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

In [2]:
```python
df=pd.read_csv("/home/rmdstic/Documents/TEA-14/Mall_Customers.csv")
```

In [3]: df

Out[3]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40
5	6	Female	22	17	76
6	7	Female	35	18	6
7	8	Female	23	18	94
8	9	Male	64	19	3
9	10	Female	30	19	72
10	11	Male	67	19	14
11	12	Female	35	19	99
12	13	Female	58	20	15
13	14	Female	24	20	77
14	15	Male	37	20	13
15	16	Male	22	20	79
16	17	Female	35	21	35
17	18	Male	20	21	66
18	19	Male	52	23	29
19	20	Female	35	23	98
20	21	Male	35	24	35
21	22	Male	25	24	73
22	23	Female	46	25	5
23	24	Male	31	25	73
24	25	Female	54	28	14
25	26	Male	29	28	82
26	27	Female	45	28	32
27	28	Male	35	28	61
28	29	Female	40	29	31
29	30	Female	23	29	87
...	...	...	...	...	...
170	171	Male	40	87	13
171	172	Male	28	87	75
172	173	Male	36	87	10
173	174	Male	36	87	92

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
174	175	Female	52	88	13
175	176	Female	30	88	86
176	177	Male	58	88	15
177	178	Male	27	88	69
178	179	Male	59	93	14
179	180	Male	35	93	90
180	181	Female	37	97	32
181	182	Female	32	97	86
182	183	Male	46	98	15
183	184	Female	29	98	88
184	185	Female	41	99	39
185	186	Male	30	99	97
186	187	Female	54	101	24
187	188	Male	28	101	68
188	189	Female	41	103	17
189	190	Female	36	103	85
190	191	Female	34	103	23
191	192	Female	32	103	69
192	193	Male	33	113	8
193	194	Female	38	113	91
194	195	Female	47	120	16
195	196	Female	35	120	79
196	197	Female	45	126	28
197	198	Male	32	126	74
198	199	Male	32	137	18
199	200	Male	30	137	83

200 rows × 5 columns

```
In [4]: df.mean()
```

```
Out[4]: CustomerID              100.50
        Age                      38.85
        Annual Income (k$)       60.56
        Spending Score (1-100)   50.20
        dtype: float64
```

In [5]: `df.median()`

Out[5]:
```
CustomerID               100.5
Age                       36.0
Annual Income (k$)        61.5
Spending Score (1-100)    50.0
dtype: float64
```

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

Out[6]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Female	32.0	54.0	42.0
1	2	NaN	NaN	78.0	NaN
2	3	NaN	NaN	NaN	NaN
3	4	NaN	NaN	NaN	NaN
4	5	NaN	NaN	NaN	NaN
5	6	NaN	NaN	NaN	NaN
6	7	NaN	NaN	NaN	NaN
7	8	NaN	NaN	NaN	NaN
8	9	NaN	NaN	NaN	NaN
9	10	NaN	NaN	NaN	NaN
10	11	NaN	NaN	NaN	NaN
11	12	NaN	NaN	NaN	NaN
12	13	NaN	NaN	NaN	NaN
13	14	NaN	NaN	NaN	NaN
14	15	NaN	NaN	NaN	NaN
15	16	NaN	NaN	NaN	NaN
16	17	NaN	NaN	NaN	NaN
17	18	NaN	NaN	NaN	NaN
18	19	NaN	NaN	NaN	NaN
19	20	NaN	NaN	NaN	NaN
20	21	NaN	NaN	NaN	NaN
21	22	NaN	NaN	NaN	NaN
22	23	NaN	NaN	NaN	NaN
23	24	NaN	NaN	NaN	NaN
24	25	NaN	NaN	NaN	NaN
25	26	NaN	NaN	NaN	NaN
26	27	NaN	NaN	NaN	NaN
27	28	NaN	NaN	NaN	NaN
28	29	NaN	NaN	NaN	NaN
29	30	NaN	NaN	NaN	NaN
...	...	...	...	...	...
170	171	NaN	NaN	NaN	NaN
171	172	NaN	NaN	NaN	NaN
172	173	NaN	NaN	NaN	NaN
173	174	NaN	NaN	NaN	NaN

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
174	175	NaN	NaN	NaN	NaN
175	176	NaN	NaN	NaN	NaN
176	177	NaN	NaN	NaN	NaN
177	178	NaN	NaN	NaN	NaN
178	179	NaN	NaN	NaN	NaN
179	180	NaN	NaN	NaN	NaN
180	181	NaN	NaN	NaN	NaN
181	182	NaN	NaN	NaN	NaN
182	183	NaN	NaN	NaN	NaN
183	184	NaN	NaN	NaN	NaN
184	185	NaN	NaN	NaN	NaN
185	186	NaN	NaN	NaN	NaN
186	187	NaN	NaN	NaN	NaN
187	188	NaN	NaN	NaN	NaN
188	189	NaN	NaN	NaN	NaN
189	190	NaN	NaN	NaN	NaN
190	191	NaN	NaN	NaN	NaN
191	192	NaN	NaN	NaN	NaN
192	193	NaN	NaN	NaN	NaN
193	194	NaN	NaN	NaN	NaN
194	195	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 [7]: df.max()
```

```
Out[7]: CustomerID                200
        Gender                   Male
        Age                        70
        Annual Income (k$)        137
        Spending Score (1-100)     99
        dtype: object
```

In [8]:
```python
df.std()
```

Out[8]:
```
CustomerID              57.879185
Age                     13.969007
Annual Income (k$)      26.264721
Spending Score (1-100)  25.823522
dtype: float64
```

In [9]:
```python
df.groupby(["Gender"])["Age"].mean()
```

Out[9]:
```
Gender
Female    38.098214
Male      39.806818
Name: Age, dtype: float64
```

In [10]:
```python
df.groupby(["Gender"])["Annual Income (k$)"].mean()
```

Out[10]:
```
Gender
Female    59.250000
Male      62.227273
Name: Annual Income (k$), dtype: float64
```

In [11]:
```python
df.groupby(["Gender"]).mean()
```

Out[11]:

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
Gender				
Female	97.562500	38.098214	59.250000	51.526786
Male	104.238636	39.806818	62.227273	48.511364

In [12]:
```python
df.groupby(["Gender"])["Age"].median()
```

Out[12]:
```
Gender
Female    35
Male      37
Name: Age, dtype: int64
```

In [13]:
```python
x=df.drop(axis=1,columns=["Gender"])
```

In [14]: x

Out[14]:

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
0	1	19	15	39
1	2	21	15	81
2	3	20	16	6
3	4	23	16	77
4	5	31	17	40
5	6	22	17	76
6	7	35	18	6
7	8	23	18	94
8	9	64	19	3
9	10	30	19	72
10	11	67	19	14
11	12	35	19	99
12	13	58	20	15
13	14	24	20	77
14	15	37	20	13
15	16	22	20	79
16	17	35	21	35
17	18	20	21	66
18	19	52	23	29
19	20	35	23	98
20	21	35	24	35
21	22	25	24	73
22	23	46	25	5
23	24	31	25	73
24	25	54	28	14
25	26	29	28	82
26	27	45	28	32
27	28	35	28	61
28	29	40	29	31
29	30	23	29	87
...	...	...	...	...
170	171	40	87	13
171	172	28	87	75
172	173	36	87	10
173	174	36	87	92

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
174	175	52	88	13
175	176	30	88	86
176	177	58	88	15
177	178	27	88	69
178	179	59	93	14
179	180	35	93	90
180	181	37	97	32
181	182	32	97	86
182	183	46	98	15
183	184	29	98	88
184	185	41	99	39
185	186	30	99	97
186	187	54	101	24
187	188	28	101	68
188	189	41	103	17
189	190	36	103	85
190	191	34	103	23
191	192	32	103	69
192	193	33	113	8
193	194	38	113	91
194	195	47	120	16
195	196	35	120	79
196	197	45	126	28
197	198	32	126	74
198	199	32	137	18
199	200	30	137	83

200 rows × 4 columns

In [15]:
```python
from sklearn import preprocessing
enc = preprocessing.OneHotEncoder()
enc_df=(enc.fit_transform(df[['Gender']]).toarray())
enc_df
y=pd.DataFrame(enc_df)
df_encode=x.join(y)
df_encode
```

Out[15]:

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)	0	1
0	1	19	15	39	0.0	1.0
1	2	21	15	81	0.0	1.0
2	3	20	16	6	1.0	0.0
3	4	23	16	77	1.0	0.0
4	5	31	17	40	1.0	0.0
5	6	22	17	76	1.0	0.0
6	7	35	18	6	1.0	0.0
7	8	23	18	94	1.0	0.0
8	9	64	19	3	0.0	1.0
9	10	30	19	72	1.0	0.0
10	11	67	19	14	0.0	1.0
11	12	35	19	99	1.0	0.0
12	13	58	20	15	1.0	0.0
13	14	24	20	77	1.0	0.0
14	15	37	20	13	0.0	1.0
15	16	22	20	79	0.0	1.0
16	17	35	21	35	1.0	0.0
17	18	20	21	66	0.0	1.0
18	19	52	23	29	0.0	1.0
19	20	35	23	98	1.0	0.0
20	21	35	24	35	0.0	1.0
21	22	25	24	73	0.0	1.0
22	23	46	25	5	1.0	0.0
23	24	31	25	73	0.0	1.0
24	25	54	28	14	1.0	0.0
25	26	29	28	82	0.0	1.0
26	27	45	28	32	1.0	0.0
27	28	35	28	61	0.0	1.0
28	29	40	29	31	1.0	0.0
29	30	23	29	87	1.0	0.0
...	...	...	...	...	...	...

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)	0	1
170	171	40	87	13	0.0	1.0
171	172	28	87	75	0.0	1.0
172	173	36	87	10	0.0	1.0
173	174	36	87	92	0.0	1.0
174	175	52	88	13	1.0	0.0
175	176	30	88	86	1.0	0.0
176	177	58	88	15	0.0	1.0
177	178	27	88	69	0.0	1.0
178	179	59	93	14	0.0	1.0
179	180	35	93	90	0.0	1.0
180	181	37	97	32	1.0	0.0
181	182	32	97	86	1.0	0.0
182	183	46	98	15	0.0	1.0
183	184	29	98	88	1.0	0.0
184	185	41	99	39	1.0	0.0
185	186	30	99	97	0.0	1.0
186	187	54	101	24	1.0	0.0
187	188	28	101	68	0.0	1.0
188	189	41	103	17	1.0	0.0
189	190	36	103	85	1.0	0.0
190	191	34	103	23	1.0	0.0
191	192	32	103	69	1.0	0.0
192	193	33	113	8	0.0	1.0
193	194	38	113	91	1.0	0.0
194	195	47	120	16	1.0	0.0
195	196	35	120	79	1.0	0.0
196	197	45	126	28	1.0	0.0
197	198	32	126	74	0.0	1.0
198	199	32	137	18	0.0	1.0
199	200	30	137	83	0.0	1.0

200 rows × 6 columns

```
In [18]:  df=pd.read_csv("/home/rmdstic/Documents/TEA-14/iris.csv")
```

In [19]: df

Out[19]:

	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
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa
10	5.4	3.7	1.5	0.2	setosa
11	4.8	3.4	1.6	0.2	setosa
12	4.8	3.0	1.4	0.1	setosa
13	4.3	3.0	1.1	0.1	setosa
14	5.8	4.0	1.2	0.2	setosa
15	5.7	4.4	1.5	0.4	setosa
16	5.4	3.9	1.3	0.4	setosa
17	5.1	3.5	1.4	0.3	setosa
18	5.7	3.8	1.7	0.3	setosa
19	5.1	3.8	1.5	0.3	setosa
20	5.4	3.4	1.7	0.2	setosa
21	5.1	3.7	1.5	0.4	setosa
22	4.6	3.6	1.0	0.2	setosa
23	5.1	3.3	1.7	0.5	setosa
24	4.8	3.4	1.9	0.2	setosa
25	5.0	3.0	1.6	0.2	setosa
26	5.0	3.4	1.6	0.4	setosa
27	5.2	3.5	1.5	0.2	setosa
28	5.2	3.4	1.4	0.2	setosa
29	4.7	3.2	1.6	0.2	setosa
...	...	...	...	...	...
120	6.9	3.2	5.7	2.3	virginica
121	5.6	2.8	4.9	2.0	virginica
122	7.7	2.8	6.7	2.0	virginica
123	6.3	2.7	4.9	1.8	virginica

	sepal_length	sepal_width	petal_length	petal_width	species
124	6.7	3.3	5.7	2.1	virginica
125	7.2	3.2	6.0	1.8	virginica
126	6.2	2.8	4.8	1.8	virginica
127	6.1	3.0	4.9	1.8	virginica
128	6.4	2.8	5.6	2.1	virginica
129	7.2	3.0	5.8	1.6	virginica
130	7.4	2.8	6.1	1.9	virginica
131	7.9	3.8	6.4	2.0	virginica
132	6.4	2.8	5.6	2.2	virginica
133	6.3	2.8	5.1	1.5	virginica
134	6.1	2.6	5.6	1.4	virginica
135	7.7	3.0	6.1	2.3	virginica
136	6.3	3.4	5.6	2.4	virginica
137	6.4	3.1	5.5	1.8	virginica
138	6.0	3.0	4.8	1.8	virginica
139	6.9	3.1	5.4	2.1	virginica
140	6.7	3.1	5.6	2.4	virginica
141	6.9	3.1	5.1	2.3	virginica
142	5.8	2.7	5.1	1.9	virginica
143	6.8	3.2	5.9	2.3	virginica
144	6.7	3.3	5.7	2.5	virginica
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 [21]:
```python
set1=(df['species']=='versicolor')
```

In [22]:
```python
print(df[set1].describe())
```

```
       sepal_length  sepal_width  petal_length  petal_width
count     50.000000    50.000000     50.000000    50.000000
mean       5.936000     2.770000      4.260000     1.326000
std        0.516171     0.313798      0.469911     0.197753
min        4.900000     2.000000      3.000000     1.000000
25%        5.600000     2.525000      4.000000     1.200000
50%        5.900000     2.800000      4.350000     1.300000
75%        6.300000     3.000000      4.600000     1.500000
max        7.000000     3.400000      5.100000     1.800000
```

```
In [24]: set1=(df['species']=='versicolor')
```

```
In [25]: print(df[set1].describe())
```

```
       sepal_length  sepal_width  petal_length  petal_width
count     50.000000    50.000000     50.000000    50.000000
mean       5.936000     2.770000      4.260000     1.326000
std        0.516171     0.313798      0.469911     0.197753
min        4.900000     2.000000      3.000000     1.000000
25%        5.600000     2.525000      4.000000     1.200000
50%        5.900000     2.800000      4.350000     1.300000
75%        6.300000     3.000000      4.600000     1.500000
max        7.000000     3.400000      5.100000     1.800000
```

```
In [26]: set1=(df['species']=='versicolor')
```

```
In [28]: print(df[set1].describe())
```

```
       sepal_length  sepal_width  petal_length  petal_width
count     50.000000    50.000000     50.000000    50.000000
mean       5.936000     2.770000      4.260000     1.326000
std        0.516171     0.313798      0.469911     0.197753
min        4.900000     2.000000      3.000000     1.000000
25%        5.600000     2.525000      4.000000     1.200000
50%        5.900000     2.800000      4.350000     1.300000
75%        6.300000     3.000000      4.600000     1.500000
max        7.000000     3.400000      5.100000     1.800000
```

```
In [29]: df['species'].unique()
```

```
Out[29]: array(['setosa', 'versicolor', 'virginica'], dtype=object)
```

```
In [31]: set2=(df['species']=='setosa')
```

```
In [33]: print(df[set2].describe())
```

```
       sepal_length  sepal_width  petal_length  petal_width
count      50.00000    50.000000     50.000000     50.00000
mean        5.00600     3.418000      1.464000      0.24400
std         0.35249     0.381024      0.173511      0.10721
min         4.30000     2.300000      1.000000      0.10000
25%         4.80000     3.125000      1.400000      0.20000
50%         5.00000     3.400000      1.500000      0.20000
75%         5.20000     3.675000      1.575000      0.30000
max         5.80000     4.400000      1.900000      0.60000
```

```
In [37]: print("df-Setosa")
         print(df[set2].describe())
         print("df-Versicolor")
         print(df[set2].describe())
         print("df-Virginica")
         print(df[set2].describe())
```

```
df-Setosa
       sepal_length  sepal_width  petal_length  petal_width
count     50.00000    50.000000     50.000000     50.00000
mean       5.00600     3.418000      1.464000      0.24400
std        0.35249     0.381024      0.173511      0.10721
min        4.30000     2.300000      1.000000      0.10000
25%        4.80000     3.125000      1.400000      0.20000
50%        5.00000     3.400000      1.500000      0.20000
75%        5.20000     3.675000      1.575000      0.30000
max        5.80000     4.400000      1.900000      0.60000
df-Versicolor
       sepal_length  sepal_width  petal_length  petal_width
count     50.00000    50.000000     50.000000     50.00000
mean       5.00600     3.418000      1.464000      0.24400
std        0.35249     0.381024      0.173511      0.10721
min        4.30000     2.300000      1.000000      0.10000
25%        4.80000     3.125000      1.400000      0.20000
50%        5.00000     3.400000      1.500000      0.20000
75%        5.20000     3.675000      1.575000      0.30000
max        5.80000     4.400000      1.900000      0.60000
df-Virginica
       sepal_length  sepal_width  petal_length  petal_width
count     50.00000    50.000000     50.000000     50.00000
mean       5.00600     3.418000      1.464000      0.24400
std        0.35249     0.381024      0.173511      0.10721
min        4.30000     2.300000      1.000000      0.10000
25%        4.80000     3.125000      1.400000      0.20000
50%        5.00000     3.400000      1.500000      0.20000
75%        5.20000     3.675000      1.575000      0.30000
max        5.80000     4.400000      1.900000      0.60000
```

```
In [ ]:
```