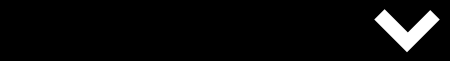
ITP 449

Data Exploration

Using Pandas and DataFrames

Lecture 6



More with NumPy, Pandas

Do the following:

Split a (8, 4) shaped matrix of random integers, ranging from 10 to 20, into 4 arrays with shapes (4, 2)

```
[[19 19 16 10]
 [17 11 15 15]
 [12 10 19 10]
 [10 16 13 16]
 [11 13 13 15]
 [15 10 10 16]
 [12 19 14 19]
 [10 16 11 12]]

 [[19 19]
 [17 11]
 [12 10]
 [10 16]]

 [[11 13]
 [15 10]
 [12 19]
 [10 16]]

 [[16 10]
 [15 15]
 [19 10]
 [13 16]]

 [[13 15]
 [10 16]
 [14 19]
 [11 12]]
```

File Edit View Navigate Code Refactor Run Tools VCS Window Help ITP449_Fall2020 - in_class_coding.py

ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py

in_class_coding

Pr...

in_class_coding.py

```
1  import numpy as np
2
3  x = np.random.randint(low=10, high=20, size=(8, 4))
4
5  [xL, xR] = np.hsplit(x, 2)
6  [xLT, xLB] = np.vsplit(xL, 2)
7  [xRT, xRB] = np.vsplit(xR, 2)
8
9  print(x, '\n\n', xLT, '\n\n', xLB, '\n\n', xRT, '\n\n', xRB)
10
```

13 128

hsplit splits the array horizontally, i.e side by side.

vsplit splits the array vertically, i.e top and bottom

ITP449_Fall2020 C:\
Class
.idea
Files
In Class Codin
in_class_coc
venv library r
External Libraries
Scratches and Conso

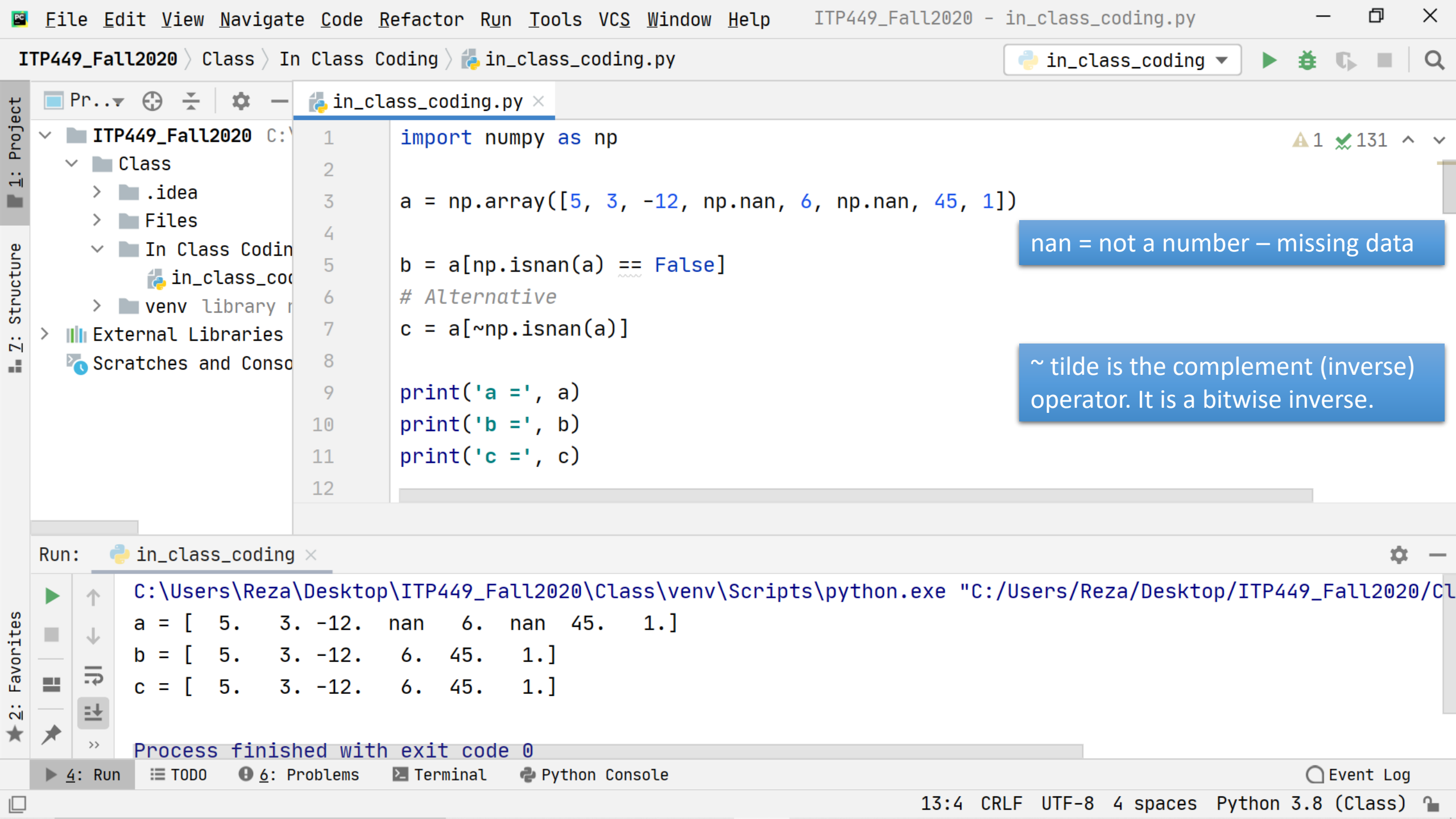
Run: in_class_coding

```
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Cl
[[10 18 19 11]
 [19 11 14 10]
 [14 19 13 15]
 [14 11 11 11]
 [10 14 18 14]
 [19 19 14 17]
 [13 11 13 19]]
```

1: Project 7: Structure 2: Favorites

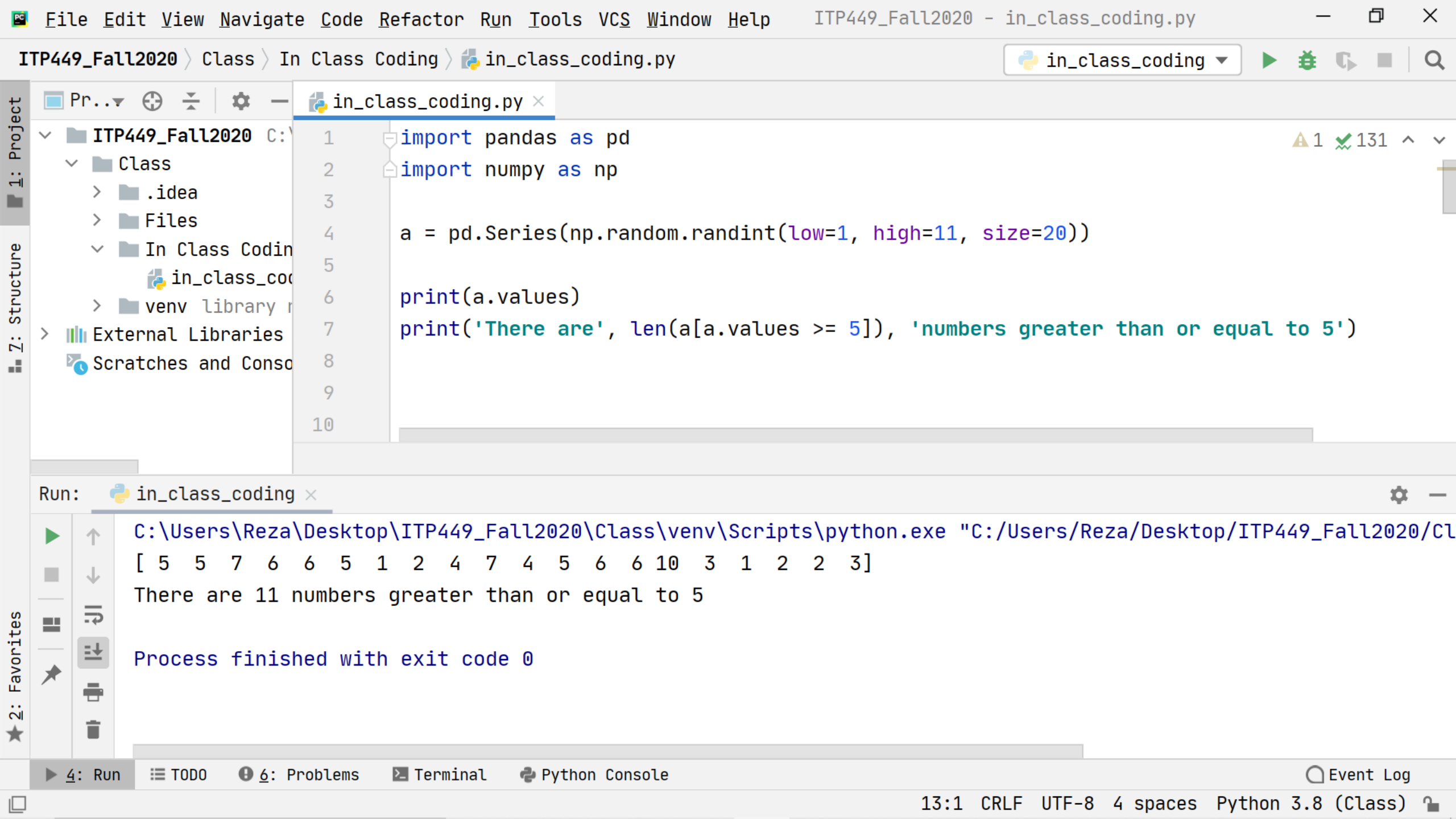
4: Run TODO 6: Problems Terminal Python Console Event Log

32:1 CRLF UTF-8 4 spaces Python 3.8 (Class)


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
in_class_coding.py
ITP449_Fall2020 C:\
Class
.idea
Files
In Class Codin
in_class_coc
venv library r
External Libraries
Scratches and Conso
1 import numpy as np
2
3 a = np.array([5, 3, -12, np.nan, 6, np.nan, 45, 1])
4
5 b = a[np.isnan(a) == False]
6 # Alternative
7 c = a[~np.isnan(a)]
8
9 print('a =', a)
10 print('b =', b)
11 print('c =', c)
12
1
131
nan = not a number – missing data
~ tilde is the complement (inverse) operator. It is a bitwise inverse.
Run: in_class_coding
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Cl
a = [5. 3. -12. nan 6. nan 45. 1.]
b = [5. 3. -12. 6. 45. 1.]
c = [5. 3. -12. 6. 45. 1.]
Process finished with exit code 0
1: Project
7: Structure
2: Favorites
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
13:4 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

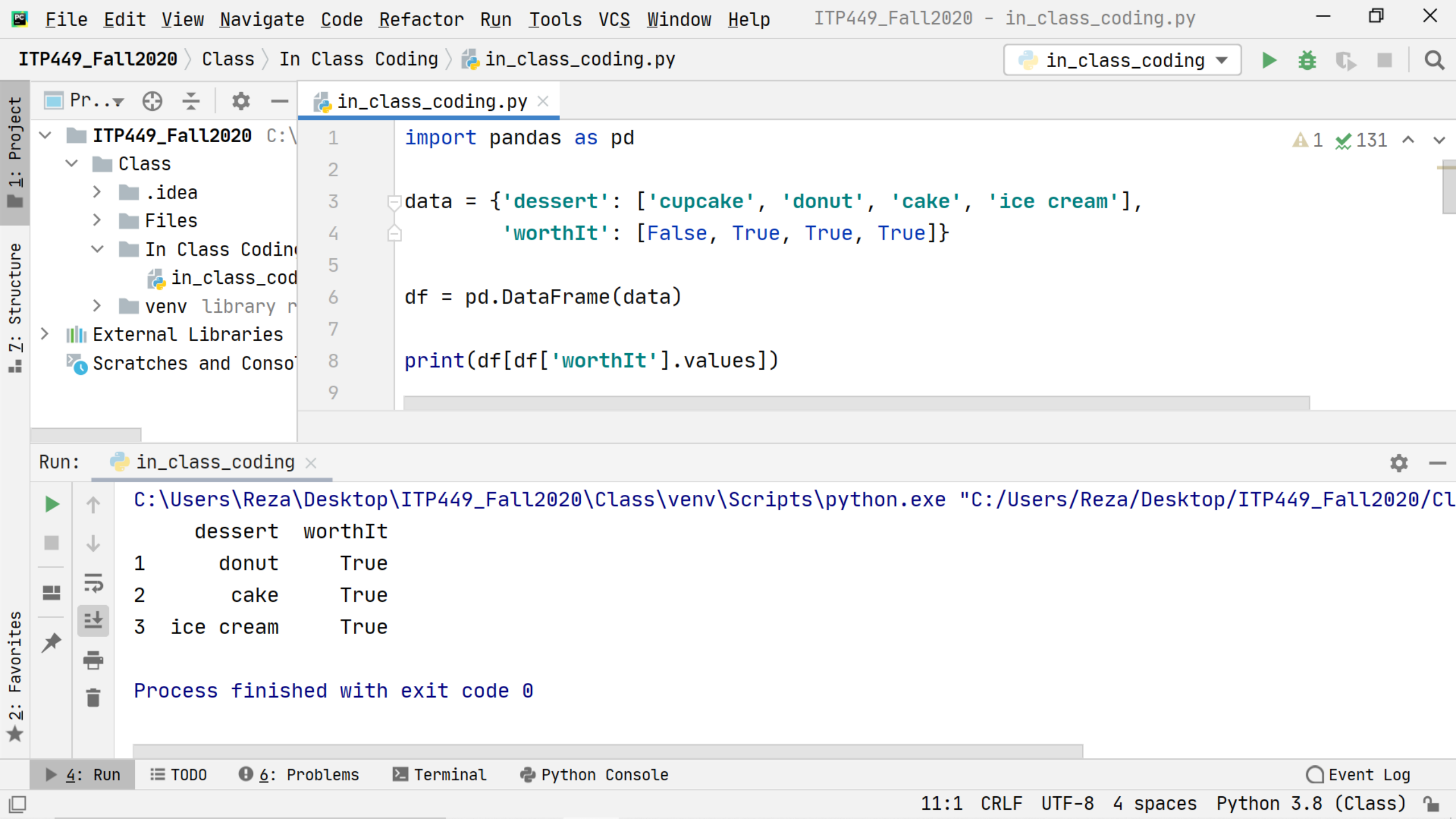
Create a Series of length 20 that contains random integers between 1 and 10. Then print the number of integers in the Series that are greater than or equal to 5.


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
1: Project
7: Structure
2: Favorites
Pr...
ITP449_Fall2020 C:\
Class
.idea
Files
In Class Codin
in_class_coc
venv library r
External Libraries
Scratches and Conso
in_class_coding.py
1 import pandas as pd
2 import numpy as np
3
4 a = pd.Series(np.random.randint(low=1, high=11, size=20))
5
6 print(a.values)
7 print('There are', len(a[a.values >= 5]), 'numbers greater than or equal to 5')
8
9
10
1
131
Run: in_class_coding
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Cl
[5 5 7 6 6 5 1 2 4 7 4 5 6 6 10 3 1 2 2 3]
There are 11 numbers greater than or equal to 5
Process finished with exit code 0
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
13:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

Create the DataFrame shown below and filter to show only the rows for desserts that are worth the calories.

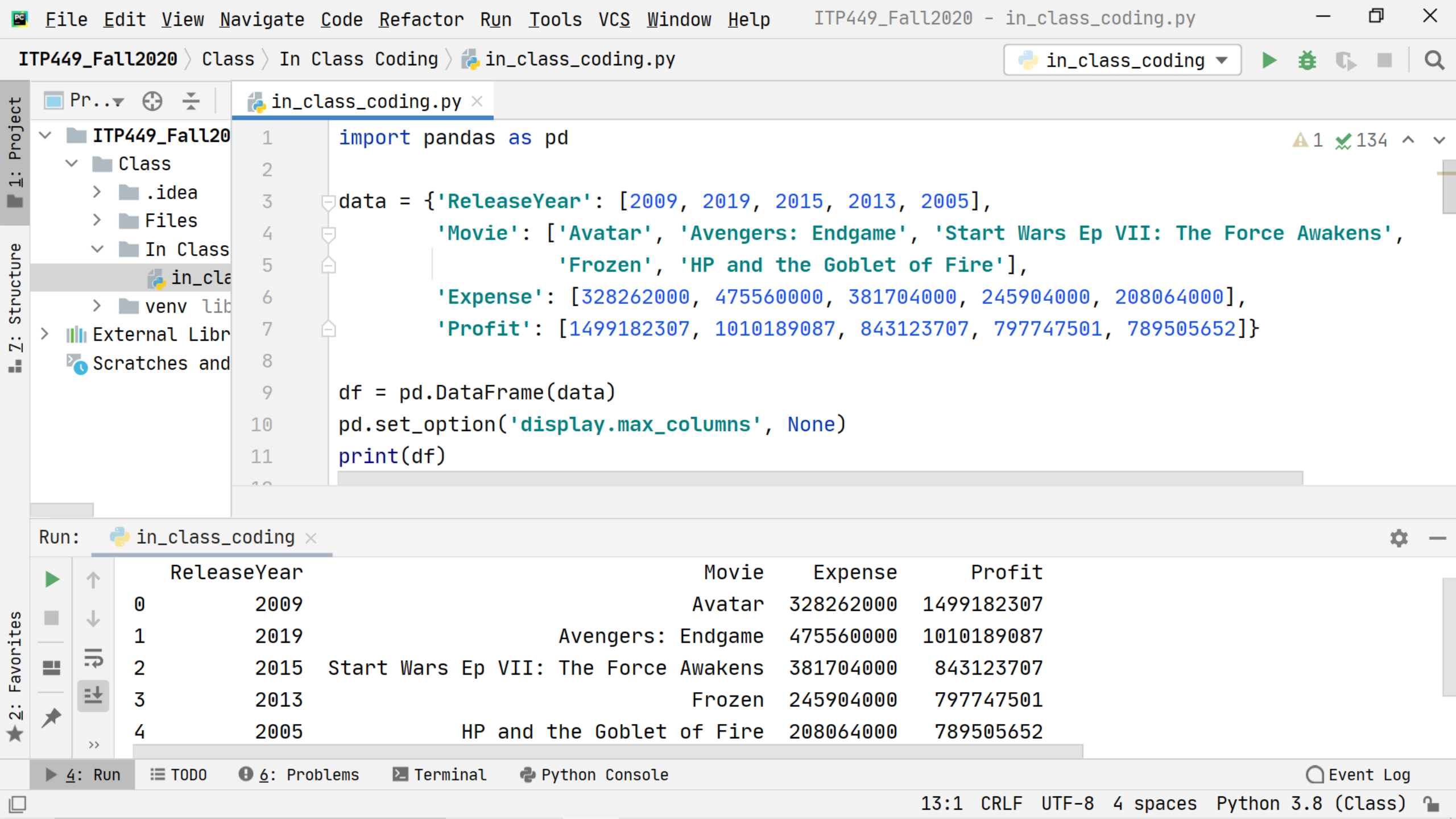
	dessert	worthIt
0	cupcake	False
1	donut	True
2	cake	True
3	ice cream	True


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
Pr...
in_class_coding.py
ITP449_Fall2020 C:\
Class
.idea
Files
In Class Codin
in_class_cod
venv library r
External Libraries
Scratches and Conso
1 import pandas as pd
2
3 data = {'dessert': ['cupcake', 'donut', 'cake', 'ice cream'],
4 'worthIt': [False, True, True, True]}
5
6 df = pd.DataFrame(data)
7
8 print(df[df['worthIt'].values])
9
1
131
Run: in_class_coding
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Cl
dessert worthIt
1 donut True
2 cake True
3 ice cream True
Process finished with exit code 0
1: Project
7: Structure
2: Favorites
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
11:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

Create a DataFrame with the following data on the highest grossing movies:

	ReleaseYear	Movie	Expense	Profit
0	2009	Avatar	328262000	1499182307
1	2019	Avengers: Endgame	475560000	1010189087
2	2015	Star Wars Ep VII: The Force Awakens	381704000	843123707
3	2013	Frozen	245904000	797747501
4	2005	HP and the Goblet of Fire	208064000	789505652


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
1: Project
7: Structure
2: Favorites
in_class_coding.py
1 import pandas as pd
2
3 data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
4 'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
5 'Frozen', 'HP and the Goblet of Fire'],
6 'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
7 'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}
8
9 df = pd.DataFrame(data)
10 pd.set_option('display.max_columns', None)
11 print(df)
ITP449_Fall20
Class
.idea
Files
In Class
in_cla
venv lib
External Libr
Scratches and
Run: in_class_coding
ReleaseYear Movie Expense Profit
0 2009 Avatar 328262000 1499182307
1 2019 Avengers: Endgame 475560000 1010189087
2 2015 Start Wars Ep VII: The Force Awakens 381704000 843123707
3 2013 Frozen 245904000 797747501
4 2005 HP and the Goblet of Fire 208064000 789505652
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
13:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Selection

Do the following:

Select the column *Movie* from the DataFrame.

in_class_coding.py

```python
import pandas as pd

data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
        'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
                  'Frozen', 'HP and the Goblet of Fire'],
        'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
        'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}

df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)
# Single column selection. All of the following are equivalent
print(df['Movie'])
print(df.Movie)
print(df.loc[:, 'Movie'])
print(df.iloc[:, 1])
```

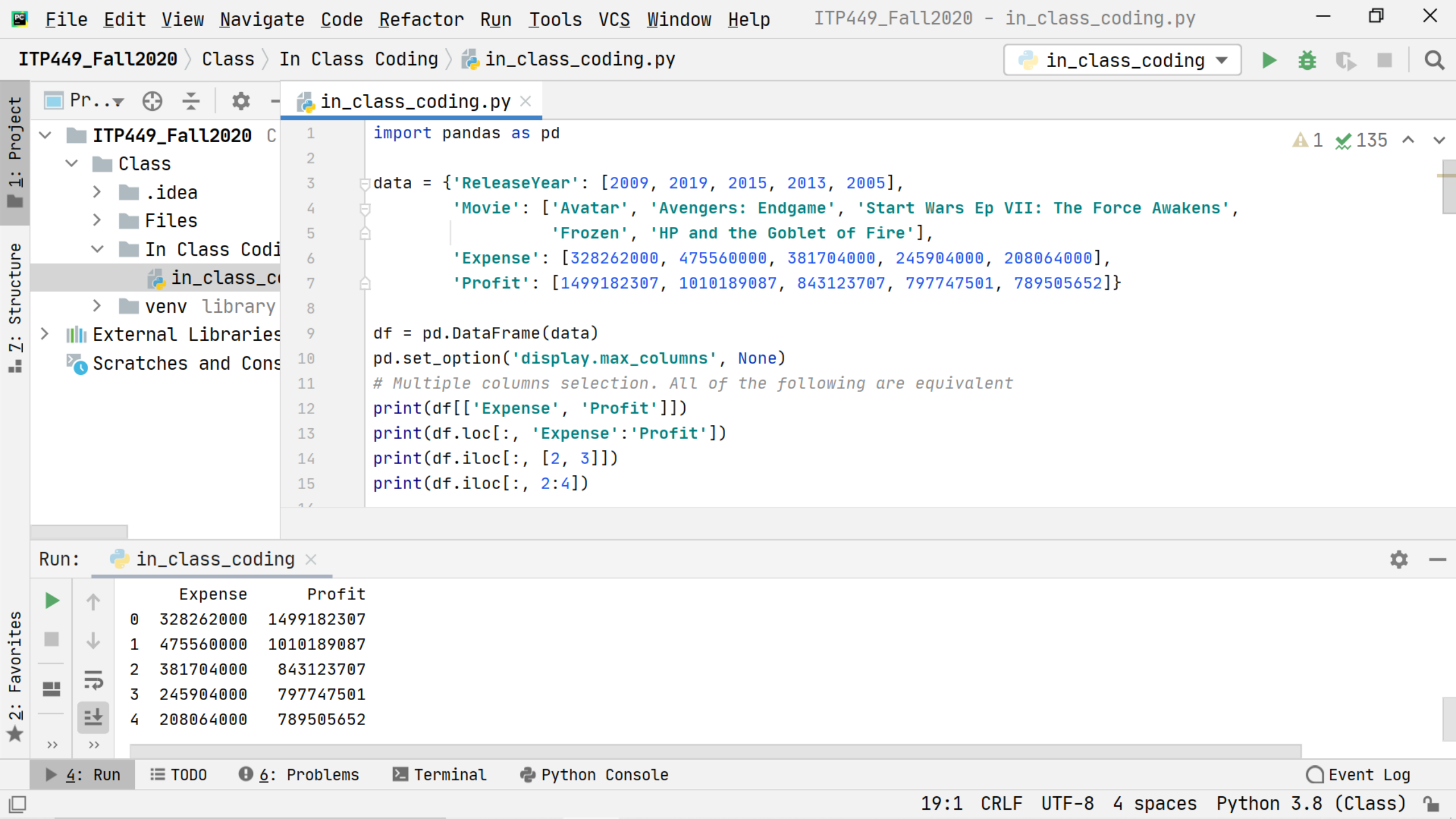
1 134

Run: in_class_coding

```
0                                  Avatar
1                      Avengers: Endgame
2    Start Wars Ep VII: The Force Awakens
3                                  Frozen
4              HP and the Goblet of Fire
Name: Movie, dtype: object
```

Do the following:

Select the columns *Expense* and *Profit* from the DataFrame.


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
Pr..
in_class_coding.py
ITP449_Fall2020 C
Class
.idea
Files
In Class Codi
in_class_c
venv library
External Libraries
Scratches and Cons
1: Project
7: Structure
2: Favorites
1 135
import pandas as pd
data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
'Frozen', 'HP and the Goblet of Fire'],
'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}
df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)
# Multiple columns selection. All of the following are equivalent
print(df[['Expense', 'Profit']])
print(df.loc[:, 'Expense':'Profit'])
print(df.iloc[:, [2, 3]])
print(df.iloc[:, 2:4])
Run: in_class_coding
Expense Profit
0 328262000 1499182307
1 475560000 1010189087
2 381704000 843123707
3 245904000 797747501
4 208064000 789505652
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
19:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Filtering

Do the following:

Create a condition that determines whether *ReleaseYear* is greater than or equal to 2010.

```python
import pandas as pd

data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
        'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
                  'Frozen', 'HP and the Goblet of Fire'],
        'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
        'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}

df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)
# Row filtering. ReleaseYear >= 2010
print(df['ReleaseYear'] >= 2010)
print(df.ReleaseYear >= 2010)
```



```
0    False
1     True
2     True
3     True
4    False
Name: ReleaseYear, dtype: bool
```

Do the following:

Display the DataFrame where *ReleaseYear* is greater than or equal to 2010.

File Edit View Navigate Code Refactor Run Tools VCS Window Help ITP449_Fall2020 - in_class_coding.py

ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py

in_class_coding

Pr...

in_class_coding.py

```
ITP449_Fall2020 C
  Class
    .idea
    Files
    In Class Codi
      in_class_c
    venv library
  External Libraries
  Scratches and Cons
```

```
1  import pandas as pd
2
3  data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
4          'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
5                    'Frozen', 'HP and the Goblet of Fire'],
6          'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
7          'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}
8
9  df = pd.DataFrame(data)
10 pd.set_option('display.max_columns', None)
11 # DataFrame selection with row filtering. ReleaseYear >= 2010
12 print(df[df['ReleaseYear'] >= 2010])
13 print(df[df.ReleaseYear >= 2010])
14 print(df.loc[df['ReleaseYear'] >= 2010, :])
15 print(df.loc[df.ReleaseYear >= 2010, :])
16
```

1 137

Run: in_class_coding

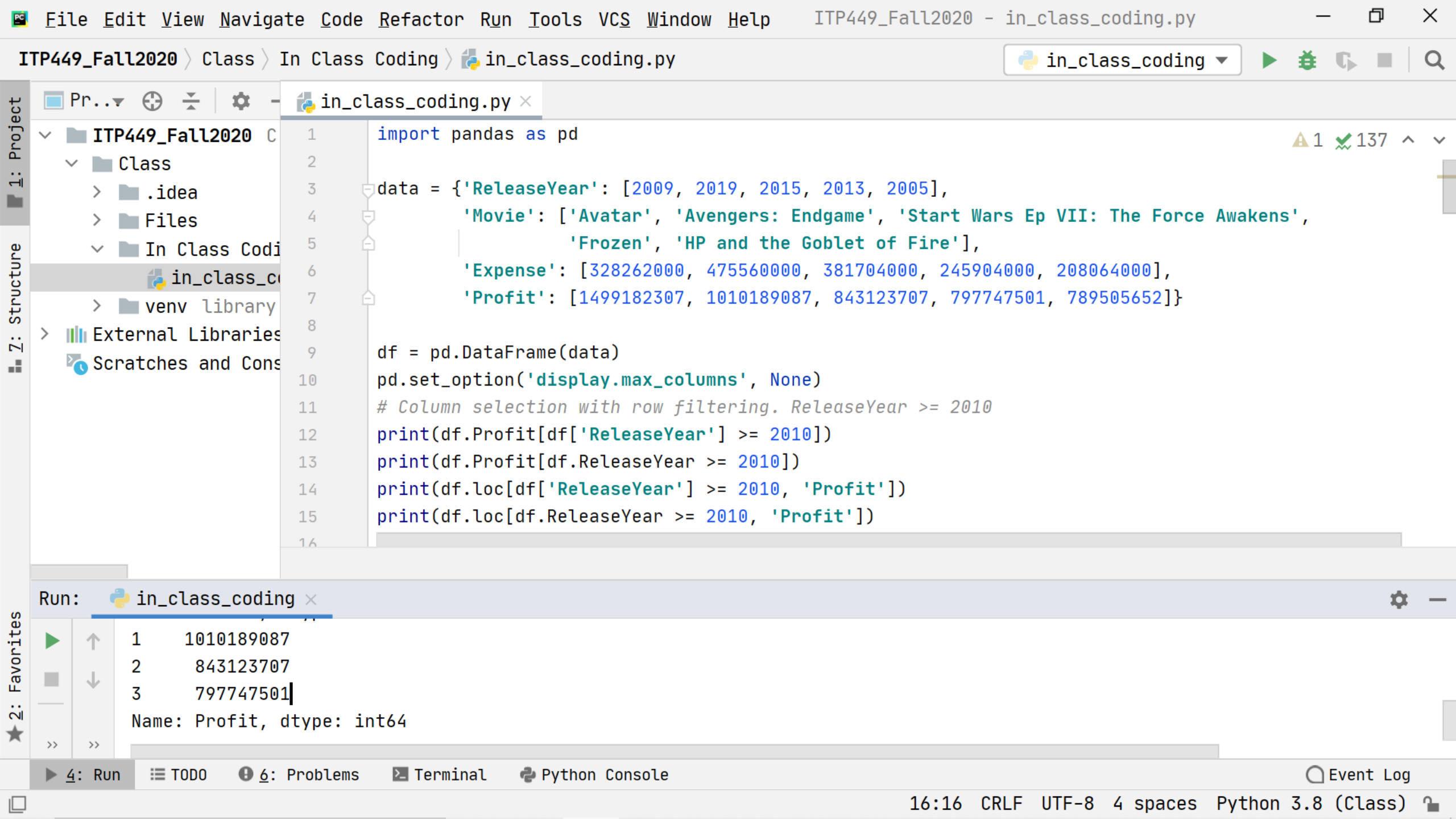
```
   ReleaseYear                                Movie    Expense      Profit
1         2019                     Avengers: Endgame  475560000  1010189087
2         2015  Start Wars Ep VII: The Force Awakens  381704000   843123707
3         2013                               Frozen  245904000   797747501
```

4: Run TODO 6: Problems Terminal Python Console Event Log

14:76 CRLF UTF-8 4 spaces Python 3.8 (Class)

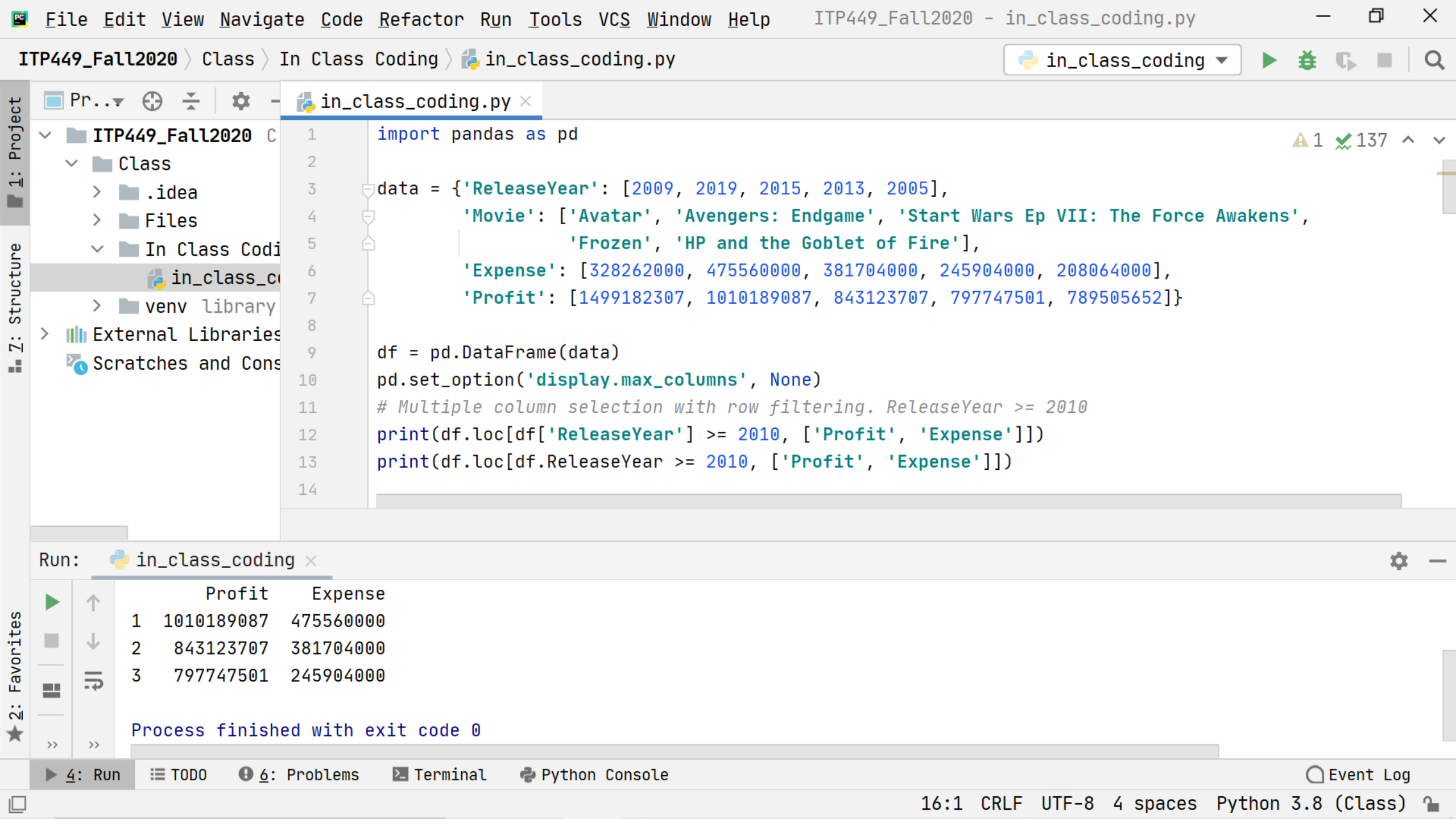
Do the following:

Display the *Profit* column where *ReleaseYear* is greater than or equal to 2010.


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
Pr..
in_class_coding.py
ITP449_Fall2020 C
Class
.idea
Files
In Class Codi
in_class_c
venv library
External Libraries
Scratches and Cons
1 import pandas as pd
2
3 data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
4 'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
5 'Frozen', 'HP and the Goblet of Fire'],
6 'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
7 'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}
8
9 df = pd.DataFrame(data)
10 pd.set_option('display.max_columns', None)
11 # Column selection with row filtering. ReleaseYear >= 2010
12 print(df.Profit[df['ReleaseYear'] >= 2010])
13 print(df.Profit[df.ReleaseYear >= 2010])
14 print(df.loc[df['ReleaseYear'] >= 2010, 'Profit'])
15 print(df.loc[df.ReleaseYear >= 2010, 'Profit'])
1 137
Run: in_class_coding
1 1010189087
2 843123707
3 797747501
Name: Profit, dtype: int64
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
16:16 CRLF UTF-8 4 spaces Python 3.8 (Class)
1: Project
7: Structure
2: Favorites

Do the following:

Display columns *Profit* and *Expense* where *ReleaseYear* is greater than or equal to 2010.


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
1: Project
7: Structure
2: Favorites
Pr..
ITP449_Fall2020 C
Class
.idea
Files
In Class Codi
in_class_c
venv library
External Libraries
Scratches and Cons
in_class_coding.py
1 import pandas as pd
2
3 data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
4 'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
5 'Frozen', 'HP and the Goblet of Fire'],
6 'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
7 'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}
8
9 df = pd.DataFrame(data)
10 pd.set_option('display.max_columns', None)
11 # Multiple column selection with row filtering. ReleaseYear >= 2010
12 print(df.loc[df['ReleaseYear'] >= 2010, ['Profit', 'Expense']])
13 print(df.loc[df.ReleaseYear >= 2010, ['Profit', 'Expense']])
14
1 137
Run: in_class_coding
Profit Expense
1 1010189087 475560000
2 843123707 381704000
3 797747501 245904000
Process finished with exit code 0
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
16:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

```python
import pandas as pd

data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
        'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
                  'Frozen', 'HP and the Goblet of Fire'],
        'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
        'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}

df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)
# Multiple column selection with row filtering. ReleaseYear >= 2010 in steps
dfFil = df[df['ReleaseYear'] >= 2010]
print(dfFil[['Profit', 'Expense']])
print(dfFil.loc[:, ['Profit', 'Expense']])
```



```
      Profit    Expense
1  1010189087  475560000
2   843123707  381704000
3   797747501  245904000

Process finished with exit code 0
```

Arithmetic

Do the following:

Calculate the *Profit* in millions of dollars: divide the *Profit* column by 1,000,000

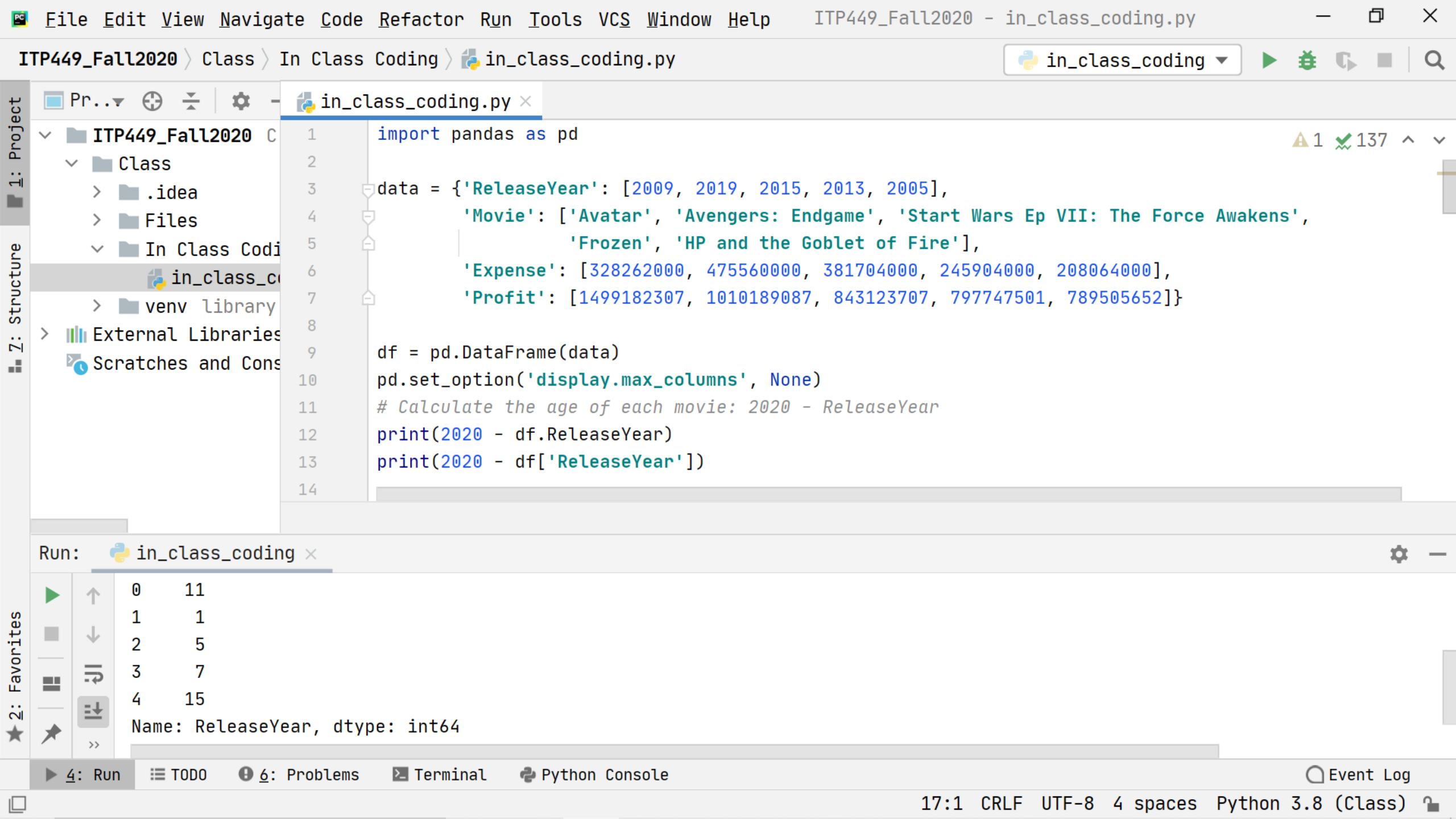
```python
import pandas as pd

data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
        'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
                  'Frozen', 'HP and the Goblet of Fire'],
        'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
        'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}

df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)
# Calculate Profit in millions. Divide by 1,000,000
print(df.Profit / 1000000)
print(df['Profit'] / 1000000)
```


Run: in_class_coding

```
0    1499.182307
1    1010.189087
2     843.123707
3     797.747501
4     789.505652
Name: Profit, dtype: float64
```

Do the following:

Calculate the age of each movie from the *ReleaseYear*


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
Pr..
in_class_coding.py
ITP449_Fall2020 C
Class
.idea
Files
In Class Codi
in_class_c
venv library
External Libraries
Scratches and Cons
1 137
import pandas as pd
data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
'Frozen', 'HP and the Goblet of Fire'],
'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}
df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)
# Calculate the age of each movie: 2020 - ReleaseYear
print(2020 - df.ReleaseYear)
print(2020 - df['ReleaseYear'])
Run: in_class_coding
0 11
1 1
2 5
3 7
4 15
Name: ReleaseYear, dtype: int64
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
17:1 CRLF UTF-8 4 spaces Python 3.8 (Class)
1: Project
7: Structure
2: Favorites

Do the following:

Calculate *Expense* as a percentage of *Profit*

```python
import pandas as pd

data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
        'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
                  'Frozen', 'HP and the Goblet of Fire'],
        'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
        'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}

df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)
# Calculate Expense as percentage of Profit
print(df.Expense / df.Profit * 100)
print(df['Expense'] / df['Profit'] * 100)
```

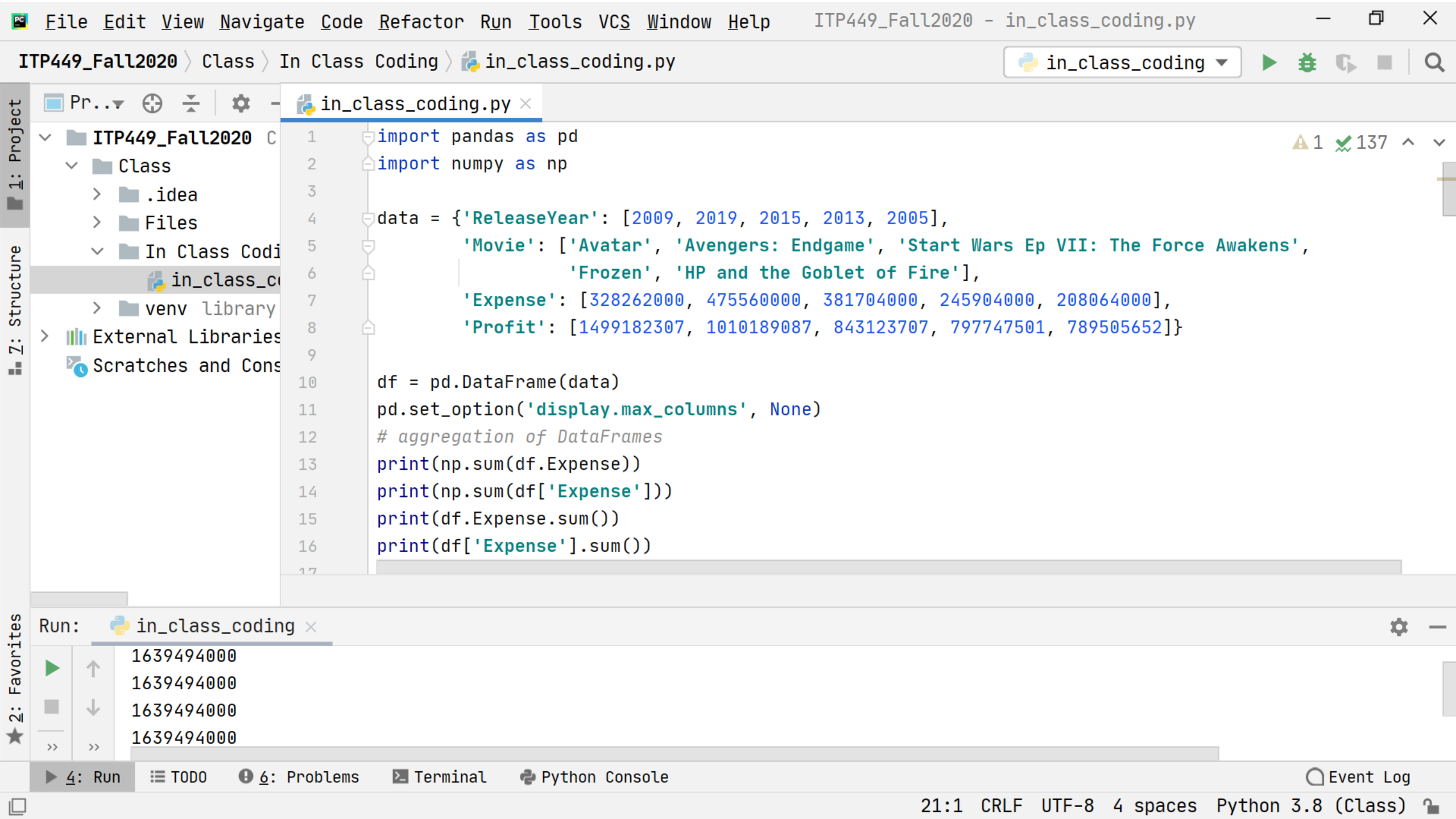


```
0    21.896070
1    47.076335
2    45.272597
3    30.824791
4    26.353706
dtype: float64
```

Aggregation

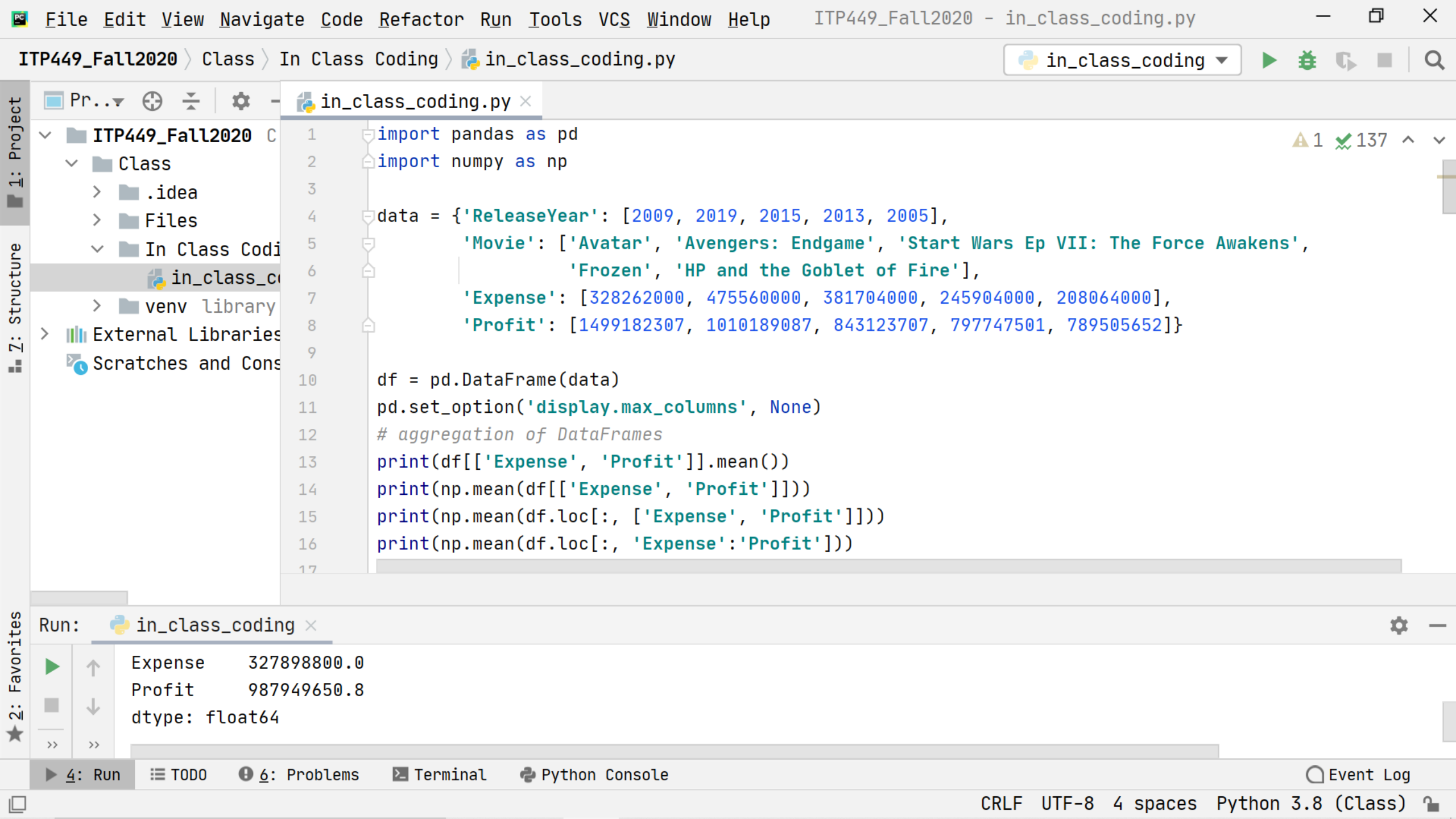
Do the following:

Calculate the total *Expense* for all the movies in the DataFrame


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
1: Project
Pr..
in_class_coding.py
ITP449_Fall2020 C
Class
.idea
Files
In Class Codi
in_class_c
venv library
External Libraries
Scratches and Cons
1 137
import pandas as pd
import numpy as np
data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
'Frozen', 'HP and the Goblet of Fire'],
'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}
df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)
# aggregation of DataFrames
print(np.sum(df.Expense))
print(np.sum(df['Expense']))
print(df.Expense.sum())
print(df['Expense'].sum())
Run: in_class_coding
1639494000
1639494000
1639494000
1639494000
7: Structure
2: Favorites
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
21:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

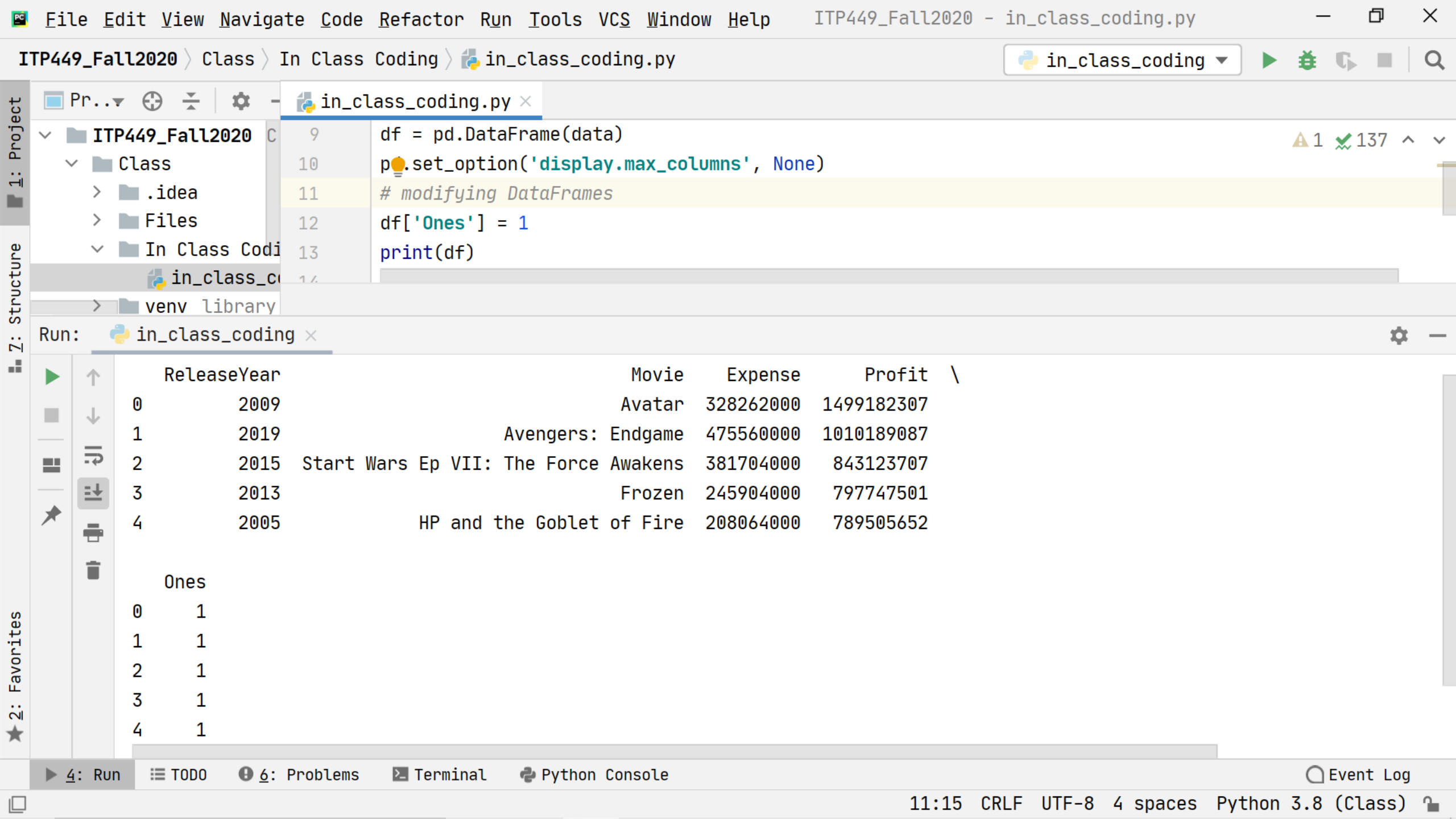
Calculate the mean *Expense* and *Profit* for all the movies in the DataFrame


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
1: Project
Pr...
in_class_coding.py
ITP449_Fall2020 C
Class
.idea
Files
In Class Codi
in_class_c
venv library
External Libraries
Scratches and Cons
1
137
import pandas as pd
import numpy as np
data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
'Frozen', 'HP and the Goblet of Fire'],
'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}
df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)
# aggregation of DataFrames
print(df[['Expense', 'Profit']].mean())
print(np.mean(df[['Expense', 'Profit']]))
print(np.mean(df.loc[:, ['Expense', 'Profit']]))
print(np.mean(df.loc[:, 'Expense':'Profit']))
Run: in_class_coding
Expense 327898800.0
Profit 987949650.8
dtype: float64
2: Favorites
7: Structure
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
CRLF UTF-8 4 spaces Python 3.8 (Class)

Modifying DataFrames

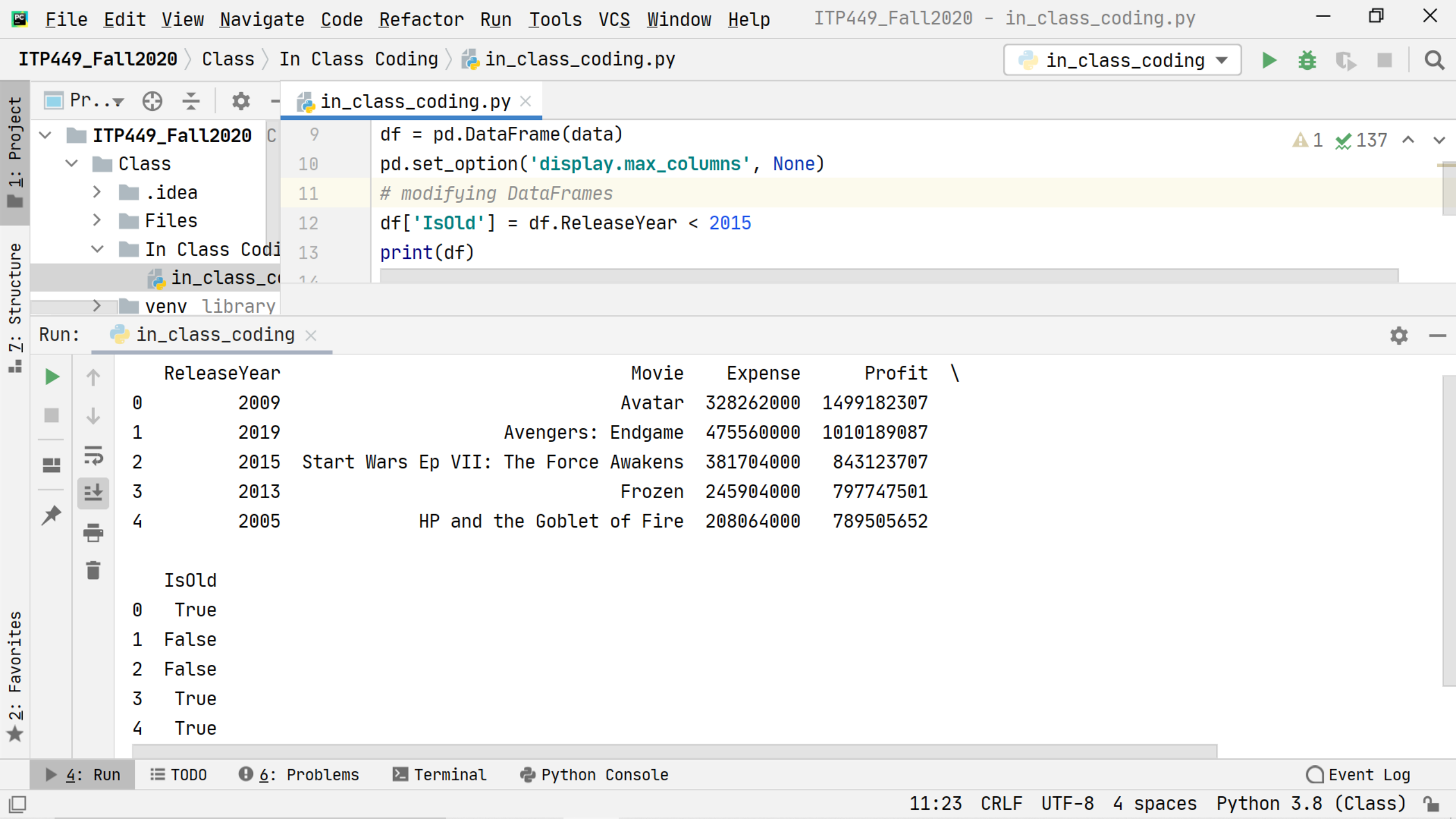
Do the following:

Add a column of 1’s to the DataFrame


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
Pr..
in_class_coding.py
ITP449_Fall2020
Class
.idea
Files
In Class Codi
in_class_c
venv library
9 df = pd.DataFrame(data)
10 p.set_option('display.max_columns', None)
11 # modifying DataFrames
12 df['Ones'] = 1
13 print(df)
1
137
Run: in_class_coding
ReleaseYear Movie Expense Profit \
0 2009 Avatar 328262000 1499182307
1 2019 Avengers: Endgame 475560000 1010189087
2 2015 Start Wars Ep VII: The Force Awakens 381704000 843123707
3 2013 Frozen 245904000 797747501
4 2005 HP and the Goblet of Fire 208064000 789505652
Ones
0 1
1 1
2 1
3 1
4 1
1: Project
7: Structure
2: Favorites
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
11:15 CRLF UTF-8 4 spaces Python 3.8 (Class)

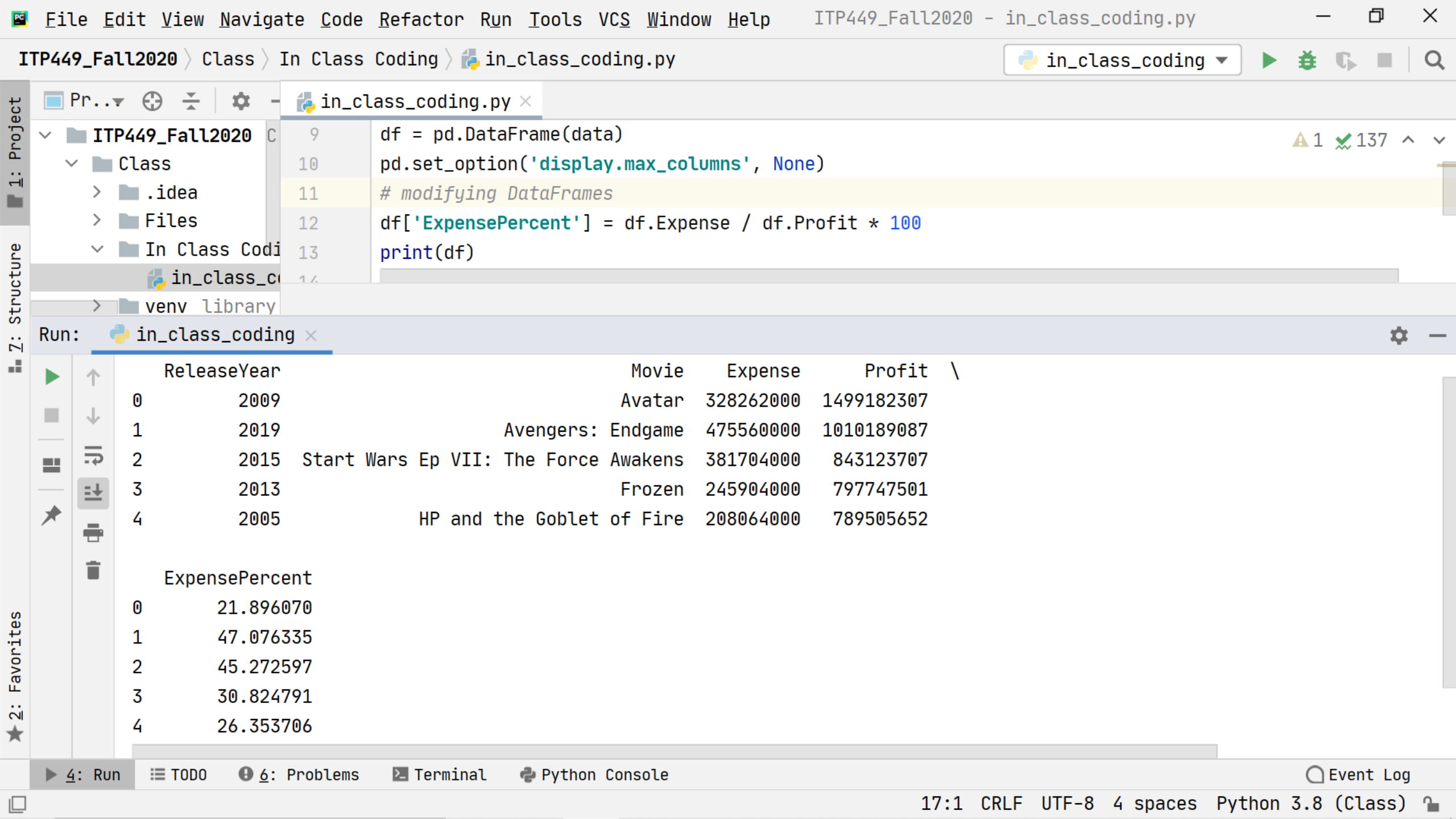
Do the following:

Add a column titled *IsOld* of boolean type that is True for any movie with a release date older than 2015


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
1: Project
7: Structure
2: Favorites
Pr..
in_class_coding.py
ITP449_Fall2020
Class
.idea
Files
In Class Codi
in_class_c
venv library
9 df = pd.DataFrame(data)
10 pd.set_option('display.max_columns', None)
11 # modifying DataFrames
12 df['IsOld'] = df.ReleaseYear < 2015
13 print(df)
1
137
Run: in_class_coding
ReleaseYear Movie Expense Profit \
0 2009 Avatar 328262000 1499182307
1 2019 Avengers: Endgame 475560000 1010189087
2 2015 Start Wars Ep VII: The Force Awakens 381704000 843123707
3 2013 Frozen 245904000 797747501
4 2005 HP and the Goblet of Fire 208064000 789505652
IsOld
0 True
1 False
2 False
3 True
4 True
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
11:23 CRLF UTF-8 4 spaces Python 3.8 (Class)

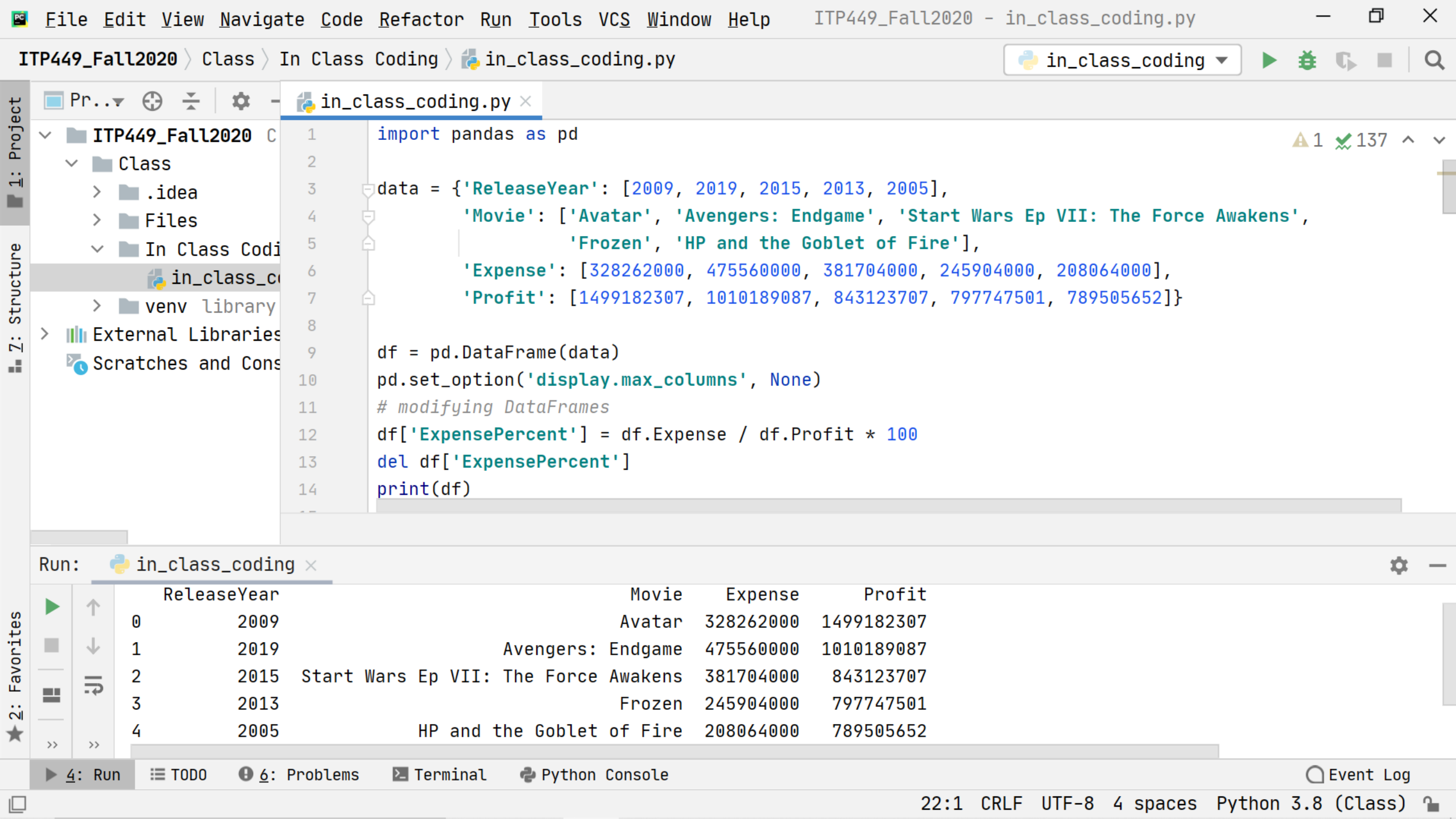
Do the following:

Add a column titled *ExpensePercent* of float type that is equal to the *Expense* as a percentage of *Profit*


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
Pr..
in_class_coding.py
ITP449_Fall2020
Class
.idea
Files
In Class Codi
in_class_c
venv library
9 df = pd.DataFrame(data)
10 pd.set_option('display.max_columns', None)
11 # modifying DataFrames
12 df['ExpensePercent'] = df.Expense / df.Profit * 100
13 print(df)
1
137
Run: in_class_coding
ReleaseYear Movie Expense Profit \
0 2009 Avatar 328262000 1499182307
1 2019 Avengers: Endgame 475560000 1010189087
2 2015 Start Wars Ep VII: The Force Awakens 381704000 843123707
3 2013 Frozen 245904000 797747501
4 2005 HP and the Goblet of Fire 208064000 789505652
ExpensePercent
0 21.896070
1 47.076335
2 45.272597
3 30.824791
4 26.353706
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
17:1 CRLF UTF-8 4 spaces Python 3.8 (Class)
1: Project
7: Structure
2: Favorites

Do the following:

Delete the *ExpensePercent* column from the DataFrame


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
1: Project
7: Structure
2: Favorites
Pr...
ITP449_Fall2020 C
Class
.idea
Files
In Class Codi
in_class_c
venv library
External Libraries
Scratches and Cons
in_class_coding.py
1 import pandas as pd
2
3 data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
4 'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
5 'Frozen', 'HP and the Goblet of Fire'],
6 'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
7 'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}
8
9 df = pd.DataFrame(data)
10 pd.set_option('display.max_columns', None)
11 # modifying DataFrames
12 df['ExpensePercent'] = df.Expense / df.Profit * 100
13 del df['ExpensePercent']
14 print(df)
1
137
Run: in_class_coding
ReleaseYear Movie Expense Profit
0 2009 Avatar 328262000 1499182307
1 2019 Avengers: Endgame 475560000 1010189087
2 2015 Start Wars Ep VII: The Force Awakens 381704000 843123707
3 2013 Frozen 245904000 797747501
4 2005 HP and the Goblet of Fire 208064000 789505652
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
22:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Combine methods for analysis

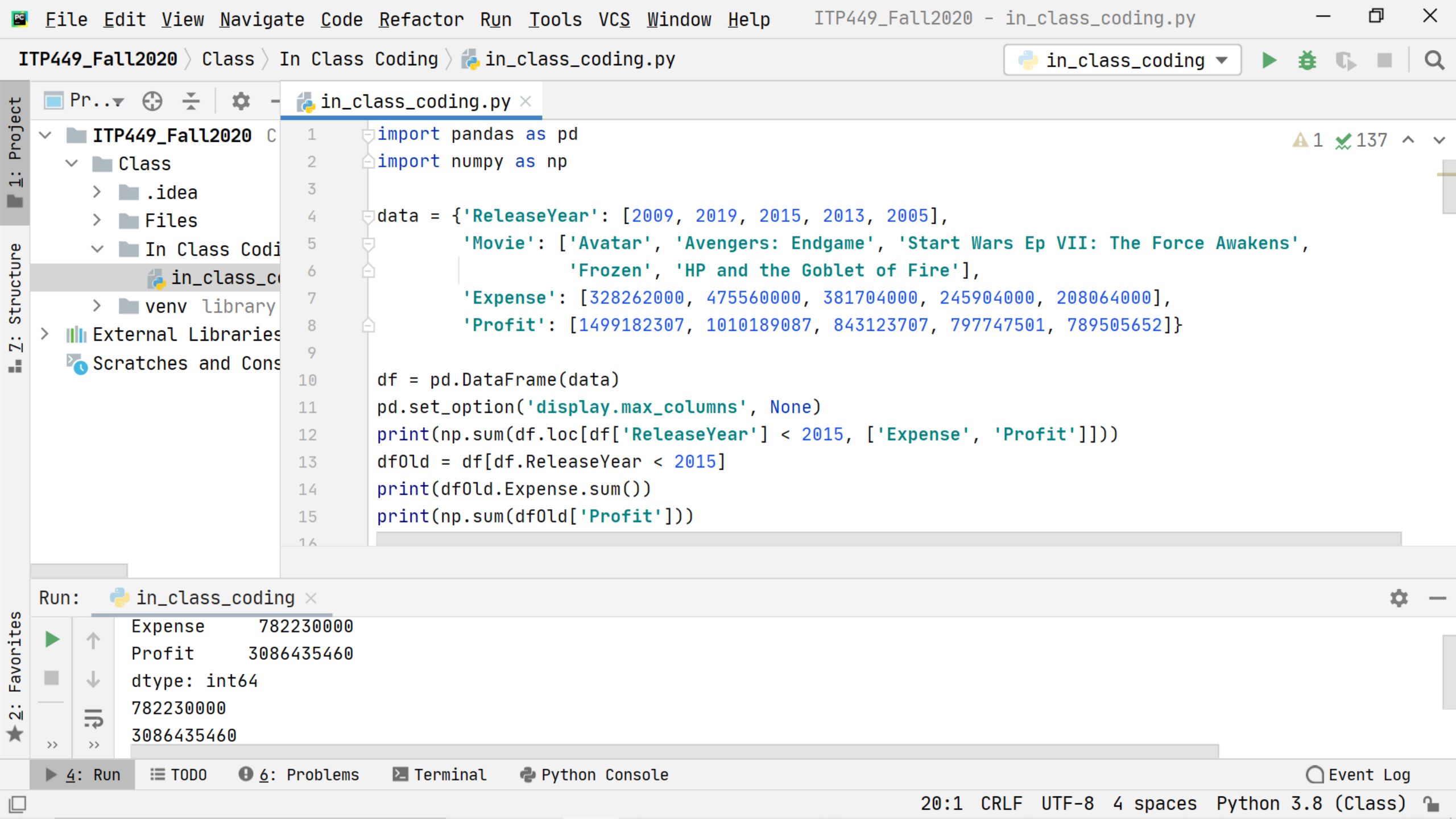
Do the following:

Calculate the total *Expense* and *Profit* for movies released before 2015

```python
import pandas as pd
import numpy as np

data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
        'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
                  'Frozen', 'HP and the Goblet of Fire'],
        'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
        'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}

df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)

dfOld = df[df.ReleaseYear < 2015]
print(dfOld.Expense.sum())
print(np.sum(dfOld['Profit']))
```



```
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Class/In Class Cod
782230000
3086435460

Process finished with exit code 0
```


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
Pr..
in_class_coding.py
ITP449_Fall2020 C
Class
.idea
Files
In Class Codi
in_class_c
venv library
External Libraries
Scratches and Cons
1 137
import pandas as pd
import numpy as np
data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
'Frozen', 'HP and the Goblet of Fire'],
'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}
df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)
print(np.sum(df.loc[df['ReleaseYear'] < 2015, ['Expense', 'Profit']]))
dfOld = df[df.ReleaseYear < 2015]
print(dfOld.Expense.sum())
print(np.sum(dfOld['Profit']))
Run: in_class_coding
Expense 782230000
Profit 3086435460
dtype: int64
782230000
3086435460
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
20:1 CRLF UTF-8 4 spaces Python 3.8 (Class)
1: Project
7: Structure
2: Favorites

Do the following:

Display the *movies* where the *Expense* is less than 30% of *Profit*

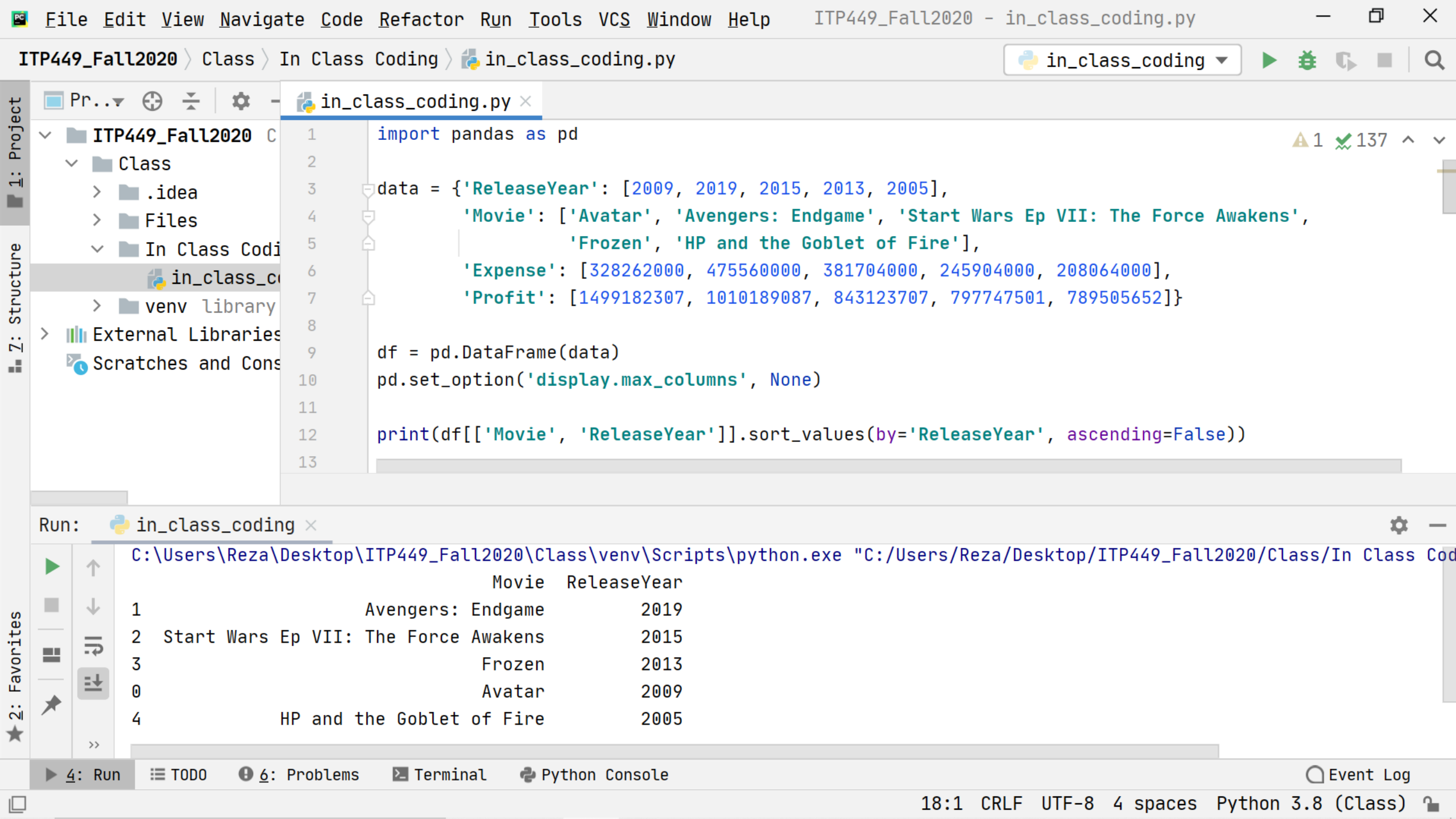
```python
import pandas as pd

data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
        'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
                  'Frozen', 'HP and the Goblet of Fire'],
        'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
        'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}

df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)

print(df.loc[df.Expense / df.Profit < 0.3, 'Movie'])
dfCheap = df[df.Expense / df.Profit < 0.3]
print(dfCheap.Movie)
```



```
0                       Avatar
4    HP and the Goblet of Fire
Name: Movie, dtype: object
0                       Avatar
4    HP and the Goblet of Fire
Name: Movie, dtype: object
```

Do the following:

Display the *Movie* and *ReleaseYear* columns sorted by the *ReleaseYear* from newest to oldest


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
1: Project
Pr..
in_class_coding.py
ITP449_Fall2020 C
Class
.idea
Files
In Class Codi
in_class_c
venv library
External Libraries
Scratches and Cons
1
137
import pandas as pd
data = {'ReleaseYear': [2009, 2019, 2015, 2013, 2005],
'Movie': ['Avatar', 'Avengers: Endgame', 'Start Wars Ep VII: The Force Awakens',
'Frozen', 'HP and the Goblet of Fire'],
'Expense': [328262000, 475560000, 381704000, 245904000, 208064000],
'Profit': [1499182307, 1010189087, 843123707, 797747501, 789505652]}
df = pd.DataFrame(data)
pd.set_option('display.max_columns', None)
print(df[['Movie', 'ReleaseYear']].sort_values(by='ReleaseYear', ascending=False))
Run: in_class_coding
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Class/In Class Cod
Movie ReleaseYear
1 Avengers: Endgame 2019
2 Start Wars Ep VII: The Force Awakens 2015
3 Frozen 2013
0 Avatar 2009
4 HP and the Goblet of Fire 2005
2: Favorites
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
18:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

NumPy and Pandas

In Action

Description

This dataset contains a list of video games with sales greater than 100,000 copies. It was generated by a scrape of vgchartz.com.

Fields include

- Rank - Ranking of overall sales
- Name - The games name
- Platform - Platform of the games release (i.e. PC,PS4, etc.)
- Year - Year of the game's release
- Genre - Genre of the game
- Publisher - Publisher of the game
- NA_Sales - Sales in North America (in millions)
- EU_Sales - Sales in Europe (in millions)
- JP_Sales - Sales in Japan (in millions)
- Other_Sales - Sales in the rest of the world (in millions)
- Global_Sales - Total worldwide sales.

The script to scrape the data is available at https://github.com/GregorUT/vgchartzScrape. It is based on BeautifulSoup using Python. There are 16,598 records. 2 records were dropped due to incomplete information.

File Edit View Navigate Code Refactor Run Tools VCS Window Help ITP449_Fall2020 - in_class_coding.py

ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py

in_class_coding

Pr...

ITP449_Fall2020 C:\
- Class
  - .idea
  - Files
  - In Class Codin
    - in_class_cod
  - venv library r
- External Libraries

in_class_coding.py

```
1  import pandas as pd
2  import os
3
4  os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
5  dfVG = pd.read_csv('vgsales.csv')
6
7  pd.set_option('display.max_columns', None)
8  print(dfVG.head())
```

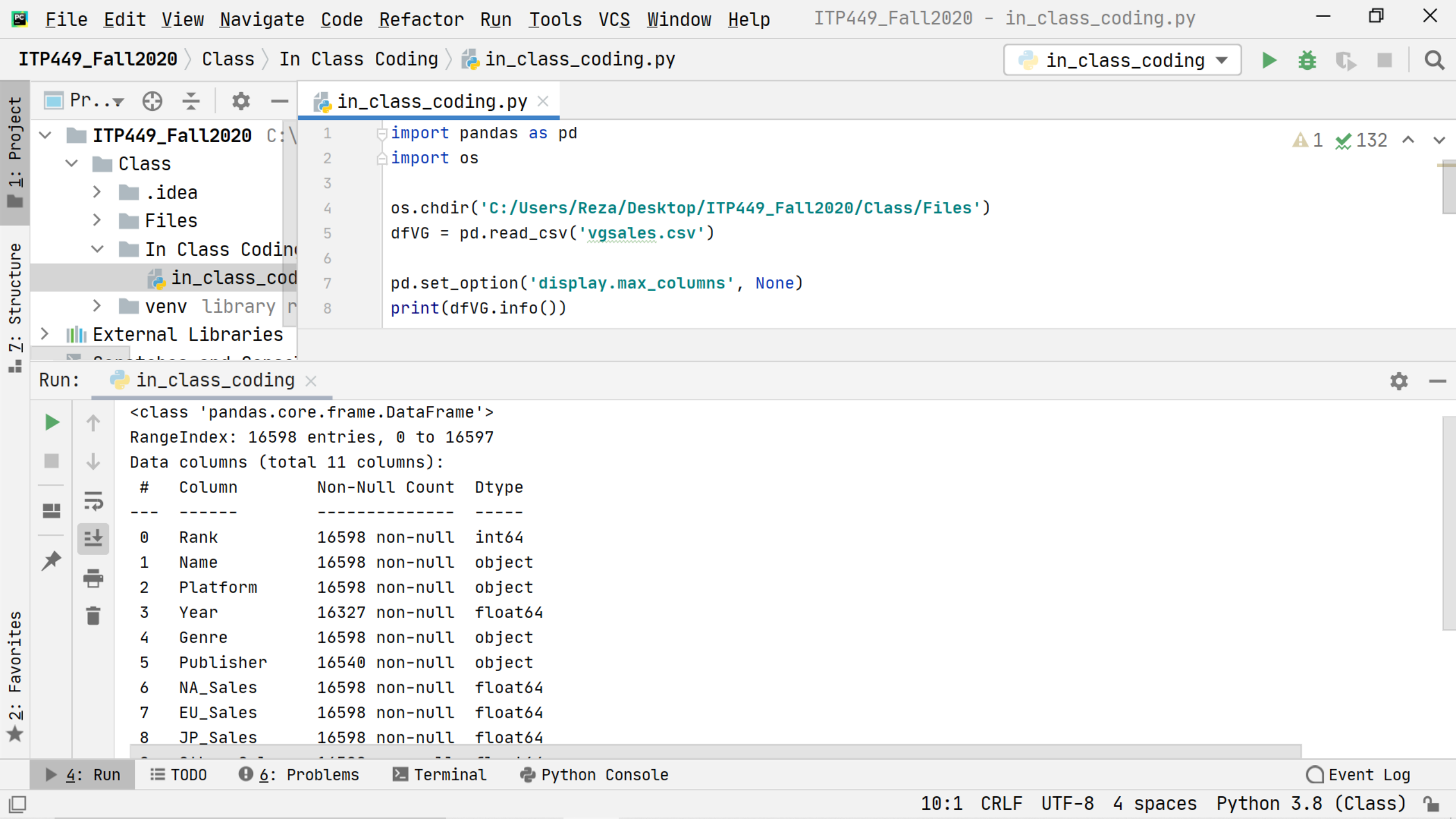
1 132

Run: in_class_coding

```
   Rank                      Name Platform    Year         Genre Publisher  \
0     1                Wii Sports      Wii  2006.0        Sports  Nintendo
1     2         Super Mario Bros.      NES  1985.0      Platform  Nintendo
2     3            Mario Kart Wii      Wii  2008.0        Racing  Nintendo
3     4         Wii Sports Resort      Wii  2009.0        Sports  Nintendo
4     5  Pokemon Red/Pokemon Blue       GB  1996.0  Role-Playing  Nintendo

   NA_Sales  EU_Sales  JP_Sales  Other_Sales  Global_Sales
0     41.49     29.02      3.77         8.46         82.74
1     29.08      3.58      6.81         0.77         40.24
2     15.85     12.88      3.79         3.31         35.82
3     15.75     11.01      3.28         2.96         33.00
4     11.27      8.89     10.22         1.00         31.37
```

4: Run TODO 6: Problems Terminal Python Console Event Log

11:1 CRLF UTF-8 4 spaces Python 3.8 (Class)


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
1: Project
Pr...
ITP449_Fall2020 C:\
Class
.idea
Files
In Class Codin
in_class_cod
venv library r
External Libraries
in_class_coding.py
1 import pandas as pd
2 import os
3
4 os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
5 dfVG = pd.read_csv('vgsales.csv')
6
7 pd.set_option('display.max_columns', None)
8 print(dfVG.info())
1
132
Run: in_class_coding
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16598 entries, 0 to 16597
Data columns (total 11 columns):
Column Non-Null Count Dtype
--- ------ -------------- -----
0 Rank 16598 non-null int64
1 Name 16598 non-null object
2 Platform 16598 non-null object
3 Year 16327 non-null float64
4 Genre 16598 non-null object
5 Publisher 16540 non-null object
6 NA_Sales 16598 non-null float64
7 EU_Sales 16598 non-null float64
8 JP_Sales 16598 non-null float64
7: Structure
2: Favorites
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
10:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

File Edit View Navigate Code Refactor Run Tools VCS Window Help ITP449_Fall2020 - in_class_coding.py

ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py

in_class_coding

Pr...

ITP449_Fall2020 C:\
- Class
  - .idea
  - Files
  - In Class Codin
    - in_class_cod
  - venv library r
- External Libraries

in_class_coding.py

```
1  import pandas as pd
2  import os
3
4  os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
5  dfVG = pd.read_csv('vgsales.csv')
6
7  pd.set_option('display.max_columns', None)
8  print(dfVG.describe())
```

1 132

Run: in_class_coding

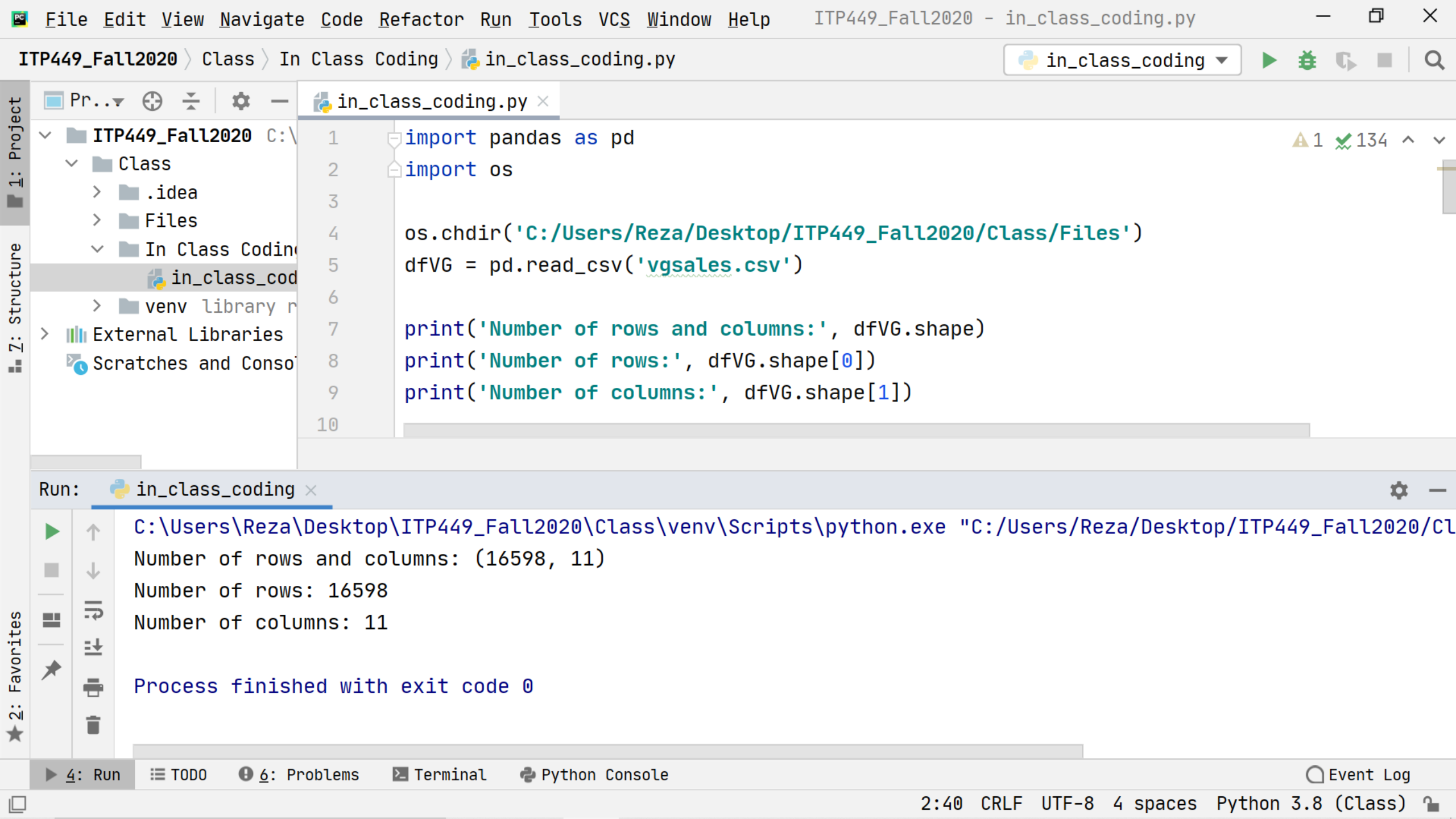
```
                Rank          Year      NA_Sales      EU_Sales      JP_Sales  \
count   16598.000000  16327.000000  16598.000000  16598.000000  16598.000000
mean     8300.605254   2006.406443      0.264667      0.146652      0.077782
std      4791.853933      5.828981      0.816683      0.505351      0.309291
min         1.000000   1980.000000      0.000000      0.000000      0.000000
25%      4151.250000   2003.000000      0.000000      0.000000      0.000000
50%      8300.500000   2007.000000      0.080000      0.020000      0.000000
75%     12449.750000   2010.000000      0.240000      0.110000      0.040000
max     16600.000000   2020.000000     41.490000     29.020000     10.220000

         Other_Sales  Global_Sales
count   16598.000000  16598.000000
mean        0.048063      0.537441
std         0.188588      1.555028
```

4: Run TODO 6: Problems Terminal Python Console Event Log

10:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

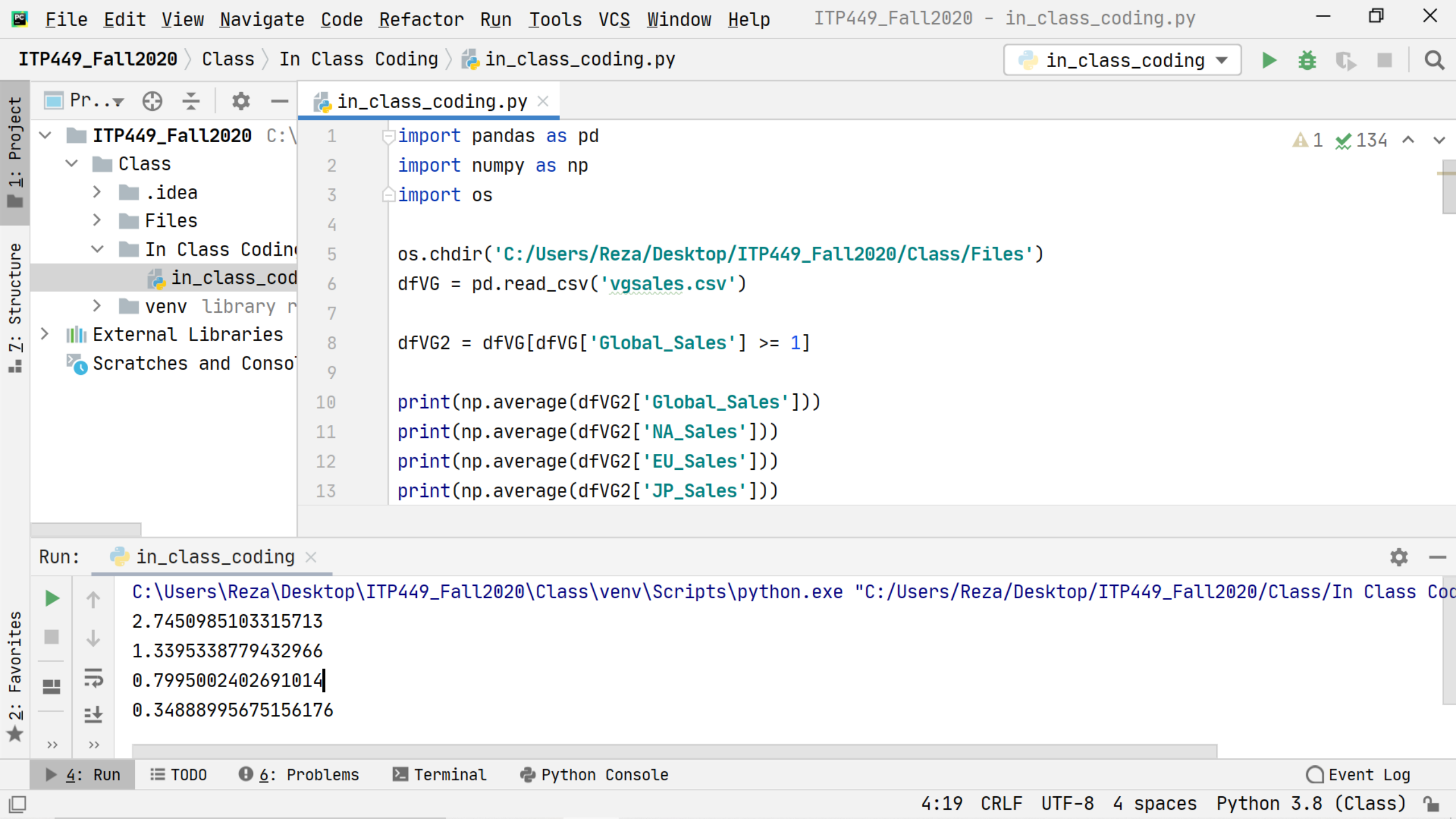
Do the following:

Print the number of rows and number of columns in the dataset.


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
1: Project
7: Structure
2: Favorites
Pr...
in_class_coding.py
ITP449_Fall2020 C:\
Class
.idea
Files
In Class Codin
in_class_cod
venv library r
External Libraries
Scratches and Conso
1 import pandas as pd
2 import os
3
4 os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
5 dfVG = pd.read_csv('vgsales.csv')
6
7 print('Number of rows and columns:', dfVG.shape)
8 print('Number of rows:', dfVG.shape[0])
9 print('Number of columns:', dfVG.shape[1])
10
1
134
Run: in_class_coding
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Cl
Number of rows and columns: (16598, 11)
Number of rows: 16598
Number of columns: 11
Process finished with exit code 0
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
2:40 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

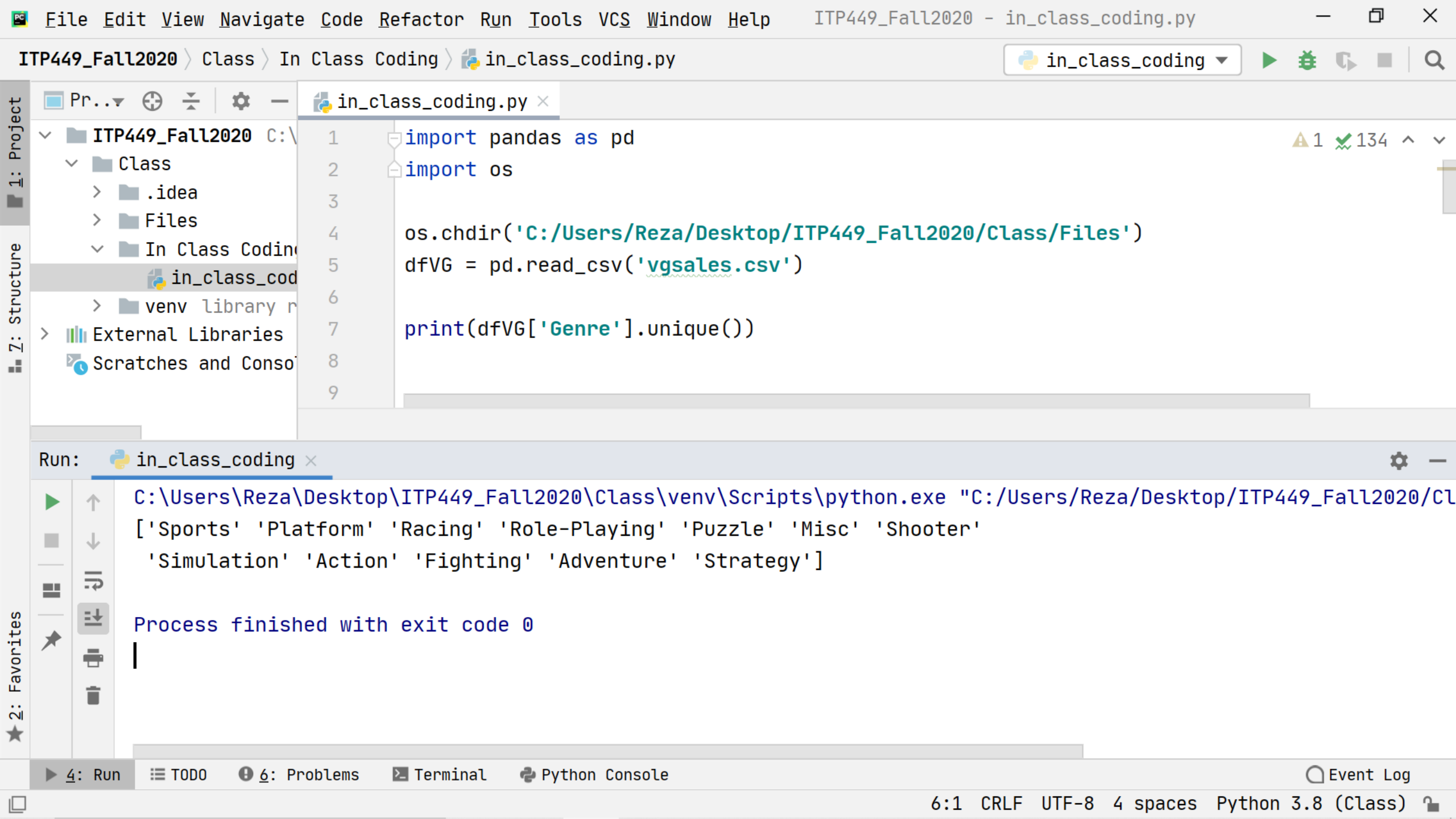
Print the average sales for Global, North America, European Union and Japan for games with global sales of at least 1 million.


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
Pr..
in_class_coding.py
ITP449_Fall2020 C:\
Class
.idea
Files
In Class Codin
in_class_cod
venv library r
External Libraries
Scratches and Conso
1 import pandas as pd
2 import numpy as np
3 import os
4
5 os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
6 dfVG = pd.read_csv('vgsales.csv')
7
8 dfVG2 = dfVG[dfVG['Global_Sales'] >= 1]
9
10 print(np.average(dfVG2['Global_Sales']))
11 print(np.average(dfVG2['NA_Sales']))
12 print(np.average(dfVG2['EU_Sales']))
13 print(np.average(dfVG2['JP_Sales']))
1
134
Run: in_class_coding
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Class/In Class Coo
2.7450985103315713
1.3395338779432966
0.7995002402691014
0.34888995675156176
1: Project
7: Structure
2: Favorites
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
4:19 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

Print all the game genres.

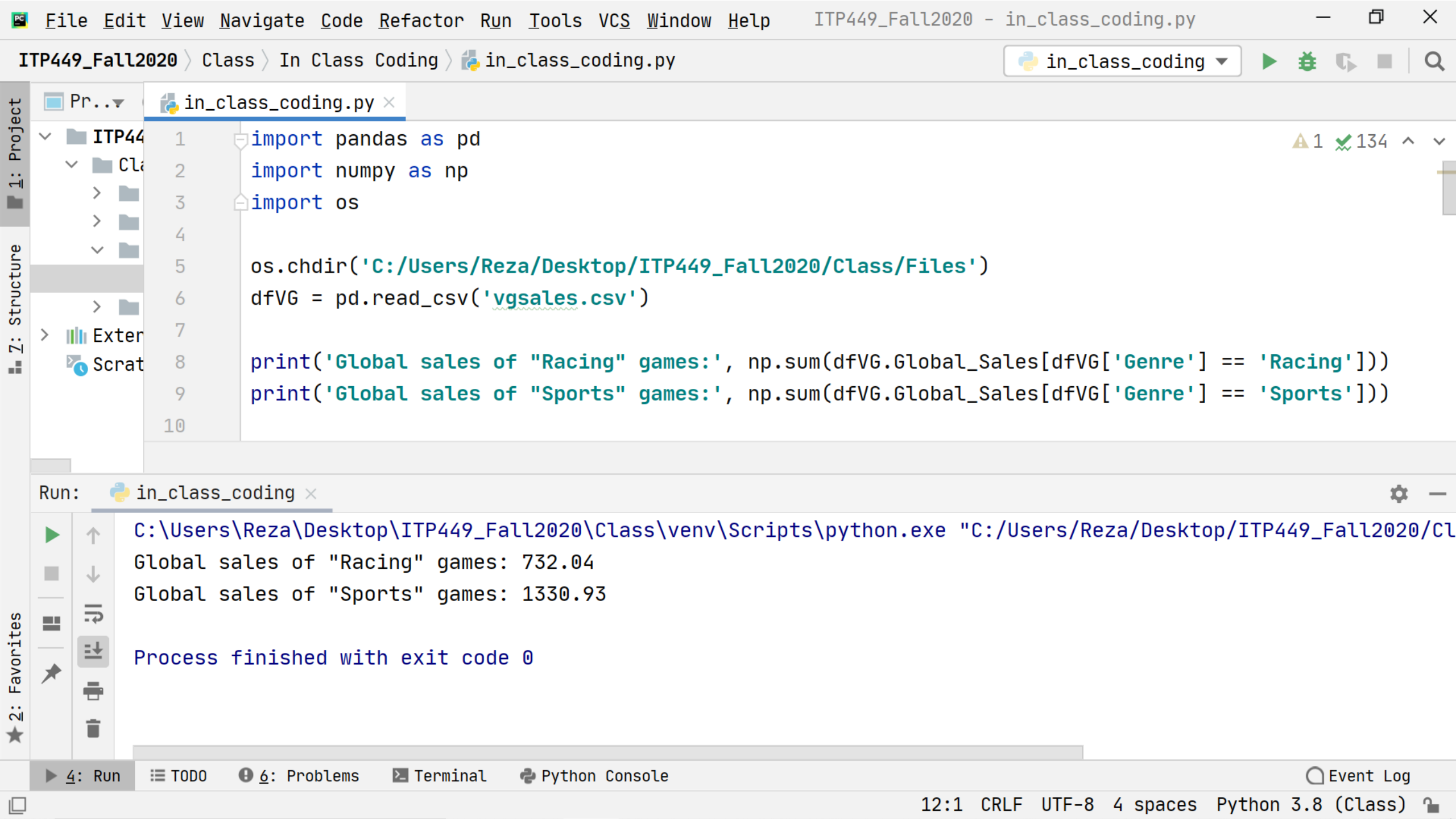
```
['Sports' 'Platform' 'Racing' 'Role-Playing' 'Puzzle' 'Misc' 'Shooter'
 'Simulation' 'Action' 'Fighting' 'Adventure' 'Strategy']
```


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
Pr...
in_class_coding.py
ITP449_Fall2020 C:\
Class
.idea
Files
In Class Codin
in_class_cod
venv library r
External Libraries
Scratches and Conso
1 import pandas as pd
2 import os
3
4 os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
5 dfVG = pd.read_csv('vgsales.csv')
6
7 print(dfVG['Genre'].unique())
8
9
1
134
Run: in_class_coding
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Cl
['Sports' 'Platform' 'Racing' 'Role-Playing' 'Puzzle' 'Misc' 'Shooter'
'Simulation' 'Action' 'Fighting' 'Adventure' 'Strategy']
Process finished with exit code 0
1: Project
7: Structure
2: Favorites
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
6:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

Print the total global sales for games of the Racing and Sports genres.

```
Global sales of Racing games: 732.04
Global sales of Sports games: 1330.93
```


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
in_class_coding.py
1 import pandas as pd
2 import numpy as np
3 import os
4
5 os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
6 dfVG = pd.read_csv('vgsales.csv')
7
8 print('Global sales of "Racing" games:', np.sum(dfVG.Global_Sales[dfVG['Genre'] == 'Racing']))
9 print('Global sales of "Sports" games:', np.sum(dfVG.Global_Sales[dfVG['Genre'] == 'Sports']))
10
1
134
Run: in_class_coding
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Cl
Global sales of "Racing" games: 732.04
Global sales of "Sports" games: 1330.93
Process finished with exit code 0
1: Project
7: Structure
2: Favorites
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
12:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

Print the top five highest grossing games in North America:

	Name	Publisher	NA_Sales
0	Wii Sports	Nintendo	41.49
1	Super Mario Bros.	Nintendo	29.08
9	Duck Hunt	Nintendo	26.93
5	Tetris	Nintendo	23.20
2	Mario Kart Wii	Nintendo	15.85

File Edit View Navigate Code Refactor Run Tools VCS Window Help ITP449_Fall2020 - in_class_coding.py

ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py

in_class_coding

in_class_coding.py

```
1  import pandas as pd
2  import os
3
4  os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
5  dfVG = pd.read_csv('vgsales.csv')
6
7  print(dfVG[['Name', 'Publisher', 'NA_Sales']].sort_values(by='NA_Sales', ascending=False).head())
8
```

1 134

Run: in_class_coding

```
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Cl
                Name Publisher  NA_Sales
0         Wii Sports  Nintendo     41.49
1  Super Mario Bros.  Nintendo     29.08
9          Duck Hunt  Nintendo     26.93
5             Tetris  Nintendo     23.20
2     Mario Kart Wii  Nintendo     15.85

Process finished with exit code 0
```

1: Project 7: Structure 2: Favorites

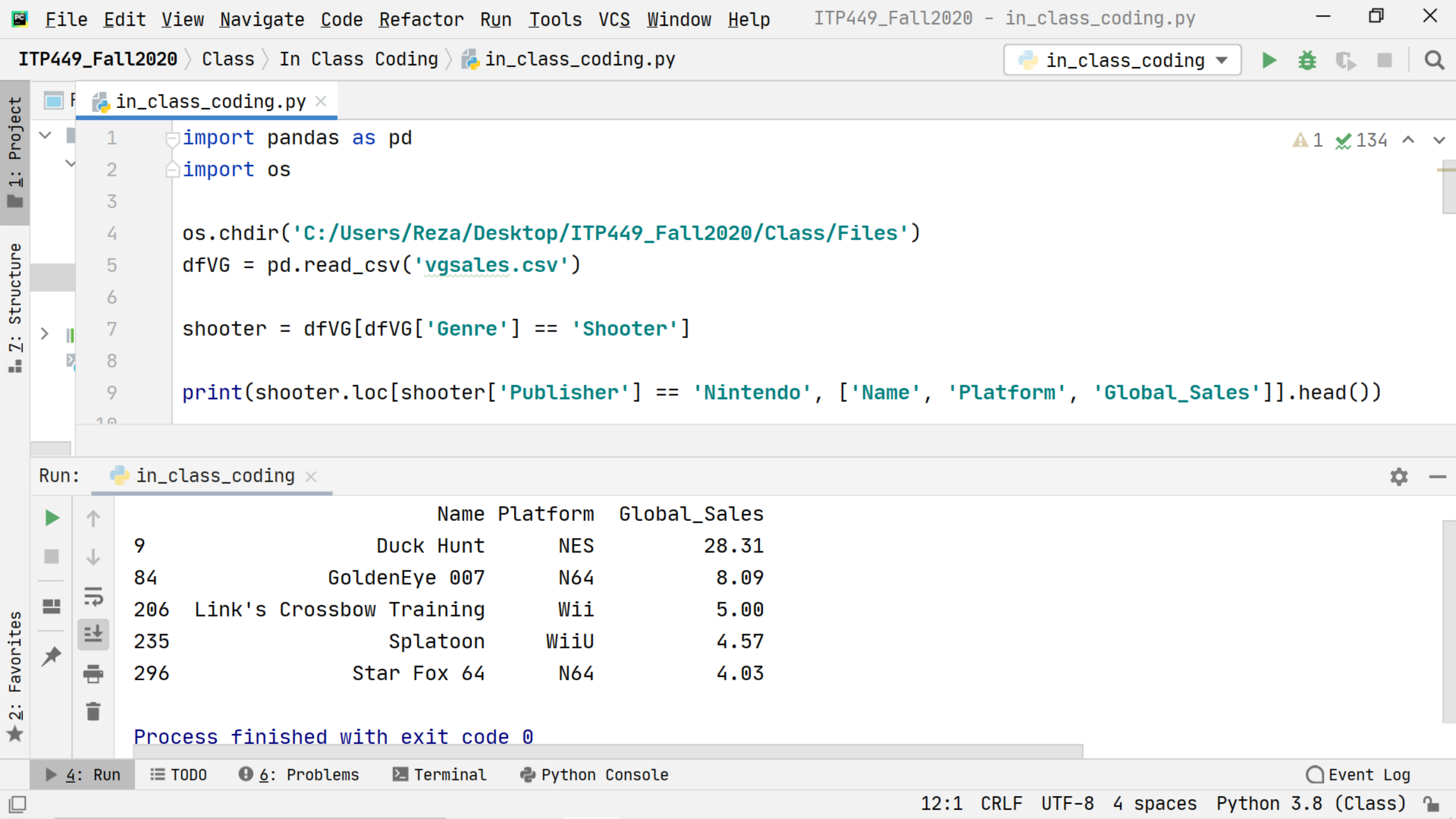
4: Run TODO 6: Problems Terminal Python Console Event Log

11:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

Print the top five Shooter games from Nintendo.

	Name	Platform	Global_Sales
9	Duck Hunt	NES	28.31
84	GoldenEye 007	N64	8.09
206	Link's Crossbow Training	Wii	5.00
235	Splatoon	WiiU	4.57
296	Star Fox 64	N64	4.03


File Edit View Navigate Code Refactor Run Tools VCS Window Help
ITP449_Fall2020 - in_class_coding.py
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py
in_class_coding
in_class_coding.py
1 import pandas as pd
2 import os
3
4 os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
5 dfVG = pd.read_csv('vgsales.csv')
6
7 shooter = dfVG[dfVG['Genre'] == 'Shooter']
8
9 print(shooter.loc[shooter['Publisher'] == 'Nintendo', ['Name', 'Platform', 'Global_Sales']].head())
1
134
Run: in_class_coding
Name Platform Global_Sales
9 Duck Hunt NES 28.31
84 GoldenEye 007 N64 8.09
206 Link's Crossbow Training Wii 5.00
235 Splatoon WiiU 4.57
296 Star Fox 64 N64 4.03
Process finished with exit code 0
1: Project
7: Structure
2: Favorites
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
12:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

Print all the sales totals for each genre.

Genre	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
Action	877.83	525.00	159.95	187.38	1751.18
Adventure	105.80	64.13	52.07	16.81	239.04
Fighting	223.59	101.32	87.35	36.68	448.91
Misc	410.24	215.98	107.76	75.32	809.96
Platform	447.05	201.63	130.77	51.59	831.37
Puzzle	123.78	50.78	57.31	12.55	244.95
Racing	359.42	238.39	56.69	77.27	732.04
Role-Playing	327.28	188.06	352.31	59.61	927.37
Shooter	582.60	313.27	38.28	102.69	1037.37
Simulation	183.31	113.38	63.70	31.52	392.20
Sports	683.35	376.85	135.37	134.97	1330.93
Strategy	68.70	45.34	49.46	11.36	175.12

File Edit View Navigate Code Refactor Run Tools VCS Window Help ITP449_Fall2020 - in_class_coding.py

ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py

in_class_coding

in_class_coding.py

```
1  import pandas as pd
2  import os
3
4  os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
5  dfVG = pd.read_csv('vgsales.csv')
6
7  print(dfVG.groupby(['Genre']).sum().loc[:, 'NA_Sales':'Global_Sales'])
8
```

Run: in_class_coding

```
              NA_Sales  EU_Sales  JP_Sales  Other_Sales  Global_Sales
Genre
Action          877.83    525.00    159.95       187.38       1751.18
Adventure       105.80     64.13     52.07        16.81        239.04
Fighting        223.59    101.32     87.35        36.68        448.91
Misc            410.24    215.98    107.76        75.32        809.96
Platform        447.05    201.63    130.77        51.59        831.37
Puzzle          123.78     50.78     57.31        12.55        244.95
Racing          359.42    238.39     56.69        77.27        732.04
Role-Playing    327.28    188.06    352.31        59.61        927.37
Shooter         582.60    313.27     38.28       102.69       1037.37
Simulation      183.31    113.38     63.70        31.52        392.20
Sports          683.35    376.85    135.37       134.97       1330.93
Strategy         68.70     45.34     49.46        11.36        175.12
```

4: Run TODO 6: Problems Terminal Python Console Event Log

18:1 CRLF UTF-8 4 spaces Python 3.8 (Class)