# Assignment No. 3

Aim : Descriptive Statistics - Measures of Central Tendency and variability Perform the following operations on any open source dataset (e.g., data.csv).

1. Provide summary statistics (mean, median, minimum, maximum, standard deviation) for a dataset (age, income etc.) with numeric variables grouped by one of the qualitative (categorical) variable. For example, if your categorical variable is age groups and quantitative variable is income, then provide summary statistics of income grouped by the age groups. Create a list that contains a numeric value for each response to the categorical variable.
2. Write a Python program to display some basic statistical details like percentile, mean, standard deviation etc. of the species of 'Iris-setosa', 'Iris-versicolor' and 'Iris-versicolor' of iris.csv dataset.

   Provide the codes with outputs and explain everything that you do in this step.

   Code :

In [1]:
```
import pandas as pd

df1 = pd.read_csv("Customers.csv")
df1
```

Out[1]:

| | CustomerID | Genre | Age | Annual_income_(k$) | Spending_score |
|---|---|---|---|---|---|
| **0** | 37 | male | 53 | 102 | 20 |
| **1** | 25 | male | 42 | 94 | 92 |
| **2** | 36 | male | 52 | 124 | 30 |
| **3** | 16 | male | 29 | 27 | 25 |
| **4** | 184 | male | 47 | 118 | 18 |
| **...** | ... | ... | ... | ... | ... |
| **194** | 37 | male | 22 | 33 | 16 |
| **195** | 75 | male | 30 | 82 | 71 |
| **196** | 18 | male | 39 | 85 | 86 |
| **197** | 183 | female | 78 | 130 | 30 |
| **198** | 129 | female | 52 | 50 | 75 |

199 rows × 5 columns

```
]: column_name = '
column_
print(c
```

```
 0
```

```
In [2]:                CustomerID'
       mean = df1["CustomerID"].mean()
       olumn_mean)
```

```
1 6.74371859296483
```

```
In [3  column_name = 'Annual_income_(k$)'
column_mean = df1["Annual_income_(k$)"].mean()
print(column_mean)
```

```
82.84422110552764
```

```
In [4]: column_name = 'Spending_score'
column_mean = df1["Spending_score"].mean()
print(column_mean)
```

```
50.120603015075375
```

```
In [12]: df1['Row_Mean'] = df1[['CustomerID', 'Spending_score']].mean(axis=1)

print(df1)
```

```
     CustomerID   Genre  Age  Annual_income_(k$)  Spending_score  Row_Mean
0            37    male   53                 102              20      28.5
1            25    male   42                  94              92      58.5
2            36    male   52                 124              30      33.0
3            16    male   29                  27              25      20.5
4           184    male   47                 118              18     101.0
..          ...     ...  ...                 ...             ...       ...
194          37    male   22                  33              16      26.5
195          75    male   30                  82              71      73.0
196          18    male   39                  85              86      52.0
197         183  female   78                 130              30     106.5
198         129  female   52                  50              75     102.0

[199 rows x 6 columns]
```

```
In [13]: column_name = 'CustomerID'
column_median = df1["CustomerID"].median()
print(column_median)
```

```
111.0
```

```
In [14]: column_name = 'Spending_score'
column_median = df1["Spending_score"].median()
print(column_median)
```

```
48.0
```

In [ ]:
```
    t
```

```
15  df1['Row_Median'] = df1[['CustomerID', 'Spending_score']].median(axis=1)

prin (df1)
```

```
     CustomerID   Genre  Age  Annual_income_(k$)  Spending_score  Row_Mean  \
0            37    male   53                 102              20      28.5
1            25    male   42                  94              92      58.5
2            36    male   52                 124              30      33.0
3            16    male   29                  27              25      20.5
4           184    male   47                 118              18     101.0
..          ...     ...  ...                 ...             ...       ...
194          37    male   22                  33              16      26.5
195          75    male   30                  82              71      73.0
196          18    male   39                  85              86      52.0
197         183  female   78                 130              30     106.5
198         129  female   52                  50              75     102.0

     Row_Median
0          28.5
1          58.5
2          33.0
3          20.5
4         101.0
..          ...
194        26.5
195        73.0
196        52.0
197       106.5
198       102.0

[199 rows x 7 columns]
```

In [9]:
```
column_name = 'Annual_income_(k$)'
column_mode = df1["Annual_income_(k$)"].mode()
print(column_mode)
```

```
0    33
dtype: int64
```

In [10]:
```
column_name = 'Age'
column_mode = df1["Age"].mode()
print(column_mode)
```

```
0    58
dtype: int64
```

In [16]:
```
column_name = 'CustomerID'
column_min = df1["CustomerID"].min()
print(column_min)
```

```
2
```

In [17]:
```
column_name = '  e
column_
print(c
```

```
0
```

In [  ]:
```
                Ag '
min = df1["Age"].min()
olumn_min)
```

In [ ]:
```
column_name = '  e
column_
print(c
```

```
 0
```

In [ ]:
```
2
```

18
```
df1['Row_Min'] = df1[['CustomerID', 'Spending_score']].min(axis=1)

print(df1)
```

```
     CustomerID   Genre  Age  Annual_income_(k$)  Spending_score  Row_Mean
\
0            37    male   53                 102              20      28.5
1            25    male   42                  94              92      58.5
2            36    male   52                 124              30      33.0
3            16    male   29                  27              25      20.5
4           184    male   47                 118              18     101.0
..          ...     ...  ...                 ...             ...       ...
194          37    male   22                  33              16      26.5
195          75    male   30                  82              71      73.0
196          18    male   39                  85              86      52.0
197         183  female   78                 130              30     106.5
198         129  female   52                  50              75     102.0

     Row_Median  Row_Min
0          28.5       20
1          58.5       25
2          33.0       30
3          20.5       16
```

In [19]:
```
column_name = 'Annual_income_(k$)'
column_min = df1["Annual_income_(k$)"].min()
print(column_min)
```

```
11
```

In [20]:
```
column_name = 'CustomerID'
column_min = df1["CustomerID"].min()
print(column_min)
```

```
2
```

In [22]:
```
column_name = 'CustomerID'
column_max = df1["CustomerID"].max()
print(column_max)
```

```
200
```

In [23]:
```
column_name = 'Age'
column_max = df1["Age"].max()
print(column_max)
```

```
80
```

In [ ]:
```
column_name = '  e
column_
print(c
```

```
 0
```

24
In [ ]:
```
               Sp nding_score'
       max = df1["Spending_score"].max()
       olumn_max)
```

```
1 0
```

25
```
df1['Row_Max'] = df1[['CustomerID', 'Age']].max(axis=1)

print(df1)
```

```
     CustomerID   Genre  Age  Annual_income_(k$)  Spending_score  Row_Mean  \
0            37    male   53                 102              20      28.5
1            25    male   42                  94              92      58.5
2            36    male   52                 124              30      33.0
3            16    male   29                  27              25      20.5
4           184    male   47                 118              18     101.0
..          ...     ...  ...                 ...             ...       ...
194          37    male   22                  33              16      26.5
195          75    male   30                  82              71      73.0
196          18    male   39                  85              86      52.0
197         183  female   78                 130              30     106.5
198         129  female   52                  50              75     102.0

     Row_Median  Row_Min  Row_Max
0          28.5       20       53
1          58.5       25       42
2          33.0       30       52
3          20.5       16       29
4         101.0       18      184
..          ...      ...      ...
194        26.5       16       37
195        73.0       71       75
196        52.0       18       39
197       106.5       30      183
198       102.0       75      129

[199 rows x 9 columns]
```

In [27]:
```
column_name = 'CustomerID'
column_standard = df1["CustomerID"].std()
print(column_standard)
```

```
59.00419132725263
```

In [28]:
```
column_name = 'Age'
column_standard = df1["Age"].std()
print(column_standard)
```

```
17.236379758179037
```

In [ ]:
```
column_name = '  e
column_
print(c
```

```
 0
```

29
In [ ]:
```
                Sp nding_score'
      standard = df1["Spending_score"].std()
      olumn_standard)
```

```
3 .427186269535365
```

30
```
df1['Row_Standard'] = df1[['CustomerID', 'Age']].std(axis=1)

print(df1)
```

```
     CustomerID   Genre  Age  Annual_income_(k$)  Spending_score  Row_Mean  \
0            37    male   53                 102              20      28.5
1            25    male   42                  94              92      58.5
2            36    male   52                 124              30      33.0
3            16    male   29                  27              25      20.5
4           184    male   47                 118              18     101.0
..          ...     ...  ...                 ...             ...       ...
194          37    male   22                  33              16      26.5
195          75    male   30                  82              71      73.0
196          18    male   39                  85              86      52.0
197         183  female   78                 130              30     106.5
198         129  female   52                  50              75     102.0

     Row_Median  Row_Min  Row_Max  Row_Standard
0          28.5       20       53     11.313708
1          58.5       25       42     12.020815
2          33.0       30       52     11.313708
3          20.5       16       29      9.192388
4         101.0       18      184     96.873629
..          ...      ...      ...           ...
194        26.5       16       37     10.606602
195        73.0       71       75     31.819805
196        52.0       18       39     14.849242
197       106.5       30      183     74.246212
198       102.0       75      129     54.447222

[199 rows x 10 columns]
```

In [31]:
```
df1.groupby(['Genre'])['Age'].mean()
```

Out[31]:
```
Genre
female    50.097087
male      47.635417
Name: Age, dtype: float64
```

In [34]:
```
df_u=df1.rename(columns= {'Annual_income_(k$)':'Income'},inplace=False)
(df_u.groupby(['Genre']).Income.mean())
```

Out[34]:
```
Genre
female    86.184466
male      79.260417
Name: Income, dtype: float64
```

In [ ]:

In [35]:
```
from sklearn import preprocessing
enc = preprocessing.OneHotEncoder()
enc_df = pd.DataFrame(enc.fit_transform(df1[['Genre']]).toarray())
enc_df
```

Out[35]:

| | 0 | 1 |
|---|---|---|
| **0** | 0.0 | 1.0 |
| **1** | 0.0 | 1.0 |
| **2** | 0.0 | 1.0 |
| **3** | 0.0 | 1.0 |
| **4** | 0.0 | 1.0 |
| **...** | ... | ... |
| **194** | 0.0 | 1.0 |
| **195** | 0.0 | 1.0 |
| **196** | 0.0 | 1.0 |
| **197** | 1.0 | 0.0 |
| **198** | 1.0 | 0.0 |

199 rows × 2 columns

In [37]:
```
df_encode =df_u.join(enc_df)
df_encode
```

Out[37]:

| | CustomerID | Genre | Age | Income | Spending_score | Row_Mean | Row_Median | Row_Min | Rov |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 37 | male | 53 | 102 | 20 | 28.5 | 28.5 | 20 | |
| **1** | 25 | male | 42 | 94 | 92 | 58.5 | 58.5 | 25 | |
| **2** | 36 | male | 52 | 124 | 30 | 33.0 | 33.0 | 30 | |
| **3** | 16 | male | 29 | 27 | 25 | 20.5 | 20.5 | 16 | |
| **4** | 184 | male | 47 | 118 | 18 | 101.0 | 101.0 | 18 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | |
| **194** | 37 | male | 22 | 33 | 16 | 26.5 | 26.5 | 16 | |
| **195** | 75 | male | 30 | 82 | 71 | 73.0 | 73.0 | 71 | |
| **196** | 18 | male | 39 | 85 | 86 | 52.0 | 52.0 | 18 | |
| **197** | 183 | female | 78 | 130 | 30 | 106.5 | 106.5 | 30 | |
| **198** | 129 | female | 52 | 50 | 75 | 102.0 | 102.0 | 75 | |

199 rows × 12 columns

In [ ]:

In [38]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from pandas import DataFrame, Series
import seaborn as ans
data = ans.load_dataset("iris")
data
```

Out[38]:

| | sepal_length | sepal_width | petal_length | petal_width | species |
|---|---|---|---|---|---|
| **0** | 5.1 | 3.5 | 1.4 | 0.2 | setosa |
| **1** | 4.9 | 3.0 | 1.4 | 0.2 | setosa |
| **2** | 4.7 | 3.2 | 1.3 | 0.2 | setosa |
| **3** | 4.6 | 3.1 | 1.5 | 0.2 | setosa |
| **4** | 5.0 | 3.6 | 1.4 | 0.2 | setosa |
| **...** | ... | ... | ... | ... | ... |
| **145** | 6.7 | 3.0 | 5.2 | 2.3 | virginica |
| **146** | 6.3 | 2.5 | 5.0 | 1.9 | virginica |
| **147** | 6.5 | 3.0 | 5.2 | 2.0 | virginica |
| **148** | 6.2 | 3.4 | 5.4 | 2.3 | virginica |
| **149** | 5.9 | 3.0 | 5.1 | 1.8 | virginica |

150 rows × 5 columns

In [43]:
```
irisSet = (data['species']== 'Iris-setosa')
print('Iris-setosa')
print(data[irisSet].describe())
```

```
Iris-setosa
       sepal_length  sepal_width  petal_length  petal_width
count           0.0          0.0           0.0          0.0
mean            NaN          NaN           NaN          NaN
std             NaN          NaN           NaN          NaN
min             NaN          NaN           NaN          NaN
25%             NaN          NaN           NaN          NaN
50%             NaN          NaN           NaN          NaN
75%             NaN          NaN           NaN          NaN
max             NaN          NaN           NaN          NaN
```

In [44]:
```
irisVer = (data['species']== 'Iris-versicolor')
```

In [ ]:

45
```
print('Iris-versicolor')
print(data[irisVer].describe())
```

```
Iris-versicolor
       sepal_length  sepal_width  petal_length  petal_width
count           0.0          0.0           0.0          0.0
mean            NaN          NaN           NaN          NaN
std             NaN          NaN           NaN          NaN
min             NaN          NaN           NaN          NaN
25%             NaN          NaN           NaN          NaN
50%             NaN          NaN           NaN          NaN
75%             NaN          NaN           NaN          NaN
max             NaN          NaN           NaN          NaN
```

In [47]:
```
irisVir = (data['species']== 'Iris-virginica')
```

In [48]:
```
print('Iris-virginica')
print(data[irisVir].describe())
```

```
Iris-virginica
       sepal_length  sepal_width  petal_length  petal_width
count           0.0          0.0           0.0          0.0
mean            NaN          NaN           NaN          NaN
std             NaN          NaN           NaN          NaN
min             NaN          NaN           NaN          NaN
25%             NaN          NaN           NaN          NaN
50%             NaN          NaN           NaN          NaN
75%             NaN          NaN           NaN          NaN
max             NaN          NaN           NaN          NaN
```

Name : AvinashKurandale

Roll No : 13223 [TECO-b2]