In [ ]:
```
print('''
7. Consider the following data frame containing a family name, gender of
income in each record.
FamilyName Gender MonthlyIncome_(Rs.)
Shah Male 44000.00
Vats Male 65000.00
Vats Female 43150.00
Kumar Female 66500.00
Vats Female 255000.00
Kumar Male 103000.00
Shah Male 55000.00
Shah Female 112400.00
Kumar Female 81030.00
Vats Male 71900.00
Write a program in Python using Pandas to perform the following:

a. Calculate and display familywise gross monthly income.
b. Display the highest and lowest monthly income for each family name
c. Calculate and display monthly income of all members earning income les
d. Display total number of females along with their average monthly incom
e. Delete rows with Monthly income less than the average income of all me
''')
```

```
7. Consider the following data frame containing a family name, gender of t
he family member and her/his monthly
income in each record.
FamilyName Gender MonthlyIncome_(Rs.)
Shah Male 44000.00
Vats Male 65000.00
Vats Female 43150.00
Kumar Female 66500.00
Vats Female 255000.00
Kumar Male 103000.00
Shah Male 55000.00
Shah Female 112400.00
Kumar Female 81030.00
Vats Male 71900.00
Write a program in Python using Pandas to perform the following:

a. Calculate and display familywise gross monthly income.
b. Display the highest and lowest monthly income for each family name
c. Calculate and display monthly income of all members earning income less
than Rs. 80000.00.
d. Display total number of females along with their average monthly income
e. Delete rows with Monthly income less than the average income of all mem
bers
```

In [ ]:
```
import pandas as pd

df = pd.read_csv("data_frame.csv",delimiter=" ")
df
```

Out[ ]:

	FamilyName	Gender	MonthlyIncome_(Rs.)
0	Shah	Male	44000.0
1	Vats	Male	65000.0
2	Vats	Female	43150.0
3	Kumar	Female	66500.0
4	Vats	Female	255000.0
5	Kumar	Male	103000.0
6	Shah	Male	55000.0
7	Shah	Female	112400.0
8	Kumar	Female	81030.0
9	Vats	Male	71900.0

In [ ]:
```
#a. Calculate and display familywise gross monthly income.

families = df.groupby("FamilyName")
families["MonthlyIncome_(Rs.)"].sum()
```

Out[ ]:
```
FamilyName
Kumar    250530.0
Shah     211400.0
Vats     435050.0
Name: MonthlyIncome_(Rs.), dtype: float64
```

In [ ]:
```
# b. Display the highest and lowest monthly income for each family name

highestIncome = families["MonthlyIncome_(Rs.)"].max()
lowestIncome = families["MonthlyIncome_(Rs.)"].min()
FamilyNames = families["FamilyName"].unique()
for i in range(len(FamilyNames)):
    print(f"highest Incode of {FamilyNames[i]}'s : {highestIncome[i]}")
    print(f"lowest Incode of {FamilyNames[i]}'s : {highestIncome[i]}")

print("\n\n")
```

```
highest Incode of ['Kumar']'s : 103000.0
lowest Incode of ['Kumar']'s : 103000.0
highest Incode of ['Shah']'s : 112400.0
lowest Incode of ['Shah']'s : 112400.0
highest Incode of ['Vats']'s : 255000.0
lowest Incode of ['Vats']'s : 255000.0
```

```
/tmp/ipykernel_5283/3072639210.py:7: FutureWarning: Series.__getitem__ tre
ating keys as positions is deprecated. In a future version, integer keys w
ill always be treated as labels (consistent with DataFrame behavior). To a
ccess a value by position, use `ser.iloc[pos]`
  print(f"highest Incode of {FamilyNames[i]}'s : {highestIncome[i]}")
/tmp/ipykernel_5283/3072639210.py:8: FutureWarning: Series.__getitem__ tre
ating keys as positions is deprecated. In a future version, integer keys w
ill always be treated as labels (consistent with DataFrame behavior). To a
ccess a value by position, use `ser.iloc[pos]`
  print(f"lowest Incode of {FamilyNames[i]}'s : {highestIncome[i]}")
```

In [ ]:
```
#c. Calculate and display monthly income of all members earning income le

below_80000_income = df[df["MonthlyIncome_(Rs.)"] < 80000]
below_80000_income
```

Out[ ]:

	FamilyName	Gender	MonthlyIncome_(Rs.)
0	Shah	Male	44000.0
1	Vats	Male	65000.0
2	Vats	Female	43150.0
3	Kumar	Female	66500.0
6	Shah	Male	55000.0
9	Vats	Male	71900.0

In [ ]:
```
# d. Display total number of females along with their average monthly inc

female_income = df[df["Gender"] == "Female"]
avrg_female_income = female_income["MonthlyIncome_(Rs.)"].mean()
no_of_females = female_income.shape[0]

print(f'There total of {no_of_females} with average income : {avrg_female
```

```
There total of 5 with average income : 111616.0
```

In [ ]:
```
# e. Delete rows with Monthly income less than the average income of all

avrg = df["MonthlyIncome_(Rs.)"].mean()
less_than_avrg = df[df["MonthlyIncome_(Rs.)"] < avrg]
df.drop(less_than_avrg.index,inplace=True)
df
```

Out[ ]:

	FamilyName	Gender	MonthlyIncome_(Rs.)
4	Vats	Female	255000.0
5	Kumar	Male	103000.0
7	Shah	Female	112400.0

In [ ]:

In [ ]:

In [ ]: