Assignment no 3

```
Aim:
    1.Summary statistics
    2.Types of Variables
    3.Summary ststistics of income grouped by the age groups
```

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

In [2]:
```
import numpy as np
```

In [3]:
```
df=pd.read_csv("employee.csv")
```

In [4]:
```
df
```

Out[4]:

	customer ID	Gender	Age	Income	Spending Score
0	1	Male	43	33761	60
1	2	Female	32	24628	65
2	3	Female	20	26349	54
3	4	Male	59	20385	28
4	5	Female	43	32093	86
...	...	...	...	...	...
195	196	Male	45	27769	100
196	197	Female	37	32039	71
197	198	Female	44	26259	100
198	199	Female	23	22732	83
199	200	Male	49	28315	26

200 rows × 5 columns

1.Measures of Dispersion

In [6]:
```
df.mean()
```

```
C:\Users\Welcome\AppData\Local\Temp\ipykernel_2300\3698961737.py:1: FutureWar
ning: The default value of numeric_only in DataFrame.mean is deprecated. In a
future version, it will default to False. In addition, specifying 'numeric_on
ly=None' is deprecated. Select only valid columns or specify the value of num
eric_only to silence this warning.
  df.mean()
```

Out[6]:
```
customer ID         100.500
Age                  40.090
Income            30235.055
Spending Score       53.600
dtype: float64
```

In [8]:
```
df.loc[:,'Age'].mean()
```

Out[8]:
```
40.09
```

In [9]:
```
df.mean(axis=1)[0:4]
```

```
C:\Users\Welcome\AppData\Local\Temp\ipykernel_2300\1148177455.py:1: FutureWar
ning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_onl
y=None') is deprecated; in a future version this will raise TypeError.  Selec
t only valid columns before calling the reduction.
  df.mean(axis=1)[0:4]
```

Out[9]:
```
0    8466.25
1    6181.75
2    6606.50
3    5119.00
dtype: float64
```

In [10]:
```
df.median()
```

```
C:\Users\Welcome\AppData\Local\Temp\ipykernel_2300\530051474.py:1: FutureWarn
ing: The default value of numeric_only in DataFrame.median is deprecated. In
a future version, it will default to False. In addition, specifying 'numeric_
only=None' is deprecated. Select only valid columns or specify the value of n
umeric_only to silence this warning.
  df.median()
```

Out[10]:
```
customer ID         100.5
Age                  40.5
Income            30839.5
Spending Score       58.5
dtype: float64
```

In [11]:
```
df.loc[:,'Age'].median()
```

Out[11]:
```
40.5
```

In [12]:
```
df.median(axis=1)[0:4]
```

```
C:\Users\Welcome\AppData\Local\Temp\ipykernel_2300\381455229.py:1: FutureWarn
ing: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only
=None') is deprecated; in a future version this will raise TypeError.  Select
only valid columns before calling the reduction.
  df.median(axis=1)[0:4]
```

Out[12]:
```
0    51.5
1    48.5
2    37.0
3    43.5
dtype: float64
```

In [13]:
```
df.mode()
```

Out[13]:

	customer ID	Gender	Age	Income	Spending Score
0	1	Male	23.0	36017.0	82.0
1	2	NaN	30.0	NaN	NaN
2	3	NaN	55.0	NaN	NaN
3	4	NaN	NaN	NaN	NaN
4	5	NaN	NaN	NaN	NaN
...	...	...	...	...	...
195	196	NaN	NaN	NaN	NaN
196	197	NaN	NaN	NaN	NaN
197	198	NaN	NaN	NaN	NaN
198	199	NaN	NaN	NaN	NaN
199	200	NaN	NaN	NaN	NaN

200 rows × 5 columns

In [14]:
```
df.loc[:,'Age'].mode()
```

Out[14]:
```
0    23
1    30
2    55
Name: Age, dtype: int64
```

In [15]:
```
df.min()
```

Out[15]:
```
customer ID           1
Gender           Female
Age                  20
Income            20069
Spending Score        1
dtype: object
```

In [16]:
```
df.loc[:,'Age'].min(skipna=False)
```

Out[16]:
```
20
```

In [17]:
```
df.max()
```

Out[17]:
```
customer ID         200
Gender             Male
Age                  60
Income            39926
Spending Score      100
dtype: object
```

In [18]:
```
df.loc[:,'Age'].max(skipna=False)
```

Out[18]:
```
60
```

In [19]:
```
df.std()
```

```
C:\Users\Welcome\AppData\Local\Temp\ipykernel_2300\3390915376.py:1: FutureWar
ning: The default value of numeric_only in DataFrame.std is deprecated. In a
future version, it will default to False. In addition, specifying 'numeric_on
ly=None' is deprecated. Select only valid columns or specify the value of num
eric_only to silence this warning.
  df.std()
```

Out[19]:
```
customer ID         57.879185
Age                 12.165604
Income            5885.749609
Spending Score      30.433881
dtype: float64
```

In [20]:
```
df.loc[:,'Age'].std()
```

Out[20]:
```
12.165603542271901
```

In [21]:
```
df.std(axis=1)[0:4]
```

```
C:\Users\Welcome\AppData\Local\Temp\ipykernel_2300\3966588610.py:1: FutureWar
ning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_onl
y=None') is deprecated; in a future version this will raise TypeError.  Selec
t only valid columns before calling the reduction.
  df.std(axis=1)[0:4]
```

Out[21]:
```
0    16863.184898
1    12297.526916
2    13161.683745
3    10177.358236
dtype: float64
```

In [22]:
```
df.groupby(['Gender'])['Age'].mean()
```

Out[22]:
```
Gender
Female    39.494505
Male      40.587156
Name: Age, dtype: float64
```

In [24]:
```
df_u=df.rename(columns={'Income':'Annual_Income'},inplace=False)
```

In [25]:
```
df_u.groupby(['Gender']).Annual_Income.mean()
```

Out[25]:
```
Gender
Female    30156.439560
Male      30300.688073
Name: Annual_Income, dtype: float64
```

In [26]:
```
from sklearn import preprocessing
enc=preprocessing.OneHotEncoder()
enc_df=pd.DataFrame(enc.fit_transform(df[['Gender']]).toarray())
enc_df
```

Out[26]:

	0	1
0	0.0	1.0
1	1.0	0.0
2	1.0	0.0
3	0.0	1.0
4	1.0	0.0
...	...	...
195	0.0	1.0
196	1.0	0.0
197	1.0	0.0
198	1.0	0.0
199	0.0	1.0

200 rows × 2 columns

In [27]:
```
df_encode=df_u.join(enc_df)
```

In [28]:
```
df_encode
```

Out[28]:

	customer ID	Gender	Age	Annual_Income	Spending Score	0	1
0	1	Male	43	33761	60	0.0	1.0
1	2	Female	32	24628	65	1.0	0.0
2	3	Female	20	26349	54	1.0	0.0
3	4	Male	59	20385	28	0.0	1.0
4	5	Female	43	32093	86	1.0	0.0
...	...	...	...	...	...	...	...
195	196	Male	45	27769	100	0.0	1.0
196	197	Female	37	32039	71	1.0	0.0
197	198	Female	44	26259	100	1.0	0.0
198	199	Female	23	22732	83	1.0	0.0
199	200	Male	49	28315	26	0.0	1.0

200 rows × 7 columns

NAME: NEHA JADHAV ROLL NO: 13247