```
Empty DataFrame
Columns: [hello, guys, how, are ]
Index: []
```

```
df_txt=pd.read_csv(r"C:\my pythonfiles\grades.txt")
df_txt.head()
```

```
  Names Initials SEM1 SEM2 SEM3 Grade
0                Joe K 9.8 10 9.9 A+
1             Rajesh M 8.9 9.1 9.3 A
2             Kissan V 9.9 9.3 9.2 A
3                 Mary N 7.7 8 7.1 B
4              Jeen K 9.8 9.1 9.9 A+
```

```
df_txt=pd.read_csv(r"C:\my pythonfiles\grades.txt",sep=' ')
df_txt.head()
```

```
    Names Initials  SEM1  SEM2  SEM3 Grade
0     Joe        K   9.8  10.0   9.9    A+
1  Rajesh        M   8.9   9.1   9.3     A
2  Kissan        V   9.9   9.3   9.2     A
3    Mary        N   7.7   8.0   7.1     B
4    Jeen        K   9.8   9.1   9.9    A+
```

modifying data type

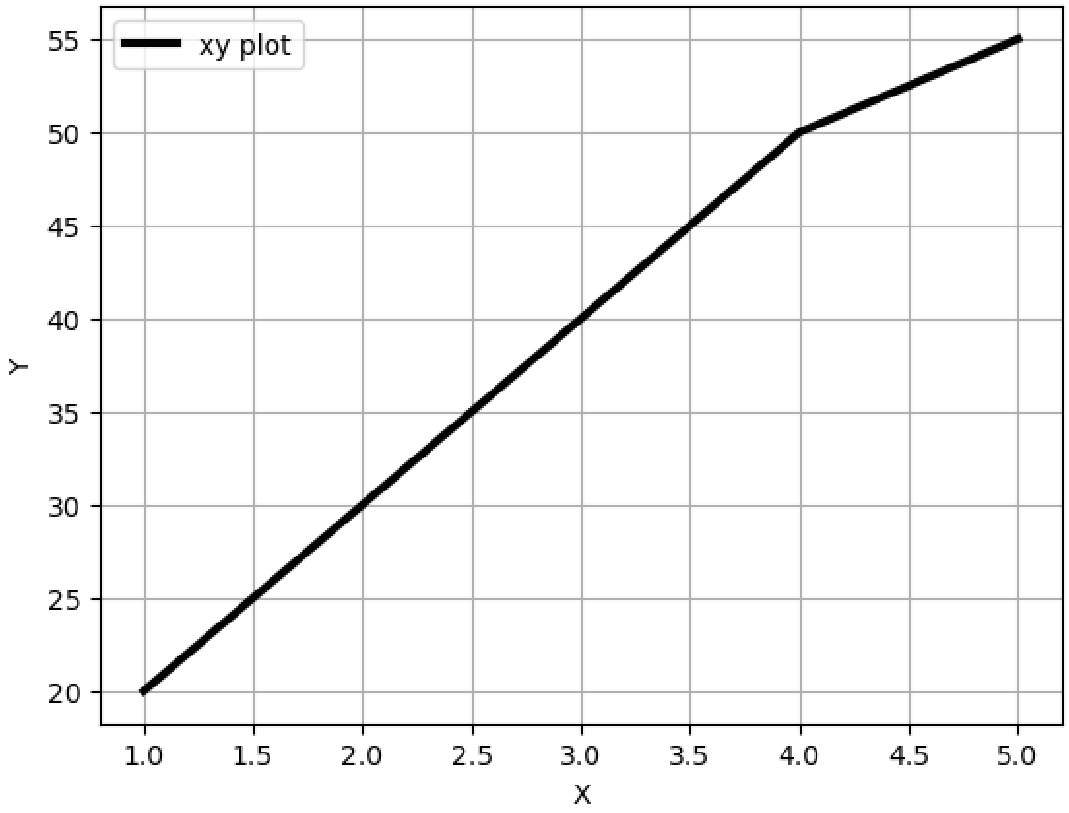
```
df_txt['SEM1_int']=df_txt['SEM1'].astype(int)
df_txt.head()
```

```
    Names Initials  SEM1  SEM2  SEM3 Grade  SEM1_int
0     Joe        K   9.8  10.0   9.9    A+         9
1  Rajesh        M   8.9   9.1   9.3     A         8
2  Kissan        V   9.9   9.3   9.2     A         9
3    Mary        N   7.7   8.0   7.1     B         7
4    Jeen        K   9.8   9.1   9.9    A+         9
```

matplot libraries

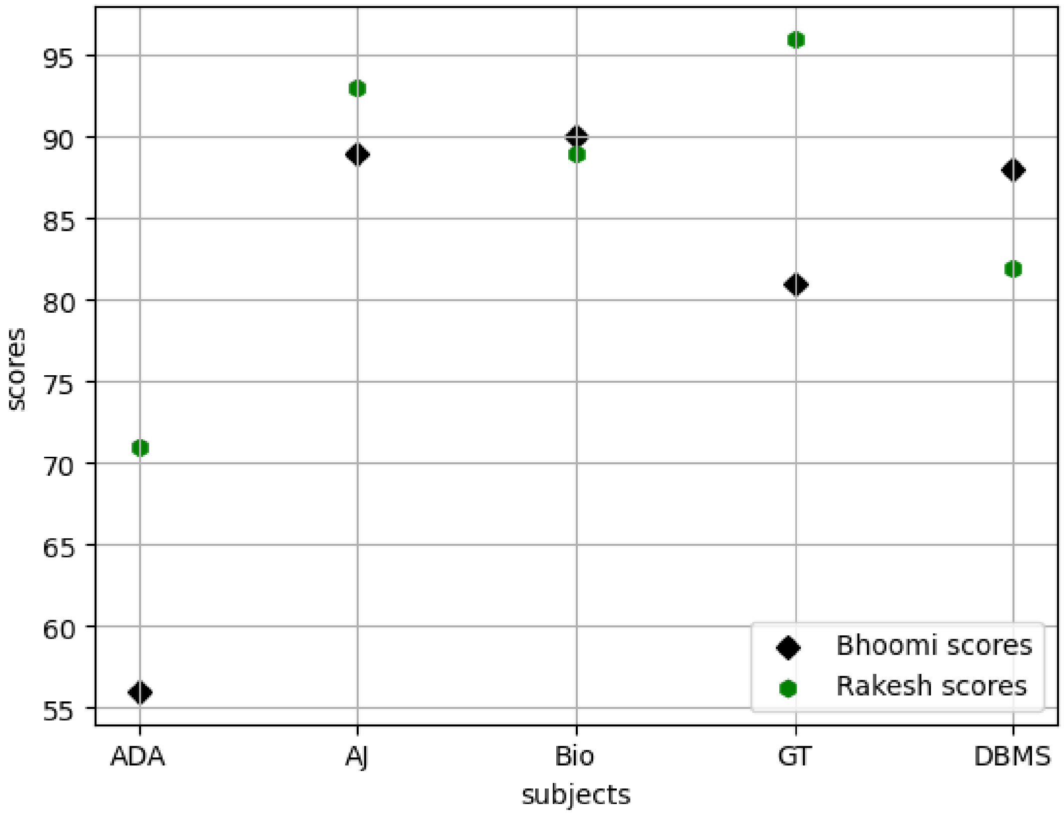
```
x=[1,2,3,4,5]
y=[20,30,40,50,55]

import matplotlib.pyplot as plt
plt.plot(x,y,color='k',label="xy plot",linestyle="-",linewidth=3)
plt.xlabel("X")
plt.ylabel("Y")
plt.grid()
plt.legend()
```

```
<matplotlib.legend.Legend at 0x1d1f113a110>
```



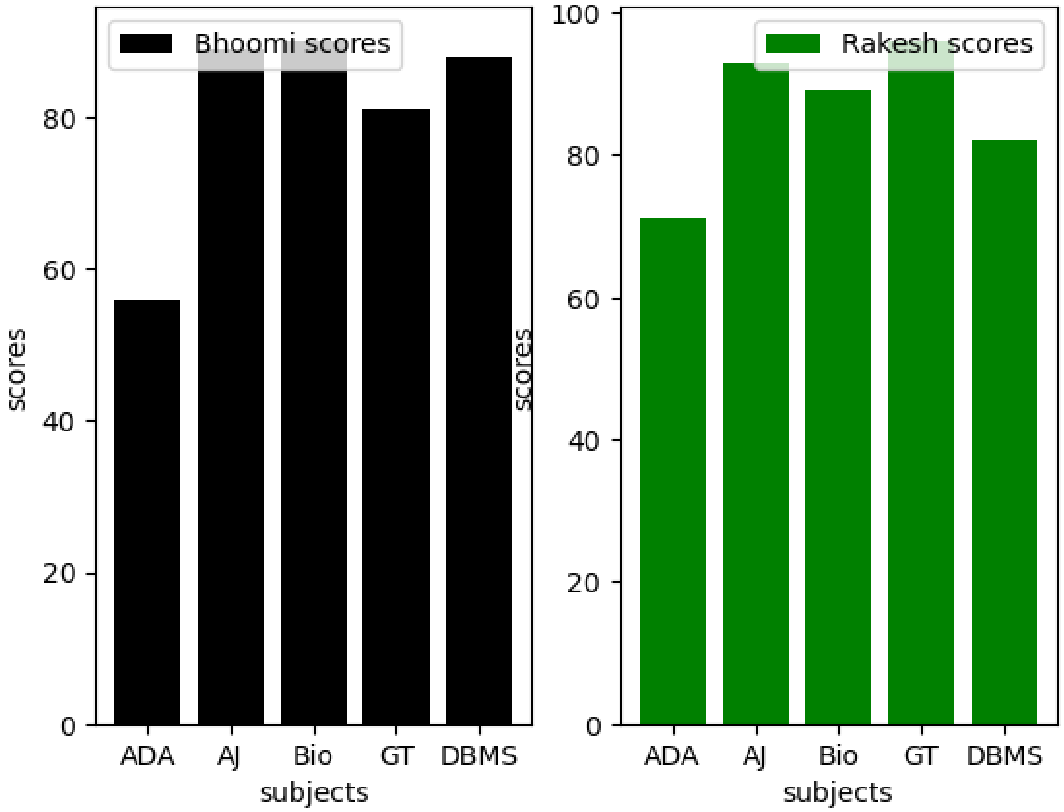
```python
import matplotlib.pyplot as plt
sub=["ADA","AJ","Bio","GT","DBMS"]
bhoomi=[56,89,90,81,88]
rakesh=[71,93,89,96,82]
plt.scatter(sub,bhoomi,color='k',label="Bhoomi scores",marker='D')
plt.scatter(sub,rakesh,color='green',label="Rakesh scores",marker='h')
plt.xlabel("subjects")
plt.ylabel("scores")
plt.grid()
plt.legend()
```

```
<matplotlib.legend.Legend at 0x1d1f113a050>
```

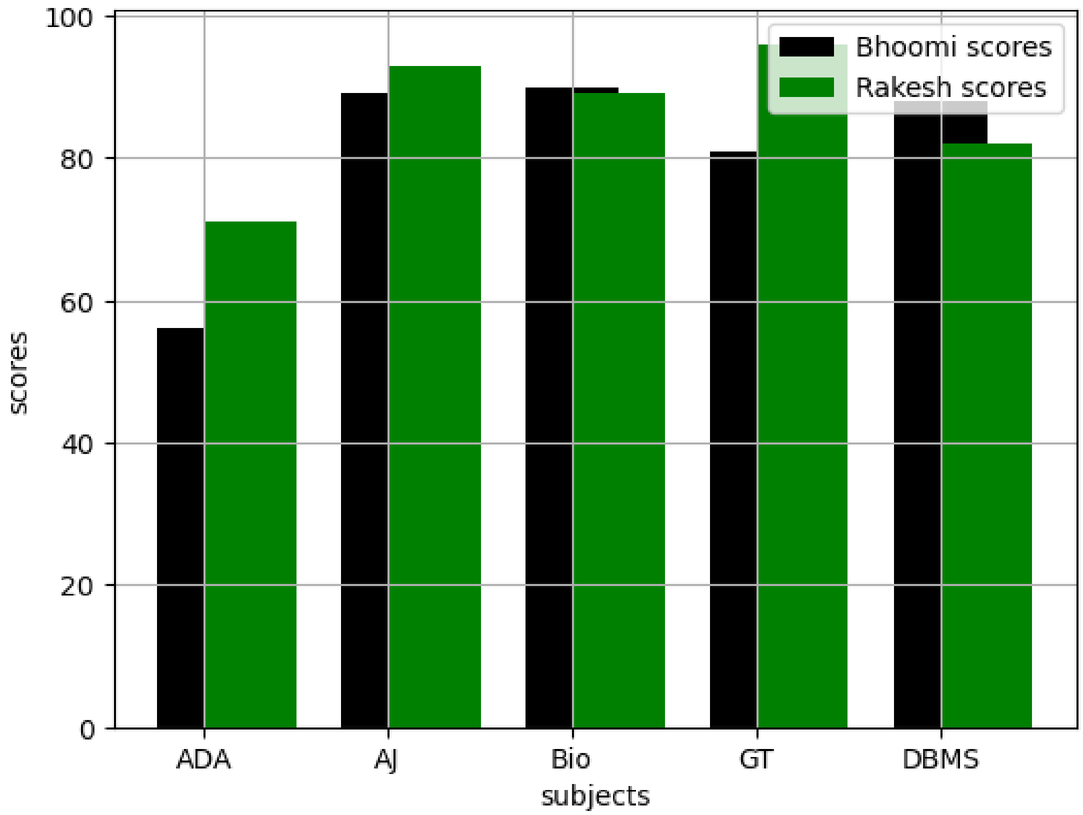


```
import matplotlib.pyplot as plt
sub=["ADA","AJ","Bio","GT","DBMS"]
bhoomi=[56,89,90,81,88]
rakesh=[71,93,89,96,82]
plt.subplot(1,2,1)
plt.bar(sub,bhoomi,color='k',label="Bhoomi scores")
plt.xlabel("subjects")
plt.ylabel("scores")
plt.legend()
plt.subplot(1,2,2)
plt.bar(sub,rakesh,color='green',label="Rakesh scores")
plt.xlabel("subjects")
plt.ylabel("scores")
plt.legend()
```

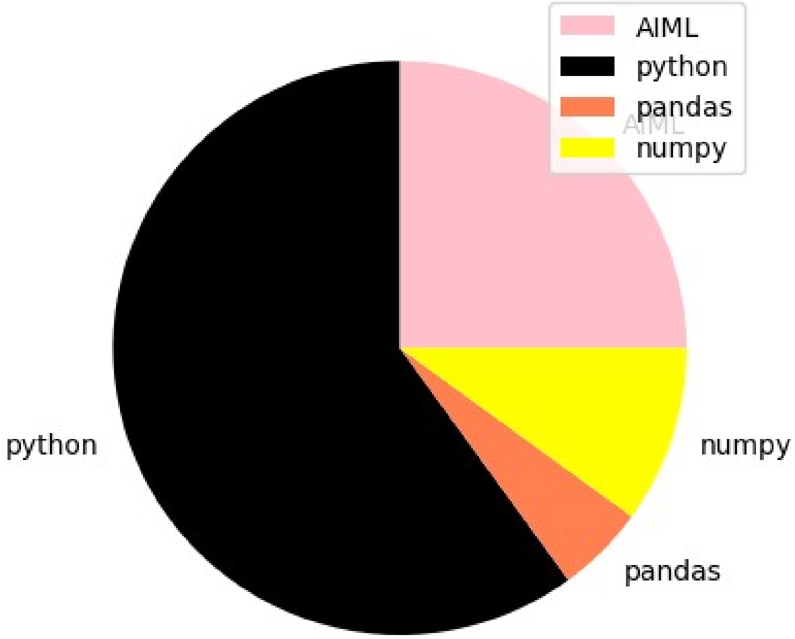
```
<matplotlib.legend.Legend at 0x1d1f587fb90>
```



```
import matplotlib.pyplot as plt
sub=["ADA","AJ","Bio","GT","DBMS"]
bhoomi=[56,89,90,81,88]
rakesh=[71,93,89,96,82]
plt.bar(sub,bhoomi,color='k',label="Bhoomi
scores",width=0.5,align="center")
plt.bar(sub,rakesh,color='green',label="Rakesh
scores",width=0.5,align="edge")
plt.xlabel("subjects")
plt.ylabel("scores")
plt.grid()
plt.legend()

<matplotlib.legend.Legend at 0x1d1f440b0d0>
```



```
import numpy as np
a=np.array([25,60,5,10])
label=["AIML","python","pandas","numpy"]
color=['pink','Black','coral','yellow']
plt.pie(a,labels=label,colors=color)
plt.legend()
plt.show()
```


AIML
python
pandas
numpy
AIML
python
numpy
pandas