ASSIGNMENT NO:1

Name: Pathak diksha Vilas

Roll NO: 58

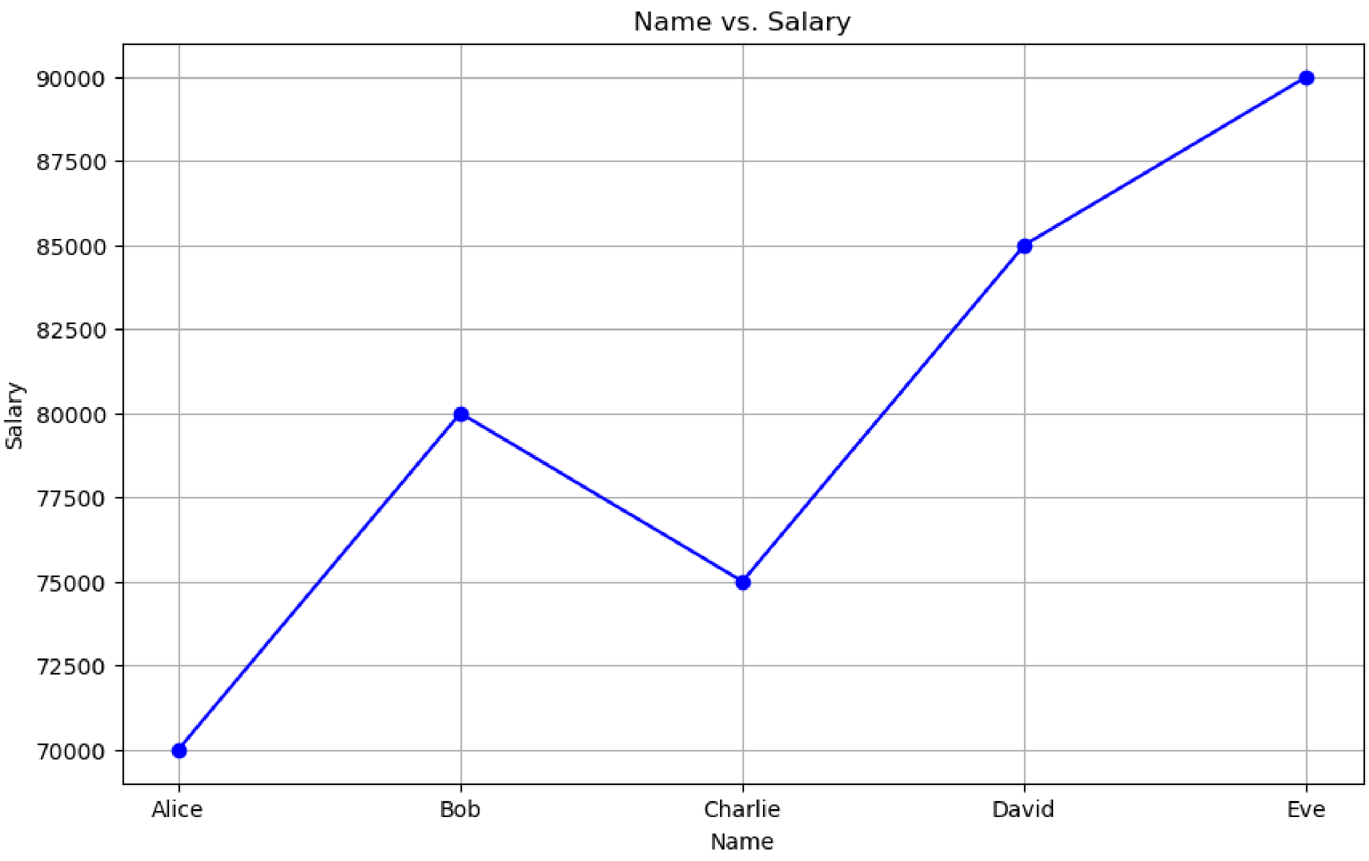
Batch: 3

Class:TYBSC(COMP.SCI)

slip 2,6,(B)

Write a python program to generate a line plot of name Vs salary

In [7]:
```
import pandas as pd
import matplotlib.pyplot as plt
data={
    'name':['Alice','Bob','Charlie','David','Eve'],
    'salary':[70000,80000,75000,85000,90000]
}
df = pd.DataFrame(data)
plt.figure(figsize=(10, 6))
plt.plot(data['name'],data['salary'], marker='o', linestyle='-', color='b')
plt.title('Name vs. Salary')
plt.xlabel('Name')
plt.ylabel('Salary')
plt.grid(True)
plt.show()
```



Slip 1,3,11,14(B)

B) Write a Python program to view basic statistical details of the data.(Use wineequality-red.csv)

In [8]:
```
import pandas as pd
file_path='winequality-red.csv'
data=pd.read_csv(file_path)
```

In [9]:
```
statistical_summary=data.describe()
```

In [10]:
```
print("Basic Statistical Details of the Data:")
print(statistical_summary)
```

```
Basic Statistical Details of the Data:
       fixed acidity  volatile acidity  citric acid  residual sugar  \
count    1599.000000       1599.000000  1599.000000     1599.000000
mean        8.319637          0.527821     0.270976        2.538806
std         1.741096          0.179060     0.194801        1.409928
min         4.600000          0.120000     0.000000        0.900000
25%         7.100000          0.390000     0.090000        1.900000
50%         7.900000          0.520000     0.260000        2.200000
75%         9.200000          0.640000     0.420000        2.600000
max        15.900000          1.580000     1.000000       15.500000

         chlorides  free sulfur dioxide  total sulfur dioxide      density  \
count  1599.000000          1599.000000           1599.000000  1599.000000
mean      0.087467            15.874922             46.467792     0.996747
std       0.047065            10.460157             32.895324     0.001887
min       0.012000             1.000000              6.000000     0.990070
25%       0.070000             7.000000             22.000000     0.995600
50%       0.079000            14.000000             38.000000     0.996750
75%       0.090000            21.000000             62.000000     0.997835
max       0.611000            72.000000            289.000000     1.003690

                pH    sulphates      alcohol      quality
count  1599.000000  1599.000000  1599.000000  1599.000000
mean      3.311113     0.658149    10.422983     5.636023
std       0.154386     0.169507     1.065668     0.807569
min       2.740000     0.330000     8.400000     3.000000
25%       3.210000     0.550000     9.500000     5.000000
50%       3.310000     0.620000    10.200000     6.000000
75%       3.400000     0.730000    11.100000     6.000000
max       4.010000     2.000000    14.900000     8.000000
```

slip2,slip 6(c)

Q.2 C) Download the heights and weights dataset and load the dataset froma given csv file into a dataframe. Print the first, last 10 rows and random 20 rows also display shape of the dataset. [5]

In [11]:
```
file_path='SOCR-HeightWeight.csv'
data=pd.read_csv(file_path)
```

In [12]:
```
print("First 10 Rows of the Dataset:")
print(data.head(10))
```

```
First 10 Rows of the Dataset:
   Index  Height(Inches)  Weight(Pounds)
0      1        65.78331        112.9925
1      2        71.51521        136.4873
2      3        69.39874        153.0269
3      4        68.21660        142.3354
4      5        67.78781        144.2971
5      6        68.69784        123.3024
6      7        69.80204        141.4947
7      8        70.01472        136.4623
8      9        67.90265        112.3723
9     10        66.78236        120.6672
```

In [13]:
```
print("\nLast 10 Rows of the Dataset:")
print(data.tail(10))
```

```
Last 10 Rows of the Dataset:
       Index  Height(Inches)  Weight(Pounds)
24990  24991        69.97767        125.3672
24991  24992        71.91656        128.2840
24992  24993        70.96218        146.1936
24993  24994        66.19462        118.7974
24994  24995        67.21126        127.6603
24995  24996        69.50215        118.0312
24996  24997        64.54826        120.1932
24997  24998        64.69855        118.2655
24998  24999        67.52918        132.2682
24999  25000        68.87761        124.8742
```

In [14]:
```
print("\nLast 20 Rows of the Dataset:")
print(data.sample(20))
```

```
Last 20 Rows of the Dataset:
       Index  Height(Inches)  Weight(Pounds)
9933    9934        66.30375        120.6304
3761    3762        68.53583        115.0374
14975  14976        71.59953        129.4179
18431  18432        68.76597        122.4740
19809  19810        70.67327        147.0100
14985  14986        68.23971        132.4932
8305    8306        71.60898        137.8791
3000    3001        70.51338        143.7382
14108  14109        66.48817        111.8809
11888  11889        70.97371        137.9285
13513  13514        68.07050        145.1079
623      624        66.98337        116.5401
7526    7527        66.09344        143.7256
1663    1664        70.99035        117.7071
15236  15237        64.68249        116.3063
1903    1904        65.71651        134.9238
1560    1561        69.73600        132.9405
21421  21422        67.88258        124.2948
11716  11717        70.32862        133.2887
18580  18581        69.09877        125.7953
```

In [15]:
```python
print("\nShape of the Dataset:",data.shape)
```

```
Shape of the Dataset: (25000, 3)
```

slip 4,5(B)

b) write a python program to print the shape, number of rows-columns,data types,feature names and the description of the data(User_Data)

In [16]:
```python
import pandas as pd
file_path ='User_Data.csv'
data=pd.read_csv(file_path)
```

In [18]:
```python
print("Shape of the Data:",data.shape)
```

```
Shape of the Data: (400, 5)
```

In [19]:
```python
print("Number of Rows",data.shape[0])
print("Number of Columns",data.shape[1])
```

```
Number of Rows 400
Number of Columns 5
```

In [20]:
```python
print("\nData Types of Each Column:")
print(data.dtypes)
```

```
Data Types of Each Column:
User ID             int64
Gender             object
Age                 int64
EstimatedSalary     int64
Purchased           int64
dtype: object
```

In [21]:
```python
print("\nfeatures Names:")
print(data.columns.tolist())
```

```
features Names:
['User ID', 'Gender', 'Age', 'EstimatedSalary', 'Purchased']
```

In [22]:
```python
print("\n description of the data:/n",data.describe())
```

```
 description of the data:/n             User ID         Age  EstimatedSalary   Purchased
count  4.000000e+02  400.000000       400.000000  400.000000
mean   1.569154e+07   37.655000     69742.500000    0.357500
std    7.165832e+04   10.482877     34096.960282    0.479864
min    1.556669e+07   18.000000     15000.000000    0.000000
25%    1.562676e+07   29.750000     43000.000000    0.000000
50%    1.569434e+07   37.000000     70000.000000    0.000000
75%    1.575036e+07   46.000000     88000.000000    1.000000
max    1.581524e+07   60.000000    150000.000000    1.000000
```

In [23]:
```python
print(":\nFirst 5 Rows of the Data:",data.head())
```

```
:
First 5 Rows of the Data:     User ID  Gender  Age  EstimatedSalary  Purchased
0  15624510    Male   19            19000          0
1  15810944    Male   35            20000          0
2  15668575  Female   26            43000          0
3  15603246  Female   27            57000          0
4  15804002    Male   19            76000          0
```

In [24]:
```python
print(":\nLast 5 Rows of the Data:",data.tail())
```

```
:
Last 5 Rows of the Data:       User ID  Gender  Age  EstimatedSalary  Purchased
395  15691863  Female   46            41000          1
396  15706071    Male   51            23000          1
397  15654296  Female   50            20000          1
398  15755018    Male   36            33000          0
399  15594041  Female   49            36000          1
```

Slip 17(b)

Q.2 B) Write a Python program to create a data frame containing columns name, age , salary,department . Add 10 rows to the data frame. View the data frame. [5]

In [25]:
```
import pandas as pd
import numpy as np
```

In [26]:
```
data={
       'name':['Alice','Bob','Charlie','David','Eve','Frank','Grace','Hannah','Issac','Judy'],
        'age':[25,30,40,45,50,55,60,65,70,75],
        'salary':[700000,80000,75000,90000,89999,50000,78000,55000,33000,60000],
         'department':['HR','Finance','It','Marketing','Sales','HR','Finance','It','Marketing','Sales']




}
df=pd.DataFrame(data)
print(df)
```

```
     name  age  salary department
0    Alice   25  700000         HR
1      Bob   30   80000    Finance
2  Charlie   40   75000         It
3    David   45   90000  Marketing
4      Eve   50   89999      Sales
5    Frank   55   50000         HR
6    Grace   60   78000    Finance
7   Hannah   65   55000         It
8    Issac   70   33000  Marketing
9     Judy   75   60000      Sales
```

Slip 19,28(1,2)

to create a dataframe containing columns name,age and percentage .Add 10 rows to the dataframe.View the dataframe

In [27]:
```
import pandas as pd
data = {
    'name': ['Alice','Bob', 'Charlie','David','Eve','Frank','Grace','Issac','Judy','Hannah'],
    'age':[25,30,40,45,50,55,60,65,70,23],
    'percentage':[85.5,90.3,77.8,88.9,91.10,66.10,90,87,88,90]
}
df=pd.DataFrame(data)
print("DataFrame:")
print(df)
```

```
DataFrame:
     name  age  percentage
0    Alice   25        85.5
1      Bob   30        90.3
2  Charlie   40        77.8
3    David   45        88.9
4      Eve   50        91.1
5    Frank   55        66.1
6    Grace   60        90.0
7    Issac   65        87.0
8     Judy   70        88.0
9   Hannah   23        90.0
```

In [28]:
```
print("\nShape of the DataFrame:",df.shape)
```

```
Shape of the DataFrame: (10, 3)
```

In [29]:
```
print("\nNumber ofrows:", df.shape[0])
print("\nNumber of columns:", df.shape[1])
```

```
Number ofrows: 10

Number of columns: 3
```

In [30]:
```
print("\nData types:\n",df.dtypes)
```

```
Data types:
 name          object
age            int64
percentage   float64
dtype: object
```

```
In [31]: print("\nDescription of the data:\n",df.describe())
```

```
Description of the data:
              age  percentage
count  10.000000    10.00000
mean   46.300000    85.47000
std    16.640312     7.82163
min    23.000000    66.10000
25%    32.500000    85.87500
50%    47.500000    88.45000
75%    58.750000    90.00000
max    70.000000    91.10000
```

SLIP 18 (B)

Q.2 B) Use the heights and weights dataset and load the dataset from a given csv file into a dataframe. Print the first, last 5 rows and random 10 row [5]

```
In [32]: import pandas as pd
         file_path='SOCR-HeightWeight.csv'
         df=pd.read_csv(file_path)
         print("First 5 rows of the dataframe:")
         print(df.head())
         print("\n last 5 rows of the dataframe:")
         print(df.tail())
         print("\n 10 random rows of the dataframe:")
         print(df.sample(n=10))
```

```
First 5 rows of the dataframe:
   Index  Height(Inches)  Weight(Pounds)
0      1        65.78331        112.9925
1      2        71.51521        136.4873
2      3        69.39874        153.0269
3      4        68.21660        142.3354
4      5        67.78781        144.2971

 last 5 rows of the dataframe:
       Index  Height(Inches)  Weight(Pounds)
24995  24996        69.50215        118.0312
24996  24997        64.54826        120.1932
24997  24998        64.69855        118.2655
24998  24999        67.52918        132.2682
24999  25000        68.87761        124.8742

 10 random rows of the dataframe:
       Index  Height(Inches)  Weight(Pounds)
16959  16960        70.52264        146.0748
5048    5049        67.60067        123.7417
12385  12386        66.65667        127.9004
1706    1707        68.89125        139.8464
9503    9504        69.28147        140.9313
1602    1603        67.15097        136.0613
4279    4280        64.39313        123.2547
5590    5591        67.32651        133.7645
15424  15425        68.71162        136.6454
3529    3530        68.18232        107.0685
```

```
In [ ]:
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js