In [1]:

```python
#11). Repeat b), c), d), e) in Q2, by using Python on the HW1.csv data set.

#Import Python Libraries
import numpy as np
import scipy as sp
import pandas as pd
import matplotlib as mpl
from IPython.display import display, HTML
```

In [2]:

```python
df=pd.read_csv('HW1.csv')
```

In [3]:

```python
df.head()
```

Out[3]:

	Cust ID	Gender	Income	Age	Rentals	Avg Per Visit	Incidentals	Genre
0	1	M	45000	25	27	2.5	Yes	Action
1	2	F	54000	33	12	3.4	No	Drama
2	3	F	32000	20	42	1.6	No	Comedy
3	4	F	59000	70	16	4.2	Yes	Drama
4	5	M	37000	35	25	3.2	Yes	Action

In [4]:

```python
#b)Use min-max normalization to transform the values of the Income attribute onto the range [1-5].

# Normalization method  Min-max normalization [1,5]
df["Income_Min-max"]=((df["Income"]-df["Income"].min())/(df["Income"].max()-df["Income"].min()))*(4)+1
display(HTML(df.head(50).to_html()))
```

	Cust ID	Gender	Income	Age	Rentals	Avg Per Visit	Incidentals	Genre	Income_Min-max
0	1	M	45000	25	27	2.5	Yes	Action	3.000000
1	2	F	54000	33	12	3.4	No	Drama	3.409091
2	3	F	32000	20	42	1.6	No	Comedy	2.409091
3	4	F	59000	70	16	4.2	Yes	Drama	3.636364
4	5	M	37000	35	25	3.2	Yes	Action	2.636364
5	6	M	18000	20	33	1.7	No	Action	1.772727
6	7	F	29000	45	19	3.8	No	Drama	2.272727
7	8	M	74000	25	31	2.4	Yes	Action	4.318182
8	9	M	38000	21	18	2.1	No	Comedy	2.681818
9	10	F	65000	40	21	3.3	No	Drama	3.909091
10	11	F	41000	22	48	2.3	Yes	Drama	2.818182
11	12	F	26000	22	29	2.9	Yes	Action	2.136364
12	13	M	83000	46	14	3.6	No	Comedy	4.727273
13	14	M	45000	36	24	2.7	No	Drama	3.000000
14	15	M	68000	30	36	2.7	Yes	Comedy	4.045455
15	16	M	17000	19	26	2.2	Yes	Action	1.727273
16	17	M	36000	35	28	3.5	Yes	Drama	2.590909
17	18	F	6000	16	39	1.8	Yes	Action	1.227273
18	19	F	24000	25	41	3.1	No	Comedy	2.045455
19	20	M	12000	16	23	2.2	Yes	Action	1.500000
20	21	F	47000	52	11	3.1	No	Drama	3.090909
21	22	M	25000	33	16	2.9	Yes	Drama	2.090909
22	23	F	2000	15	30	2.5	No	Comedy	1.045455
23	24	F	79000	35	22	3.8	Yes	Drama	4.545455
24	25	M	1000	16	25	1.4	Yes	Comedy	1.000000
25	26	F	56000	35	40	2.6	Yes	Action	3.500000
26	27	F	62000	47	32	3.6	No	Drama	3.772727
27	28	M	57000	52	22	4.1	No	Comedy	3.545455
28	29	F	15000	18	37	2.1	Yes	Action	1.636364
29	30	M	41000	25	17	1.4	Yes	Action	2.818182
30	31	F	49000	56	15	3.2	No	Comedy	3.181818
31	32	M	47000	30	21	3.1	Yes	Drama	3.090909
32	33	M	23000	25	28	2.7	No	Action	2.000000
33	34	F	29000	32	19	2.9	Yes	Action	2.272727
34	35	M	74000	29	43	4.6	Yes	Action	4.318182
35	36	F	29000	21	34	2.3	No	Comedy	2.272727
36	37	M	89000	46	12	1.2	No	Comedy	5.000000
37	38	M	41000	38	20	3.3	Yes	Drama	2.818182
38	39	F	68000	35	19	3.9	No	Comedy	4.045455
39	40	M	17000	19	32	1.8	No	Action	1.727273
40	41	F	50000	33	17	1.4	No	Drama	3.227273
41	42	M	32000	25	26	2.2	Yes	Action	2.409091
42	43	F	49000	28	48	3.3	Yes	Drama	3.181818
43	44	M	35000	24	24	1.7	No	Drama	2.545455
44	45	M	56000	38	30	3.5	Yes	Drama	3.500000
45	46	F	57000	43	9	1.1	No	Drama	3.545455
46	47	F	69000	35	22	2.8	Yes	Drama	4.090909
47	48	F	52000	47	14	1.6	No	Drama	3.318182
48	49	M	31000	25	42	3.4	Yes	Action	2.363636
49	50	M	24000	20	33	4.7	No	Action	2.045455

In [5]:

```python
#c)Use z-score normalization to standardize the values of the age attribute.

from scipy.stats import zscore
df[['Age_z-score']] = zscore(df[['Age']])
display(HTML(df.head(50).to_html()))
```

	Cust ID	Gender	Income	Age	Rentals	Avg Per Visit	Incidentals	Genre	Income_Min-max	Age_z-score
0	1	M	45000	25	27	2.5	Yes	Action	3.000000	-0.552204
1	2	F	54000	33	12	3.4	No	Drama	3.409091	0.121216
2	3	F	32000	20	42	1.6	No	Comedy	2.409091	-0.973092
3	4	F	59000	70	16	4.2	Yes	Drama	3.636364	3.235782
4	5	M	37000	35	25	3.2	Yes	Action	2.636364	0.289570
5	6	M	18000	20	33	1.7	No	Action	1.772727	-0.973092
6	7	F	29000	45	19	3.8	No	Drama	2.272727	1.131345
7	8	M	74000	25	31	2.4	Yes	Action	4.318182	-0.552204
8	9	M	38000	21	18	2.1	No	Comedy	2.681818	-0.888914
9	10	F	65000	40	21	3.3	No	Drama	3.909091	0.710458
10	11	F	41000	22	48	2.3	Yes	Drama	2.818182	-0.804737
11	12	F	26000	22	29	2.9	Yes	Action	2.136364	-0.804737
12	13	M	83000	46	14	3.6	No	Comedy	4.727273	1.215523
13	14	M	45000	36	24	2.7	No	Drama	3.000000	0.373748
14	15	M	68000	30	36	2.7	Yes	Comedy	4.045455	-0.131317
15	16	M	17000	19	26	2.2	Yes	Action	1.727273	-1.057269
16	17	M	36000	35	28	3.5	Yes	Drama	2.590909	0.289570
17	18	F	6000	16	39	1.8	Yes	Action	1.227273	-1.309801
18	19	F	24000	25	41	3.1	No	Comedy	2.045455	-0.552204
19	20	M	12000	16	23	2.2	Yes	Action	1.500000	-1.309801
20	21	F	47000	52	11	3.1	No	Drama	3.090909	1.720587
21	22	M	25000	33	16	2.9	Yes	Drama	2.090909	0.121216
22	23	F	2000	15	30	2.5	No	Comedy	1.045455	-1.393979
23	24	F	79000	35	22	3.8	Yes	Drama	4.545455	0.289570
24	25	M	1000	16	25	1.4	Yes	Comedy	1.000000	-1.309801
25	26	F	56000	35	40	2.6	Yes	Action	3.500000	0.289570
26	27	F	62000	47	32	3.6	No	Drama	3.772727	1.299700
27	28	M	57000	52	22	4.1	No	Comedy	3.545455	1.720587
28	29	F	15000	18	37	2.1	Yes	Action	1.636364	-1.141446
29	30	M	41000	25	17	1.4	Yes	Action	2.818182	-0.552204
30	31	F	49000	56	15	3.2	No	Comedy	3.181818	2.057297
31	32	M	47000	30	21	3.1	Yes	Drama	3.090909	-0.131317
32	33	M	23000	25	28	2.7	No	Action	2.000000	-0.552204
33	34	F	29000	32	19	2.9	Yes	Action	2.272727	0.037038
34	35	M	74000	29	43	4.6	Yes	Action	4.318182	-0.215494
35	36	F	29000	21	34	2.3	No	Comedy	2.272727	-0.888914
36	37	M	89000	46	12	1.2	No	Comedy	5.000000	1.215523
37	38	M	41000	38	20	3.3	Yes	Drama	2.818182	0.542103
38	39	F	68000	35	19	3.9	No	Comedy	4.045455	0.289570
39	40	M	17000	19	32	1.8	No	Action	1.727273	-1.057269
40	41	F	50000	33	17	1.4	No	Drama	3.227273	0.121216
41	42	M	32000	25	26	2.2	Yes	Action	2.409091	-0.552204
42	43	F	49000	28	48	3.3	Yes	Drama	3.181818	-0.299672
43	44	M	35000	24	24	1.7	No	Drama	2.545455	-0.636382
44	45	M	56000	38	30	3.5	Yes	Drama	3.500000	0.542103
45	46	F	57000	43	9	1.1	No	Drama	3.545455	0.962990
46	47	F	69000	35	22	2.8	Yes	Drama	4.090909	0.289570
47	48	F	52000	47	14	1.6	No	Drama	3.318182	1.299700
48	49	M	31000	25	42	3.4	Yes	Action	2.363636	-0.552204
49	50	M	24000	20	33	4.7	No	Action	2.045455	-0.973092

In [6]:

```python
# Define the bins and labels
bins = [1, 21, 41, float('inf')]
labels = ['Young', 'MidAge', 'Old']
# Creating a new column called 'Age_Category' for discretized values
df['Age_discretization'] = pd.cut(df['Age'], bins=bins, labels=labels, right=False)
display(HTML(df.head(50).to_html()))
```

	Cust ID	Gender	Income	Age	Rentals	Avg Per Visit	Incidentals	Genre	Income_Min-max	Age_z-score	Age_discretization
0	1	M	45000	25	27	2.5	Yes	Action	3.000000	-0.552204	MidAge
1	2	F	54000	33	12	3.4	No	Drama	3.409091	0.121216	MidAge
2	3	F	32000	20	42	1.6	No	Comedy	2.409091	-0.973092	Young
3	4	F	59000	70	16	4.2	Yes	Drama	3.636364	3.235782	Old
4	5	M	37000	35	25	3.2	Yes	Action	2.636364	0.289570	MidAge
5	6	M	18000	20	33	1.7	No	Action	1.772727	-0.973092	Young
6	7	F	29000	45	19	3.8	No	Drama	2.272727	1.131345	Old
7	8	M	74000	25	31	2.4	Yes	Action	4.318182	-0.552204	MidAge
8	9	M	38000	21	18	2.1	No	Comedy	2.681818	-0.888914	MidAge
9	10	F	65000	40	21	3.3	No	Drama	3.909091	0.710458	MidAge
10	11	F	41000	22	48	2.3	Yes	Drama	2.818182	-0.804737	MidAge
11	12	F	26000	22	29	2.9	Yes	Action	2.136364	-0.804737	MidAge
12	13	M	83000	46	14	3.6	No	Comedy	4.727273	1.215523	Old
13	14	M	45000	36	24	2.7	No	Drama	3.000000	0.373748	MidAge
14	15	M	68000	30	36	2.7	Yes	Comedy	4.045455	-0.131317	MidAge
15	16	M	17000	19	26	2.2	Yes	Action	1.727273	-1.057269	Young
16	17	M	36000	35	28	3.5	Yes	Drama	2.590909	0.289570	MidAge
17	18	F	6000	16	39	1.8	Yes	Action	1.227273	-1.309801	Young
18	19	F	24000	25	41	3.1	No	Comedy	2.045455	-0.552204	MidAge
19	20	M	12000	16	23	2.2	Yes	Action	1.500000	-1.309801	Young
20	21	F	47000	52	11	3.1	No	Drama	3.090909	1.720587	Old
21	22	M	25000	33	16	2.9	Yes	Drama	2.090909	0.121216	MidAge
22	23	F	2000	15	30	2.5	No	Comedy	1.045455	-1.393979	Young
23	24	F	79000	35	22	3.8	Yes	Drama	4.545455	0.289570	MidAge
24	25	M	1000	16	25	1.4	Yes	Comedy	1.000000	-1.309801	Young
25	26	F	56000	35	40	2.6	Yes	Action	3.500000	0.289570	MidAge
26	27	F	62000	47	32	3.6	No	Drama	3.772727	1.299700	Old
27	28	M	57000	52	22	4.1	No	Comedy	3.545455	1.720587	Old
28	29	F	15000	18	37	2.1	Yes	Action	1.636364	-1.141446	Young
29	30	M	41000	25	17	1.4	Yes	Action	2.818182	-0.552204	MidAge
30	31	F	49000	56	15	3.2	No	Comedy	3.181818	2.057297	Old
31	32	M	47000	30	21	3.1	Yes	Drama	3.090909	-0.131317	MidAge
32	33	M	23000	25	28	2.7	No	Action	2.000000	-0.552204	MidAge
33	34	F	29000	32	19	2.9	Yes	Action	2.272727	0.037038	MidAge
34	35	M	74000	29	43	4.6	Yes	Action	4.318182	-0.215494	MidAge
35	36	F	29000	21	34	2.3	No	Comedy	2.272727	-0.888914	MidAge
36	37	M	89000	46	12	1.2	No	Comedy	5.000000	1.215523	Old
37	38	M	41000	38	20	3.3	Yes	Drama	2.818182	0.542103	MidAge
38	39	F	68000	35	19	3.9	No	Comedy	4.045455	0.289570	MidAge
39	40	M	17000	19	32	1.8	No	Action	1.727273	-1.057269	Young
40	41	F	50000	33	17	1.4	No	Drama	3.227273	0.121216	MidAge
41	42	M	32000	25	26	2.2	Yes	Action	2.409091	-0.552204	MidAge
42	43	F	49000	28	48	3.3	Yes	Drama	3.181818	-0.299672	MidAge
43	44	M	35000	24	24	1.7	No	Drama	2.545455	-0.636382	MidAge
44	45	M	56000	38	30	3.5	Yes	Drama	3.500000	0.542103	MidAge
45	46	F	57000	43	9	1.1	No	Drama	3.545455	0.962990	Old
46	47	F	69000	35	22	2.8	Yes	Drama	4.090909	0.289570	MidAge
47	48	F	52000	47	14	1.6	No	Drama	3.318182	1.299700	Old
48	49	M	31000	25	42	3.4	Yes	Action	2.363636	-0.552204	MidAge
49	50	M	24000	20	33	4.7	No	Action	2.045455	-0.973092	Young

In [11]:

```python
import pandas as pd
from sklearn.preprocessing import LabelEncoder
df2=pd.read_csv('HW1.csv')
# Use LabelEncoder to convert the nominal variable 'Gender' to binary (0 for 'F', 1 for 'M')
le = LabelEncoder()
df2['Gender'] = le.fit_transform(df2['Gender'])
print(df2)
#We can also use if loop to get a similar output
```

```
    Cust ID  Gender  Income  Age  Rentals  Avg Per Visit Incidentals   Genre
0         1       1   45000   25       27            2.5         Yes  Action
1         2       0   54000   33       12            3.4          No   Drama
2         3       0   32000   20       42            1.6          No  Comedy
3         4       0   59000   70       16            4.2         Yes   Drama
4         5       1   37000   35       25            3.2         Yes  Action
5         6       1   18000   20       33            1.7          No  Action
6         7       0   29000   45       19            3.8          No   Drama
7         8       1   74000   25       31            2.4         Yes  Action
8         9       1   38000   21       18            2.1          No  Comedy
9        10       0   65000   40       21            3.3          No   Drama
10       11       0   41000   22       48            2.3         Yes   Drama
11       12       0   26000   22       29            2.9         Yes  Action
12       13       1   83000   46       14            3.6          No  Comedy
13       14       1   45000   36       24            2.7          No   Drama
14       15       1   68000   30       36            2.7         Yes  Comedy
15       16       1   17000   19       26            2.2         Yes  Action
16       17       1   36000   35       28            3.5         Yes   Drama
17       18       0    6000   16       39            1.8         Yes  Action
18       19       0   24000   25       41            3.1          No  Comedy
19       20       1   12000   16       23            2.2         Yes  Action
20       21       0   47000   52       11            3.1          No   Drama
21       22       1   25000   33       16            2.9         Yes   Drama
22       23       0    2000   15       30            2.5          No  Comedy
23       24       0   79000   35       22            3.8         Yes   Drama
24       25       1    1000   16       25            1.4         Yes  Comedy
25       26       0   56000   35       40            2.6         Yes  Action
26       27       0   62000   47       32            3.6          No   Drama
27       28       1   57000   52       22            4.1          No  Comedy
28       29       0   15000   18       37            2.1         Yes  Action
29       30       1   41000   25       17            1.4         Yes  Action
30       31       0   49000   56       15            3.2          No  Comedy
31       32       1   47000   30       21            3.1         Yes   Drama
32       33       1   23000   25       28            2.7          No  Action
33       34       0   29000   32       19            2.9         Yes  Action
34       35       1   74000   29       43            4.6         Yes  Action
35       36       0   29000   21       34            2.3          No  Comedy
36       37       1   89000   46       12            1.2          No  Comedy
37       38       1   41000   38       20            3.3         Yes   Drama
38       39       0   68000   35       19            3.9          No  Comedy
39       40       1   17000   19       32            1.8          No  Action
40       41       0   50000   33       17            1.4          No   Drama
41       42       1   32000   25       26            2.2         Yes  Action
42       43       0   49000   28       48            3.3         Yes   Drama
43       44       1   35000   24       24            1.7          No   Drama
44       45       1   56000   38       30            3.5         Yes   Drama
45       46       0   57000   43        9            1.1          No   Drama
46       47       0   69000   35       22            2.8         Yes   Drama
47       48       0   52000   47       14            1.6          No   Drama
48       49       1   31000   25       42            3.4         Yes  Action
49       50       1   24000   20       33            4.7          No  Action
```

In [13]:

```python
le = LabelEncoder()
df2['Incidentals'] = le.fit_transform(df2['Incidentals'])
print(df2)
```

```
    Cust ID  Gender  Income  Age  Rentals  Avg Per Visit  Incidentals   Genre
0         1       1   45000   25       27            2.5            1  Action
1         2       0   54000   33       12            3.4            0   Drama
2         3       0   32000   20       42            1.6            0  Comedy
3         4       0   59000   70       16            4.2            1   Drama
4         5       1   37000   35       25            3.2            1  Action
5         6       1   18000   20       33            1.7            0  Action
6         7       0   29000   45       19            3.8            0   Drama
7         8       1   74000   25       31            2.4            1  Action
8         9       1   38000   21       18            2.1            0  Comedy
9        10       0   65000   40       21            3.3            0   Drama
10       11       0   41000   22       48            2.3            1   Drama
11       12       0   26000   22       29            2.9            1  Action
12       13       1   83000   46       14            3.6            0  Comedy
13       14       1   45000   36       24            2.7            0   Drama
14       15       1   68000   30       36            2.7            1  Comedy
15       16       1   17000   19       26            2.2            1  Action
16       17       1   36000   35       28            3.5            1   Drama
17       18       0    6000   16       39            1.8            1  Action
18       19       0   24000   25       41            3.1            0  Comedy
19       20       1   12000   16       23            2.2            1  Action
20       21       0   47000   52       11            3.1            0   Drama
21       22       1   25000   33       16            2.9            1   Drama
22       23       0    2000   15       30            2.5            0  Comedy
23       24       0   79000   35       22            3.8            1   Drama
24       25       1    1000   16       25            1.4            1  Comedy
25       26       0   56000   35       40            2.6            1  Action
26       27       0   62000   47       32            3.6            0   Drama
27       28       1   57000   52       22            4.1            0  Comedy
28       29       0   15000   18       37            2.1            1  Action
29       30       1   41000   25       17            1.4            1  Action
30       31       0   49000   56       15            3.2            0  Comedy
31       32       1   47000   30       21            3.1            1   Drama
32       33       1   23000   25       28            2.7            0  Action
33       34       0   29000   32       19            2.9            1  Action
34       35       1   74000   29       43            4.6            1  Action
35       36       0   29000   21       34            2.3            0  Comedy
36       37       1   89000   46       12            1.2            0  Comedy
37       38       1   41000   38       20            3.3            1   Drama
38       39       0   68000   35       19            3.9            0  Comedy
39       40       1   17000   19       32            1.8            0  Action
40       41       0   50000   33       17            1.4            0   Drama
41       42       1   32000   25       26            2.2            1  Action
42       43       0   49000   28       48            3.3            1   Drama
43       44       1   35000   24       24            1.7            0   Drama
44       45       1   56000   38       30            3.5            1   Drama
45       46       0   57000   43        9            1.1            0   Drama
46       47       0   69000   35       22            2.8            1   Drama
47       48       0   52000   47       14            1.6            0   Drama
48       49       1   31000   25       42            3.4            1  Action
49       50       1   24000   20       33            4.7            0  Action
```

In [17]:

```python
df = pd.DataFrame(df2)

# Initialize new columns with 0
df['Genre_Action'] = 0
df['Genre_Comedy'] = 0


for index, row in df.iterrows():
    if row['Genre'] == 'Action':
        df.at[index, 'Genre_Action'] = 1
    elif row['Genre'] == 'Comedy':
        df.at[index, 'Genre_Comedy'] = 1


df = df.drop('Genre', axis=1)
print(df)
```

```
    Cust ID  Gender  Income  Age  Rentals  Avg Per Visit  Incidentals  \
0         1       1   45000   25       27            2.5            1
1         2       0   54000   33       12            3.4            0
2         3       0   32000   20       42            1.6            0
3         4       0   59000   70       16            4.2            1
4         5       1   37000   35       25            3.2            1
5         6       1   18000   20       33            1.7            0
6         7       0   29000   45       19            3.8            0
7         8       1   74000   25       31            2.4            1
8         9       1   38000   21       18            2.1            0
9        10       0   65000   40       21            3.3            0
10       11       0   41000   22       48            2.3            1
11       12       0   26000   22       29            2.9            1
12       13       1   83000   46       14            3.6            0
13       14       1   45000   36       24            2.7            0
14       15       1   68000   30       36            2.7            1
15       16       1   17000   19       26            2.2            1
16       17       1   36000   35       28            3.5            1
17       18       0    6000   16       39            1.8            1
18       19       0   24000   25       41            3.1            0
19       20       1   12000   16       23            2.2            1
20       21       0   47000   52       11            3.1            0
21       22       1   25000   33       16            2.9            1
22       23       0    2000   15       30            2.5            0
23       24       0   79000   35       22            3.8            1
24       25       1    1000   16       25            1.4            1
25       26       0   56000   35       40            2.6            1
26       27       0   62000   47       32            3.6            0
27       28       1   57000   52       22            4.1            0
28       29       0   15000   18       37            2.1            1
29       30       1   41000   25       17            1.4            1
30       31       0   49000   56       15            3.2            0
31       32       1   47000   30       21            3.1            1
32       33       1   23000   25       28            2.7            0
33       34       0   29000   32       19            2.9            1
34       35       1   74000   29       43            4.6            1
35       36       0   29000   21       34            2.3            0
36       37       1   89000   46       12            1.2            0
37       38       1   41000   38       20            3.3            1
38       39       0   68000   35       19            3.9            0
39       40       1   17000   19       32            1.8            0
40       41       0   50000   33       17            1.4            0
41       42       1   32000   25       26            2.2            1
42       43       0   49000   28       48            3.3            1
43       44       1   35000   24       24            1.7            0
44       45       1   56000   38       30            3.5            1
45       46       0   57000   43        9            1.1            0
46       47       0   69000   35       22            2.8            1
47       48       0   52000   47       14            1.6            0
48       49       1   31000   25       42            3.4            1
49       50       1   24000   20       33            4.7            0

    Genre_Action  Genre_Comedy
0              1             0
1              0             0
2              0             1
3              0             0
4              1             0
5              1             0
6              0             0
7              1             0
8              0             1
9              0             0
10             0             0
11             1             0
12             0             1
13             0             0
14             0             1
15             1             0
16             0             0
17             1             0
18             0             1
19             1             0
20             0             0
21             0             0
22             0             1
23             0             0
24             0             1
25             1             0
26             0             0
27             0             1
28             1             0
29             1             0
30             0             1
31             0             0
32             1             0
33             1             0
34             1             0
35             0             1
36             0             1
37             0             0
38             0             1
39             1             0
40             0             0
41             1             0
42             0             0
43             0             0
44             0             0
45             0             0
46             0             0
47             0             0
48             1             0
49             1             0
```

In [20]:

```python
2)numerics = ['int16', 'int32', 'int64', 'float16', 'float32', 'float64']
# get column names
cols_numeric = df.select_dtypes(include=numerics).columns.tolist()
df_numeric = df[cols_numeric]
display(HTML(df.head(5).to_html()))
corr=df_numeric.corr(method='pearson')
display(HTML(corr.to_html()))
```

	Cust ID	Gender	Income	Age	Rentals	Avg Per Visit	Incidentals	Genre_Action	Genre_Comedy
0	1	1	45000	25	27	2.5	1	1	0
1	2	0	54000	33	12	3.4	0	0	0
2	3	0	32000	20	42	1.6	0	0	1
3	4	0	59000	70	16	4.2	1	0	0
4	5	1	37000	35	25	3.2	1	1	0

	Cust ID	Gender	Income	Age	Rentals	Avg Per Visit	Incidentals	Genre_Action	Genre_Comedy
Cust ID	1.000000	0.030515	0.061042	0.008050	-0.013263	-0.047075	-0.041611	-0.014437	-0.090862
Gender	0.030515	1.000000	-0.063841	-0.227663	-0.025164	-0.029176	0.198718	0.303576	-0.022496
Income	0.061042	-0.063841	1.000000	0.627013	-0.274873	0.305221	-0.063841	-0.362514	0.071588
Age	0.008050	-0.227663	0.627013	1.000000	-0.549611	0.397372	-0.214184	-0.480794	0.016872
Rentals	-0.013263	-0.025164	-0.274873	-0.549611	1.000000	0.070302	0.297452	0.326052	-0.032456
Avg Per Visit	-0.047075	-0.029176	0.305221	0.397372	0.070302	1.000000	0.078885	-0.101414	-0.067208
Incidentals	-0.041611	0.198718	-0.063841	-0.214184	0.297452	0.078885	1.000000	0.386976	-0.397430
Genre_Action	-0.014437	0.303576	-0.362514	-0.480794	0.326052	-0.101414	0.386976	1.000000	-0.421464
Genre_Comedy	-0.090862	-0.022496	0.071588	0.016872	-0.032456	-0.067208	-0.397430	-0.421464	1.000000

In [21]:

```python
#Rentals are comparitively highly correlated with Age(negative correaltion) and Genre Action
```

In [25]:

```python
df=pd.read_csv('HW1.csv')
from scipy.stats import f_oneway

genre_action = df[df['Genre']=='Action']['Age']
genre_drama = df[df['Genre']=='Drama']['Age']
genre_comedy = df[df['Genre']=='Comedy']['Age']
print(f_oneway(genre_action, genre_drama,genre_comedy))
```

```
F_onewayResult(statistic=8.830189952460437, pvalue=0.0005550099142521056)
```

In [26]:

```python
from scipy.stats import f_oneway

genre_action = df[df['Genre']=='Action']['Rentals']
genre_drama = df[df['Genre']=='Drama']['Rentals']
genre_comedy = df[df['Genre']=='Comedy']['Rentals']
print(f_oneway(genre_action, genre_drama,genre_comedy))
```

```
F_onewayResult(statistic=3.1956455356365185, pvalue=0.049974070580121745)
```

In [31]:

```python
from scipy.stats import f_oneway

genre_action = df[df['Genre']=='Action']['Income']
genre_drama = df[df['Genre']=='Drama']['Income']
genre_comedy = df[df['Genre']=='Comedy']['Income']
print(f_oneway(genre_action, genre_drama,genre_comedy))
```

```
F_onewayResult(statistic=3.8076066330686538, pvalue=0.029337916333349683)
```

In [33]:

```python
# The F_onewayResult(statistic how much the variable is dependent on Genre.Higher the value of it more it is dependable.
#It has significant dependence but not strong. Age has highest dependency comparitively
```

In [ ]: