ITP 449

Time Series

Lecture 5


citi bike

Citi Bike Trip Histories

We publish downloadable files of Citi Bike trip data. The data includes:

- Trip Duration (seconds)
- Start Time and Date
- Stop Time and Date
- Start Station Name
- End Station Name
- Station ID
- Station Lat/Long
- Bike ID
- User Type (Customer = 24-hour pass or 3-day pass user; Subscriber = Annual Member)
- Gender (Zero=unknown; 1=male; 2=female)
- Year of Birth

This data has been processed to remove trips that are taken by staff as they service and inspect the system, trips that are taken to/from any of our “test” stations (which we were using more in June and July 2013), and any trips that were below 60 seconds in length (potentially false starts or users trying to re-dock a bike to ensure it's secure).

➔ Download Citi Bike trip history data

https://s3.amazonaws.com/tripdata/index.html

Do the following:

Create two *DataFrame* variables containing the CitiBike data from the 02/18 and 07/18 CSV files.

File Edit View Navigate Code Refactor Run Tools VCS Window

ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py

Pr.. in_class_coding.py

Run: in_class_coding

```
   tripduration                starttime                 stoptime  \
0           771  2018-02-01 00:14:16.4120  2018-02-01 00:27:08.2290
1           264  2018-02-01 05:14:45.1790  2018-02-01 05:19:09.6860
2           819  2018-02-01 06:48:55.2290  2018-02-01 07:02:35.0290
3           646  2018-02-01 07:12:50.1740  2018-02-01 07:23:36.5280
4          1312  2018-02-01 07:46:48.8750  2018-02-01 08:08:41.5430

   start station id start station name  start station latitude  \
0                72   W 52 St & 11 Ave               40.767272
1                72   W 52 St & 11 Ave               40.767272
2                72   W 52 St & 11 Ave               40.767272
3                72   W 52 St & 11 Ave               40.767272
4                72   W 52 St & 11 Ave               40.767272

   start station longitude  end station id              end station name  \
0               -73.993929             379               W 31 St & 7 Ave
1               -73.993929             478              11 Ave & W 41 St
2               -73.993929             405  Washington St & Gansevoort St
3               -73.993929            2006         Central Park S & 6 Ave
4               -73.993929             435               W 21 St & 6 Ave

   end station latitude  end station longitude  bikeid    usertype  \
0             40.749156             -73.991600   14536  Subscriber
1             40.760301             -73.998842   32820  Subscriber
2             40.739323             -74.008119   16131  Subscriber
3             40.765909             -73.976342   20831  Subscriber
4             40.741740             -73.994156   15899  Subscriber

   birth year  gender
0        1952       1
1        1965       1
2        1968       1
3        1990       2
4        1957       1
```

4: Run TODO 6: Problems Terminal Python Console

File Edit View Navigate Code Refactor Run Tools VCS Window

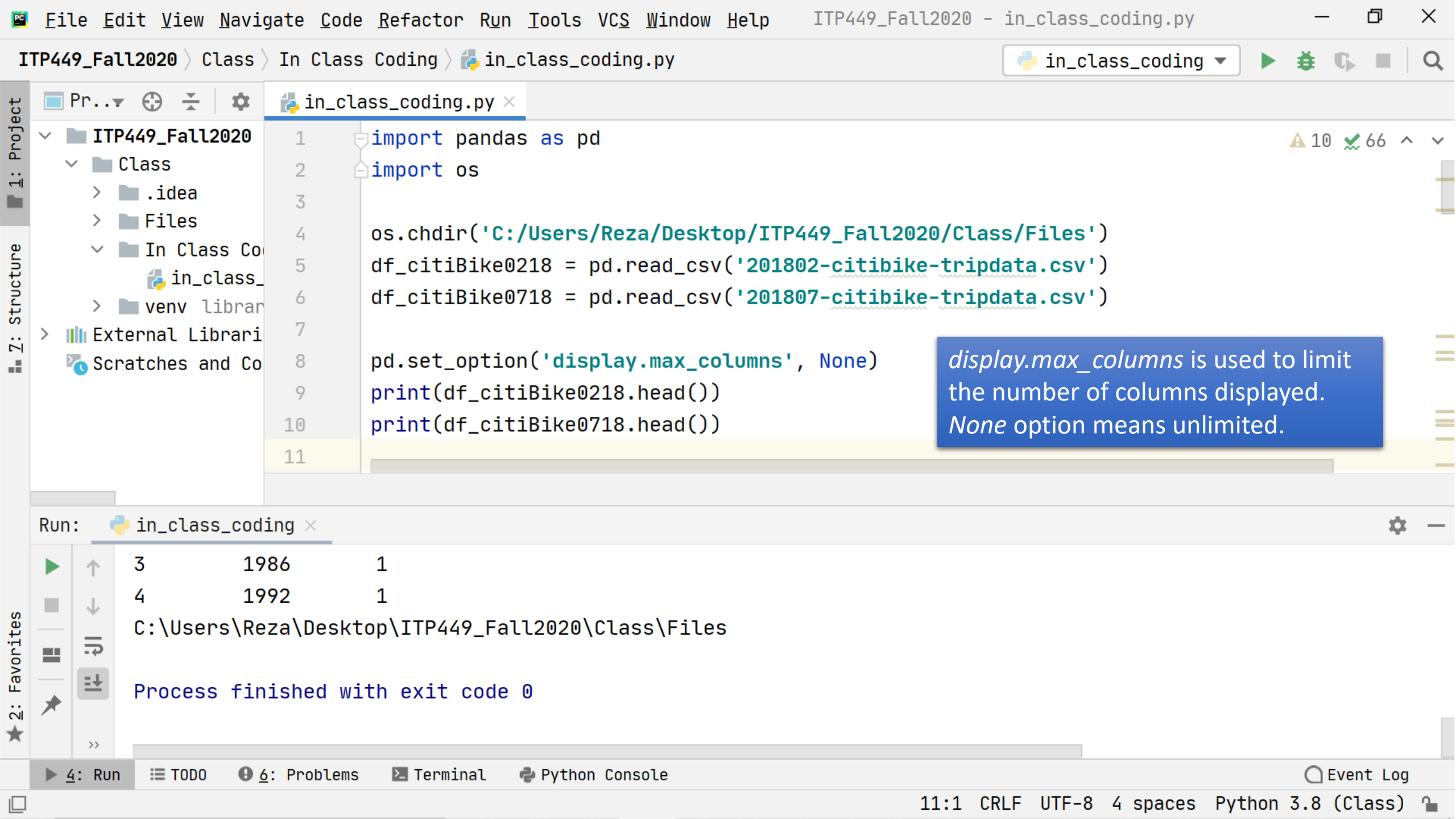
ITP449_Fall2020 〉 Class 〉 In Class Coding 〉 in_class_coding.py

Pr.. in_class_coding.py

Run: in_class_coding

```
   tripduration                starttime                 stoptime  \
0           500  2018-07-01 00:33:51.2640  2018-07-01 00:42:12.0280
1           455  2018-07-01 02:06:54.0270  2018-07-01 02:14:29.0900
2          1080  2018-07-01 02:09:16.4640  2018-07-01 02:27:17.0290
3           632  2018-07-01 02:55:04.5870  2018-07-01 03:05:36.7910
4          1676  2018-07-01 03:24:27.4350  2018-07-01 03:52:23.7760

   start station id start station name  start station latitude  \
0                72   W 52 St & 11 Ave               40.767272
1                72   W 52 St & 11 Ave               40.767272
2                72   W 52 St & 11 Ave               40.767272
3                72   W 52 St & 11 Ave               40.767272
4                72   W 52 St & 11 Ave               40.767272

   start station longitude  end station id            end station name  \
0               -73.993929            3236  W 42 St & Dyer Ave
1               -73.993929             529     W 42 St & 8 Ave
2               -73.993929             453     W 22 St & 8 Ave
3               -73.993929             465  Broadway & W 41 St
4               -73.993929            3425    2 Ave & E 104 St

   end station latitude  end station longitude  bikeid    usertype  \
0             40.758985             -73.993800   16583  Subscriber
1             40.757570             -73.990985   31481  Subscriber
2             40.744751             -73.999154   25004  Subscriber
3             40.755136             -73.986580   25867  Subscriber
4             40.789210             -73.943708   31351  Subscriber

   birth year  gender
0        1981       1
1        1988       1
2        1982       2
3        1986       1
4        1992       1
```

4: Run TODO 6: Problems Terminal Python Console


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 Co
in_class_
venv librar
External Librari
Scratches and Co
1 import pandas as pd
2 import os
3
4 os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
5 df_citiBike0218 = pd.read_csv('201802-citibike-tripdata.csv')
6 df_citiBike0718 = pd.read_csv('201807-citibike-tripdata.csv')
7
8 pd.set_option('display.max_columns', None)
9 print(df_citiBike0218.head())
10 print(df_citiBike0718.head())
11
10
66
display.max_columns is used to limit the number of columns displayed. None option means unlimited.
Run: in_class_coding
3 1986 1
4 1992 1
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\Files
Process finished with exit code 0
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 two ndarray variables for trip duration (in minutes) from 02/18 and 07/18.

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
- Class
  - .idea
  - Files
  - In Class Co
    - in_class_
  - venv librar
- External Librari
- Scratches and Co

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  df_citiBike0218 = pd.read_csv('201802-citibike-tripdata.csv')
6  df_citiBike0718 = pd.read_csv('201807-citibike-tripdata.csv')
7
8  arTripDuration0218 = df_citiBike0218['tripduration'].values/60
9  arTripDuration0718 = df_citiBike0718['tripduration'].values/60
10
11
12
13
14
```

11 68

Dataframe['column'].values returns the values in that column so that they can be assigned to an array.

7: Structure

Run: in_class_coding

```
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Cl

Process finished with exit code 0
```

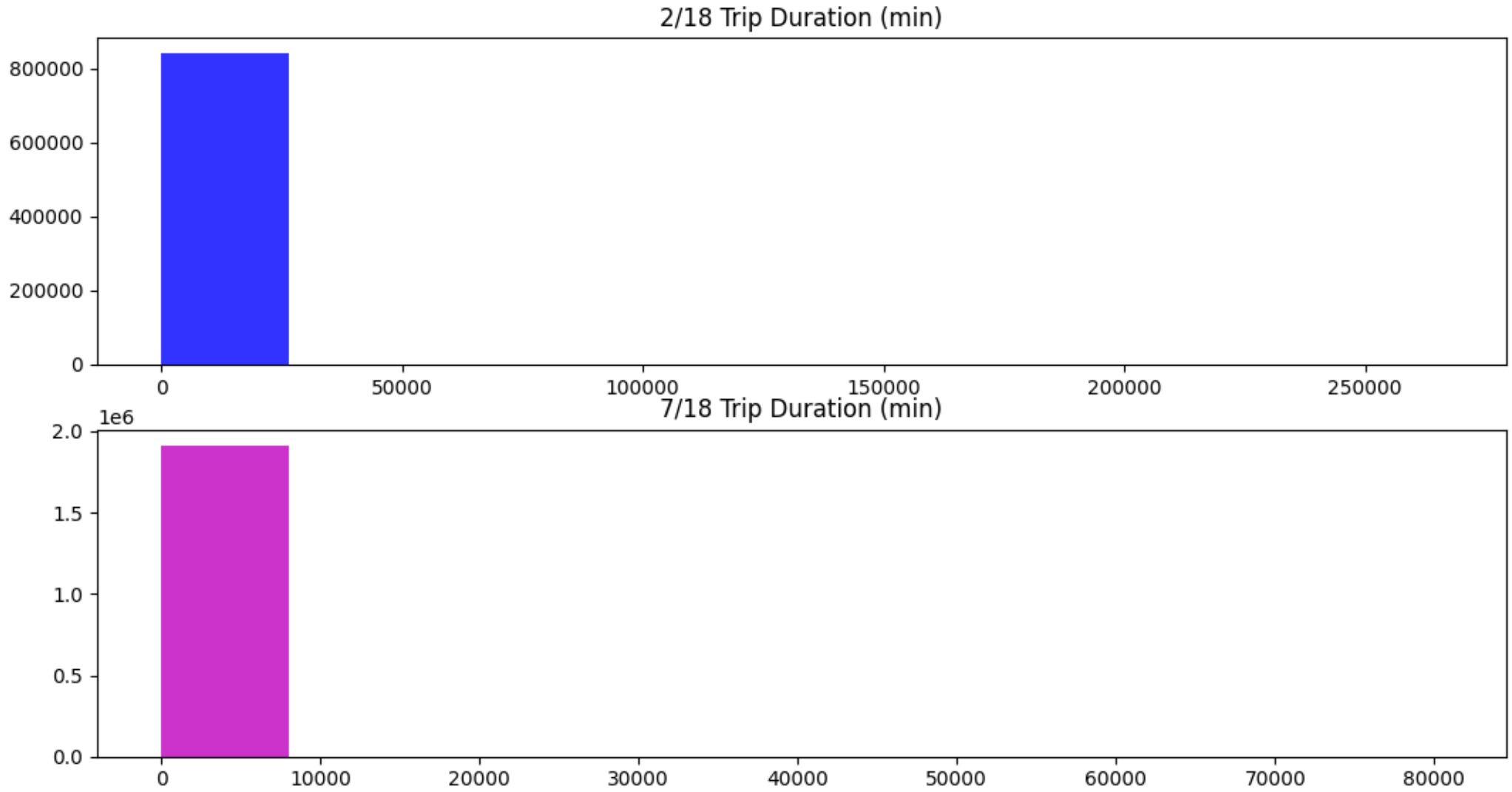
2: Favorites

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

16:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

Create a figure of 2 x 1 subplots that displays the histograms (distribution) of the trip durations from 02/18 and 07/18.


2/18 Trip Duration (min)
800000
600000
400000
200000
0
0
50000
100000
150000
200000
250000
7/18 Trip Duration (min)
1e6
2.0
1.5
1.0
0.5
0.0
0
10000
20000
30000
40000
50000
60000
70000
80000

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 Co
            in_class_
        venv  librar
    External Librari
    Scratches and Co
```

```
1  import pandas as pd
2  import os
3  import matplotlib.pyplot as plt
4
5  os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
6  df_citiBike0218 = pd.read_csv('201802-citibike-tripdata.csv')
7  df_citiBike0718 = pd.read_csv('201807-citibike-tripdata.csv')
8
9  arTripDuration0218 = df_citiBike0218['tripduration'].values/60
10 arTripDuration0718 = df_citiBike0718['tripduration'].values/60
11
12 myFig = plt.figure()
13 ax1 = myFig.add_subplot(2, 1, 1)
14 ax2 = myFig.add_subplot(2, 1, 2)
15
16 ax1.hist(arTripDuration0218, color='b', alpha=0.8)
17 ax1.set_title('2/18 Trip Duration (min)')
18
19 ax2.hist(arTripDuration0718, color='m', alpha=0.8)
20 ax2.set_title('7/18 Trip Duration (min)')
21
22 plt.show()
```

11 68

Run: in_class_coding

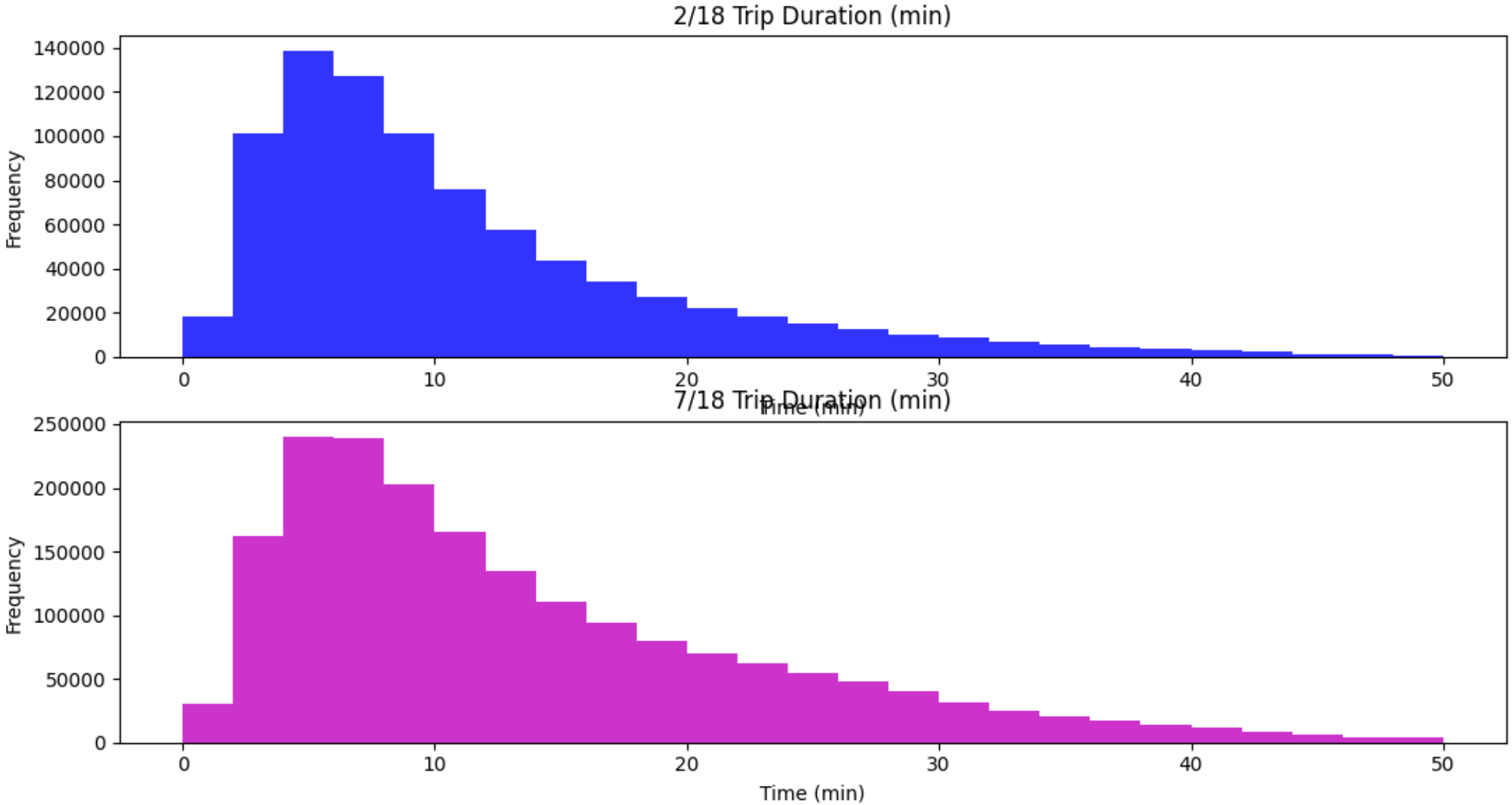
1: Project 7: Structure 2: Favorites

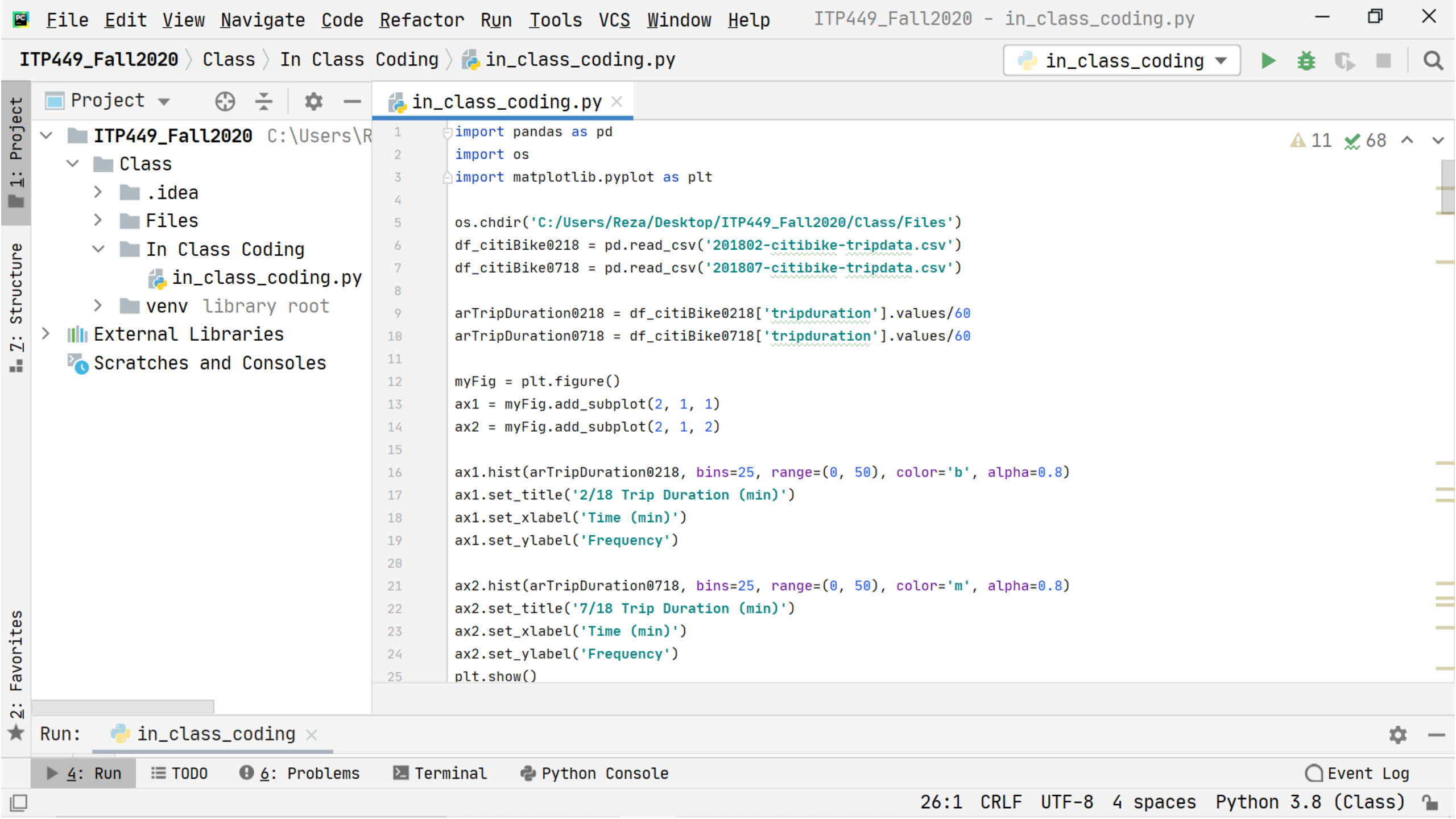
Type here to search

ENG 8:52 PM 9/22/2020

Do the following:

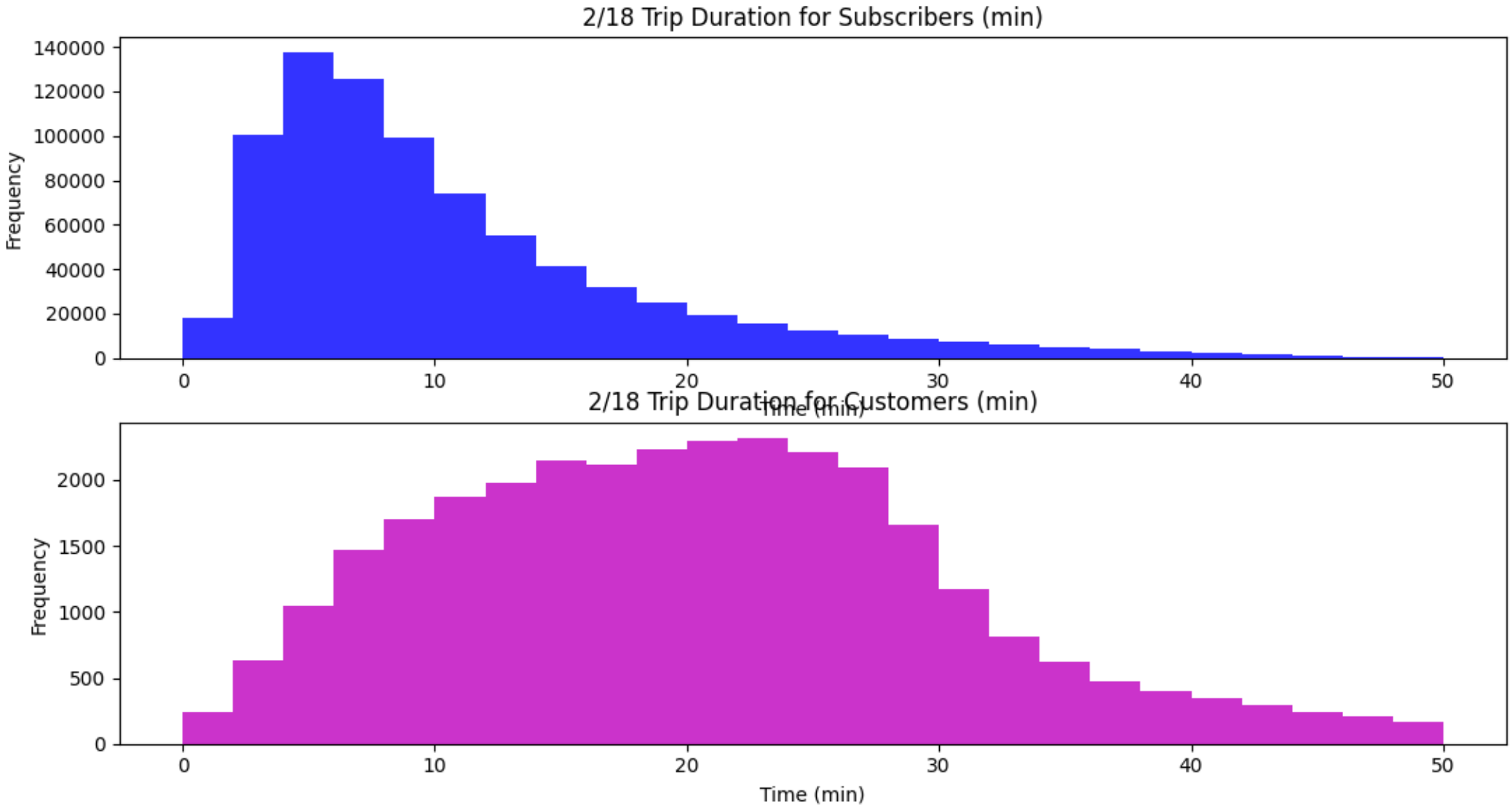
Scale the histograms to display more details (more bins).


2/18 Trip Duration (min)
Frequency
140000
120000
100000
80000
60000
40000
20000
0
0
10
20
30
40
50
Time (min)
7/18 Trip Duration (min)
Frequency
250000
200000
150000
100000
50000
0
0
10
20
30
40
50
Time (min)


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
Project
in_class_coding.py
ITP449_Fall2020 C:\Users\R
Class
.idea
Files
In Class Coding
in_class_coding.py
venv library root
External Libraries
Scratches and Consoles
1: Project
7: Structure
2: Favorites
11 68
import pandas as pd
import os
import matplotlib.pyplot as plt
os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
df_citiBike0218 = pd.read_csv('201802-citibike-tripdata.csv')
df_citiBike0718 = pd.read_csv('201807-citibike-tripdata.csv')
arTripDuration0218 = df_citiBike0218['tripduration'].values/60
arTripDuration0718 = df_citiBike0718['tripduration'].values/60
myFig = plt.figure()
ax1 = myFig.add_subplot(2, 1, 1)
ax2 = myFig.add_subplot(2, 1, 2)
ax1.hist(arTripDuration0218, bins=25, range=(0, 50), color='b', alpha=0.8)
ax1.set_title('2/18 Trip Duration (min)')
ax1.set_xlabel('Time (min)')
ax1.set_ylabel('Frequency')
ax2.hist(arTripDuration0718, bins=25, range=(0, 50), color='m', alpha=0.8)
ax2.set_title('7/18 Trip Duration (min)')
ax2.set_xlabel('Time (min)')
ax2.set_ylabel('Frequency')
plt.show()
Run: in_class_coding
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
26:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Do the following:

Create a figure of 2 x 1 subplots that displays the histograms (distribution) of the trip durations for subscribers and customers from 02/18.

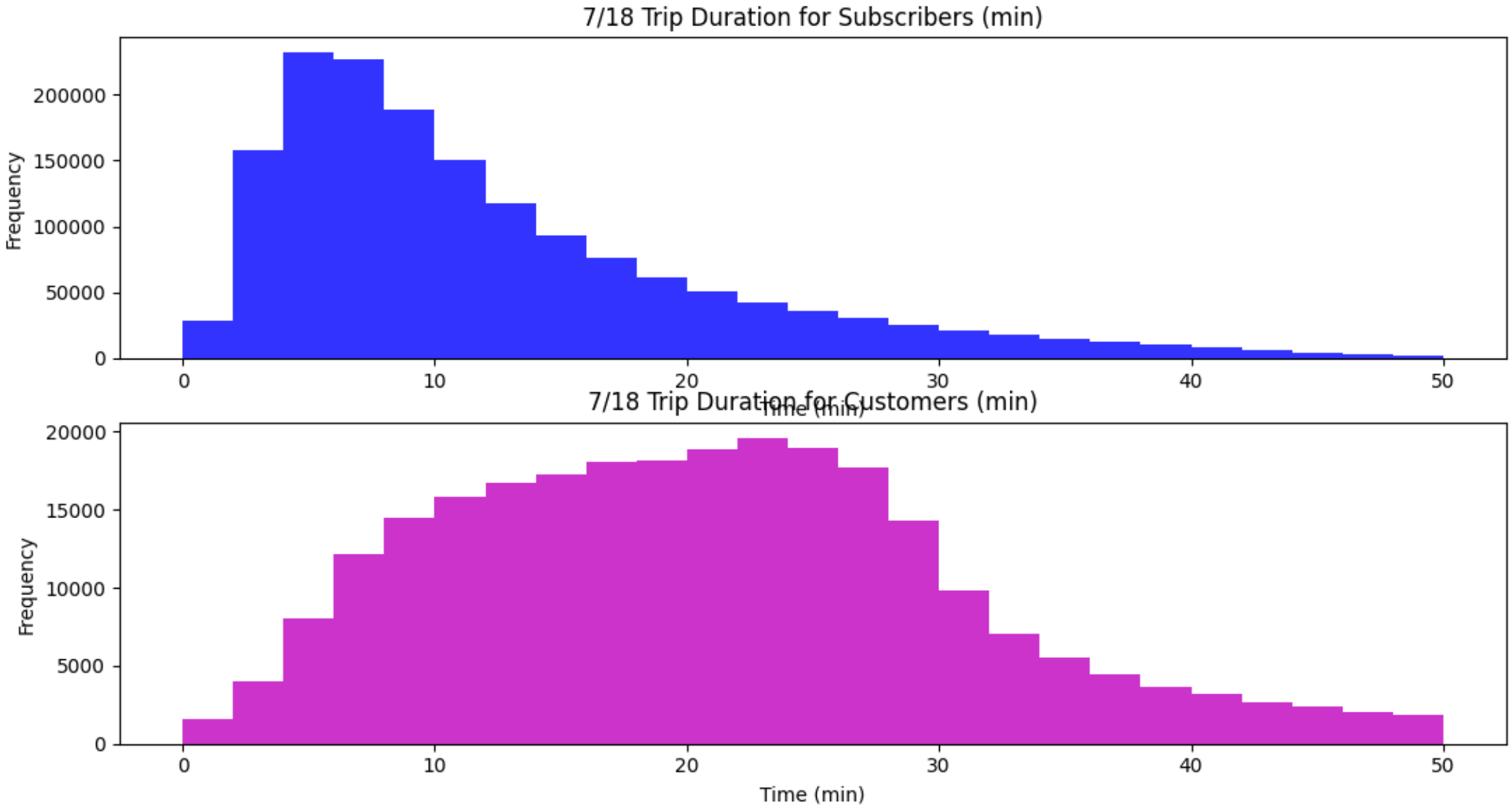

2/18 Trip Duration for Subscribers (min)
Frequency
140000
120000
100000
80000
60000
40000
20000
0
0
10
20
30
40
50
Time (min)
2/18 Trip Duration for Customers (min)
Frequency
2000
1500
1000
500
0
0
10
20
30
40
50
Time (min)

in_class_coding.py

```
os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
df_citiBike0218 = pd.read_csv('201802-citibike-tripdata.csv')

arTripDuration0218 = df_citiBike0218['tripduration'].values/60
arTripUserType0218 = df_citiBike0218['usertype'].values

arTripDuration0218Sub = arTripDuration0218[arTripUserType0218 == 'Subscriber']
arTripDuration0218Cus = arTripDuration0218[arTripUserType0218 == 'Customer']

myFig = plt.figure()
ax1 = myFig.add_subplot(2, 1, 1)
ax2 = myFig.add_subplot(2, 1, 2)

ax1.hist(arTripDuration0218Sub, bins=25, range=(0, 50), color='b', alpha=0.8)
ax1.set_title('2/18 Trip Duration for Subscribers (min)')
ax1.set_xlabel('Time (min)')
ax1.set_ylabel('Frequency')

ax2.hist(arTripDuration0218Cus, bins=25, range=(0, 50), color='m', alpha=0.8)
ax2.set_title('2/18 Trip Duration for Customers (min)')
ax2.set_xlabel('Time (min)')
ax2.set_ylabel('Frequency')
plt.show()
```

Do the following:

Create a figure of 2 x 1 subplots that displays the histograms (distribution) of the trip durations for subscribers and customers from 07/18.


7/18 Trip Duration for Subscribers (min)
Frequency
200000
150000
100000
50000
0
0
10
20
30
40
50
Time (min)
7/18 Trip Duration for Customers (min)
Frequency
20000
15000
10000
5000
0
0
10
20
30
40
50
Time (min)

Pr...

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

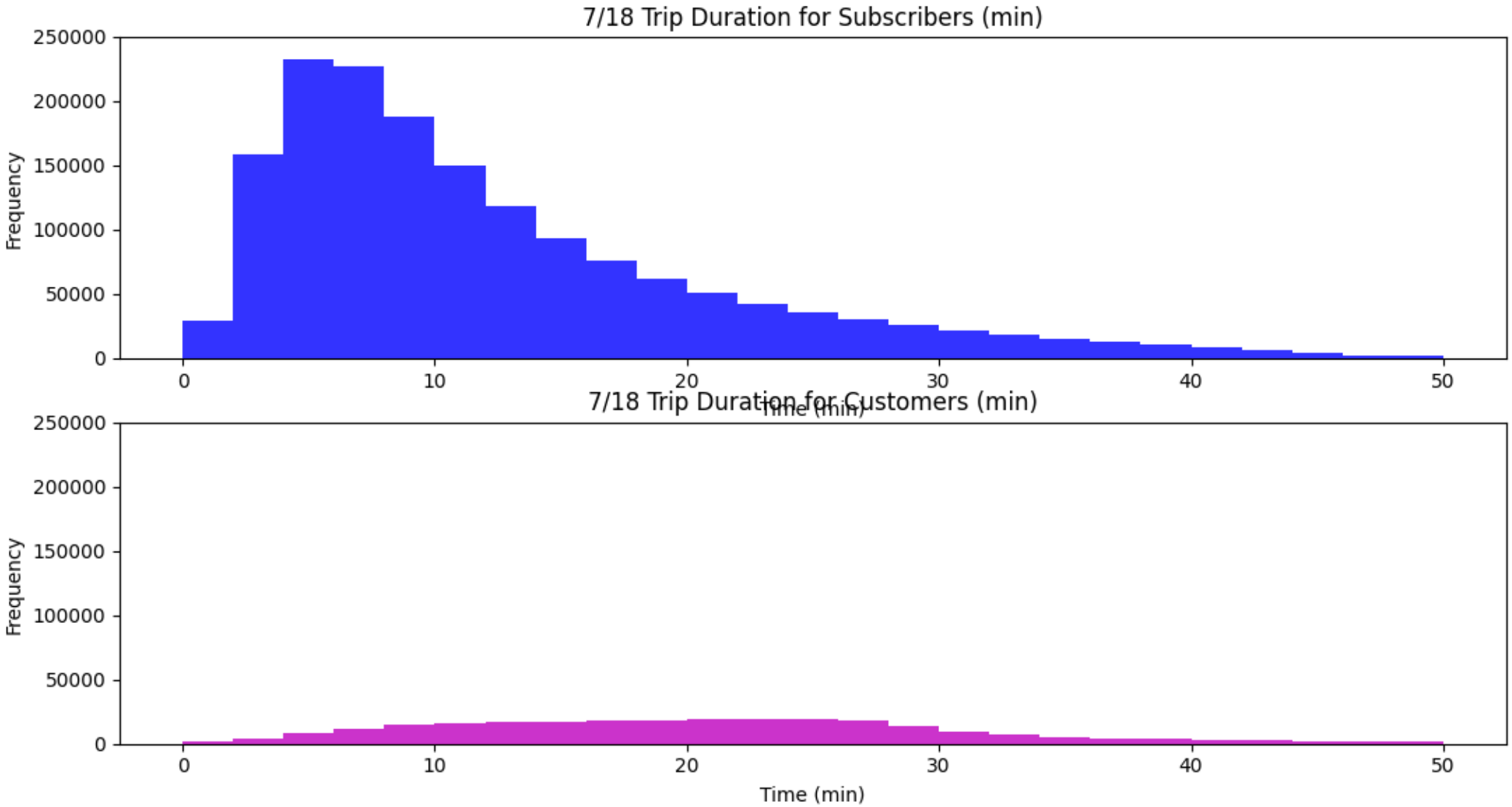
in_class_coding.py

11 84

```
os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
df_citiBike0718 = pd.read_csv('201807-citibike-tripdata.csv')

arTripDuration0718 = df_citiBike0718['tripduration'].values/60
arTripUserType0718 = df_citiBike0718['usertype'].values

arTripDuration0718Sub = arTripDuration0718[arTripUserType0718 == 'Subscriber']
arTripDuration0718Cus = arTripDuration0718[arTripUserType0718 == 'Customer']

myFig = plt.figure()
ax1 = myFig.add_subplot(2, 1, 1)
ax2 = myFig.add_subplot(2, 1, 2)

ax1.hist(arTripDuration0718Sub, bins=25, range=(0, 50), color='b', alpha=0.8)
ax1.set_title('7/18 Trip Duration for Subscribers (min)')
ax1.set_xlabel('Time (min)')
ax1.set_ylabel('Frequency')

ax2.hist(arTripDuration0718Cus, bins=25, range=(0, 50), color='m', alpha=0.8)
ax2.set_title('7/18 Trip Duration for Customers (min)')
ax2.set_xlabel('Time (min)')
ax2.set_ylabel('Frequency')
plt.show()
```

Do the following:

Scale the 07/18 histograms so that the y axis range is from 0 to 250,000 for both plots.


7/18 Trip Duration for Subscribers (min)
Frequency
250000
200000
150000
100000
50000
0
0
10
20
30
40
50
Time (min)
7/18 Trip Duration for Customers (min)
Frequency
250000
200000
150000
100000
50000
0
0
10
20
30
40
50
Time (min)

1: Project
7: Structure
2: Favorites

Pr...

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

in_class_coding.py

```
os.chdir('C:/Users/Reza/Desktop/ITP449_Fall2020/Class/Files')
df_citiBike0718 = pd.read_csv('201807-citibike-tripdata.csv')

arTripDuration0718 = df_citiBike0718['tripduration'].values/60
arTripUserType0718 = df_citiBike0718['usertype'].values

arTripDuration0718Sub = arTripDuration0718[arTripUserType0718 == 'Subscriber']
arTripDuration0718Cus = arTripDuration0718[arTripUserType0718 == 'Customer']

myFig = plt.figure()
ax1 = myFig.add_subplot(2, 1, 1)
ax2 = myFig.add_subplot(2, 1, 2)

ax1.hist(arTripDuration0718Sub, bins=25, range=(0, 50), color='b', alpha=0.8)
ax1.set_title('7/18 Trip Duration for Subscribers (min)')
ax1.set_xlabel('Time (min)')
ax1.set_ylabel('Frequency')
ax1.set_ylim([0, 250000])

ax2.hist(arTripDuration0718Cus, bins=25, range=(0, 50), color='m', alpha=0.8)
ax2.set_title('7/18 Trip Duration for Customers (min)')
ax2.set_xlabel('Time (min)')
ax2.set_ylabel('Frequency')
ax2.set_ylim([0, 250000])
plt.show()
```

11 84

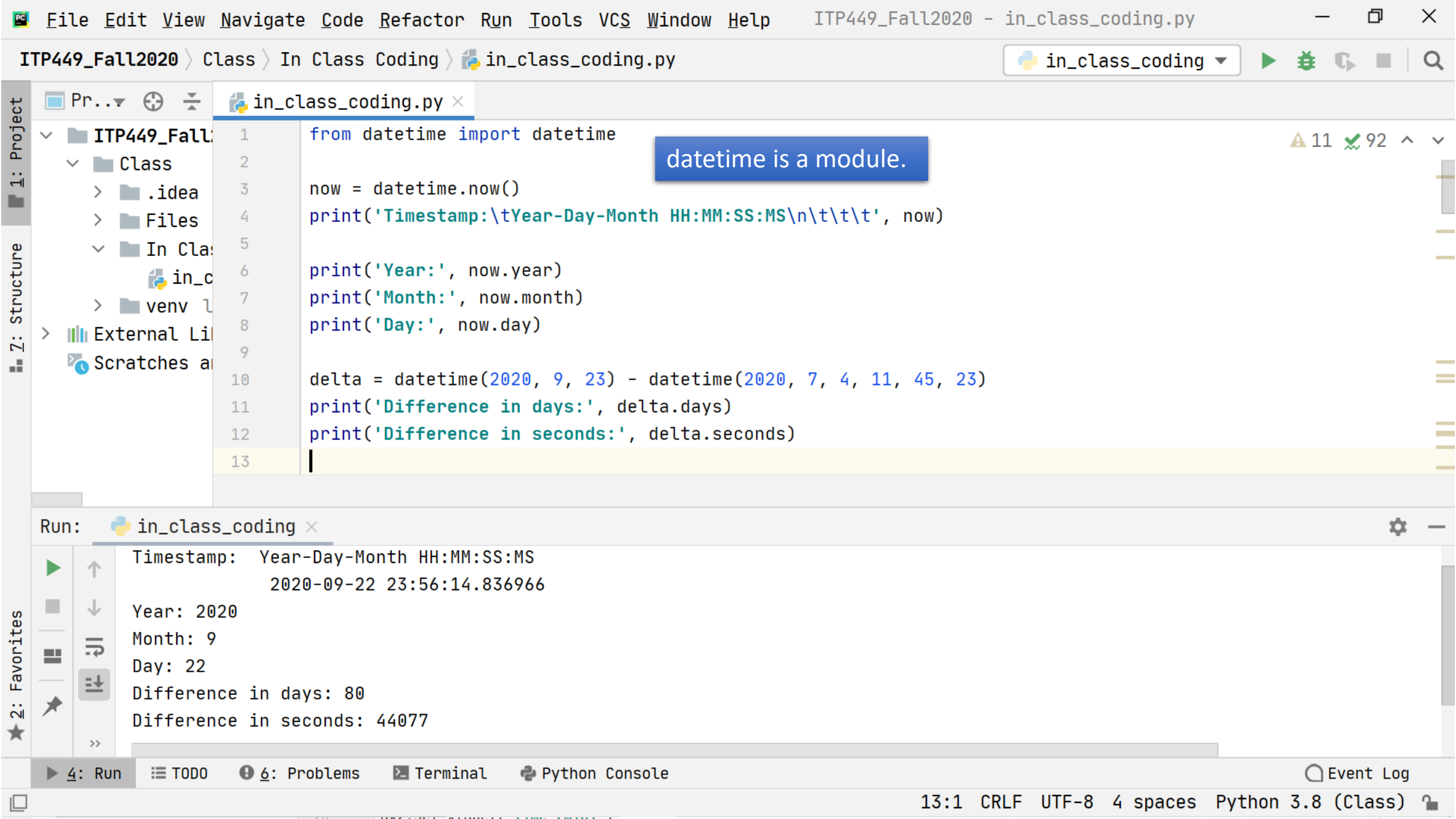
Time Series

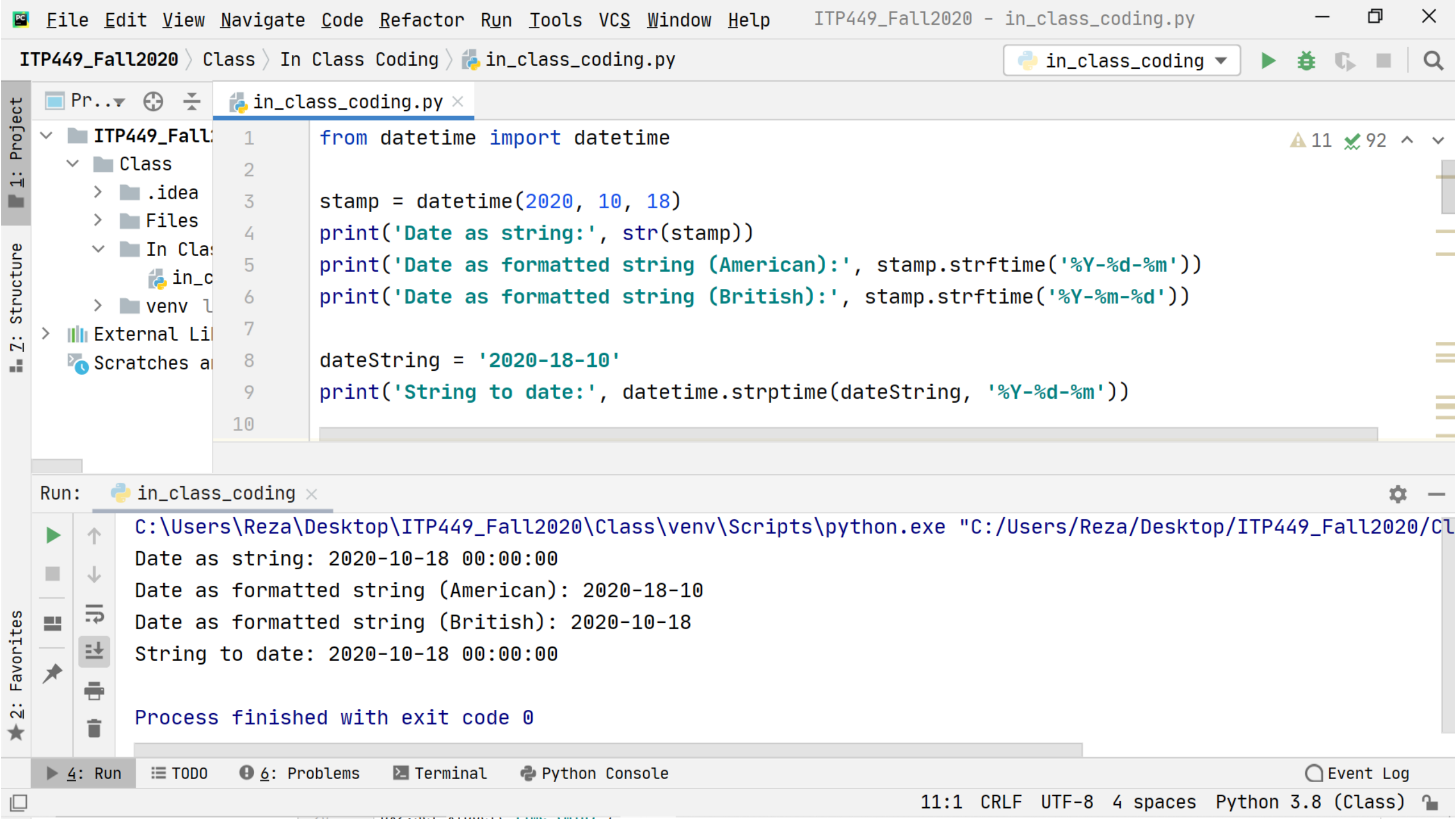
Fixed frequency – Data points occur at regular intervals according to some rule

Timestamp – specific instance in time

Fixed periods – logical period of time such as the month January 2007 or the full year 2010

Intervals of time – indicated by a start and end timestamp. (Periods can be thought of as special cases of intervals)


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_Fall
Class
.idea
Files
In Cla
in_c
venv
External Li
Scratches a
7: Structure
1 from datetime import datetime
2
3 now = datetime.now()
4 print('Timestamp:\tYear-Day-Month HH:MM:SS:MS\n\t\t\t', now)
5
6 print('Year:', now.year)
7 print('Month:', now.month)
8 print('Day:', now.day)
9
10 delta = datetime(2020, 9, 23) - datetime(2020, 7, 4, 11, 45, 23)
11 print('Difference in days:', delta.days)
12 print('Difference in seconds:', delta.seconds)
13
datetime is a module.
11 92
Run: in_class_coding
Timestamp: Year-Day-Month HH:MM:SS:MS
2020-09-22 23:56:14.836966
Year: 2020
Month: 9
Day: 22
Difference in days: 80
Difference in seconds: 44077
2: Favorites
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
13:1 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
1: Project
7: Structure
2: Favorites
ITP449_Fall
Class
.idea
Files
In Cla
in_c
venv
External Li
Scratches a
1 from datetime import datetime
2
3 stamp = datetime(2020, 10, 18)
4 print('Date as string:', str(stamp))
5 print('Date as formatted string (American):', stamp.strftime('%Y-%d-%m'))
6 print('Date as formatted string (British):', stamp.strftime('%Y-%m-%d'))
7
8 dateString = '2020-18-10'
9 print('String to date:', datetime.strptime(dateString, '%Y-%d-%m'))
10
11
92
Run: in_class_coding
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Cl
Date as string: 2020-10-18 00:00:00
Date as formatted string (American): 2020-18-10
Date as formatted string (British): 2020-10-18
String to date: 2020-10-18 00:00:00
Process finished with exit code 0
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
11:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Type	Description
`%Y`	Four-digit year
`%y`	Two-digit year
`%m`	Two-digit month [01, 12]
`%d`	Two-digit day [01, 31]
`%H`	Hour (24-hour clock) [00, 23]
`%I`	Hour (12-hour clock) [01, 12]
`%M`	Two-digit minute [00, 59]
`%S`	Second [00, 61] (seconds 60, 61 account for leap seconds)
`%w`	Weekday as integer [0 (Sunday), 6]
`%U`	Week number of the year [00, 53]; Sunday is considered the first day of the week, and days before the first Sunday of the year are “week 0”
`%W`	Week number of the year [00, 53]; Monday is considered the first day of the week, and days before the first Monday of the year are “week 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

1: Project

Pr...

in_class_coding.py

```
ITP449_Fall
  Class
    .idea
    Files
    In Cla
      in_c
    venv l
External Li
Scratches a
```

```
1  import pandas as pd
2  import numpy as np
3  from datetime import datetime
4
5  dates = [datetime(2020, 9, 1),
6           datetime(2020, 9, 8),
7           datetime(2020, 9, 11),
8           datetime(2020, 9, 12)]
9  ts = pd.Series(np.random.randn(4), index=dates)
10 print('Timestamp as Series Index:')
11 print(ts)
```

11 95

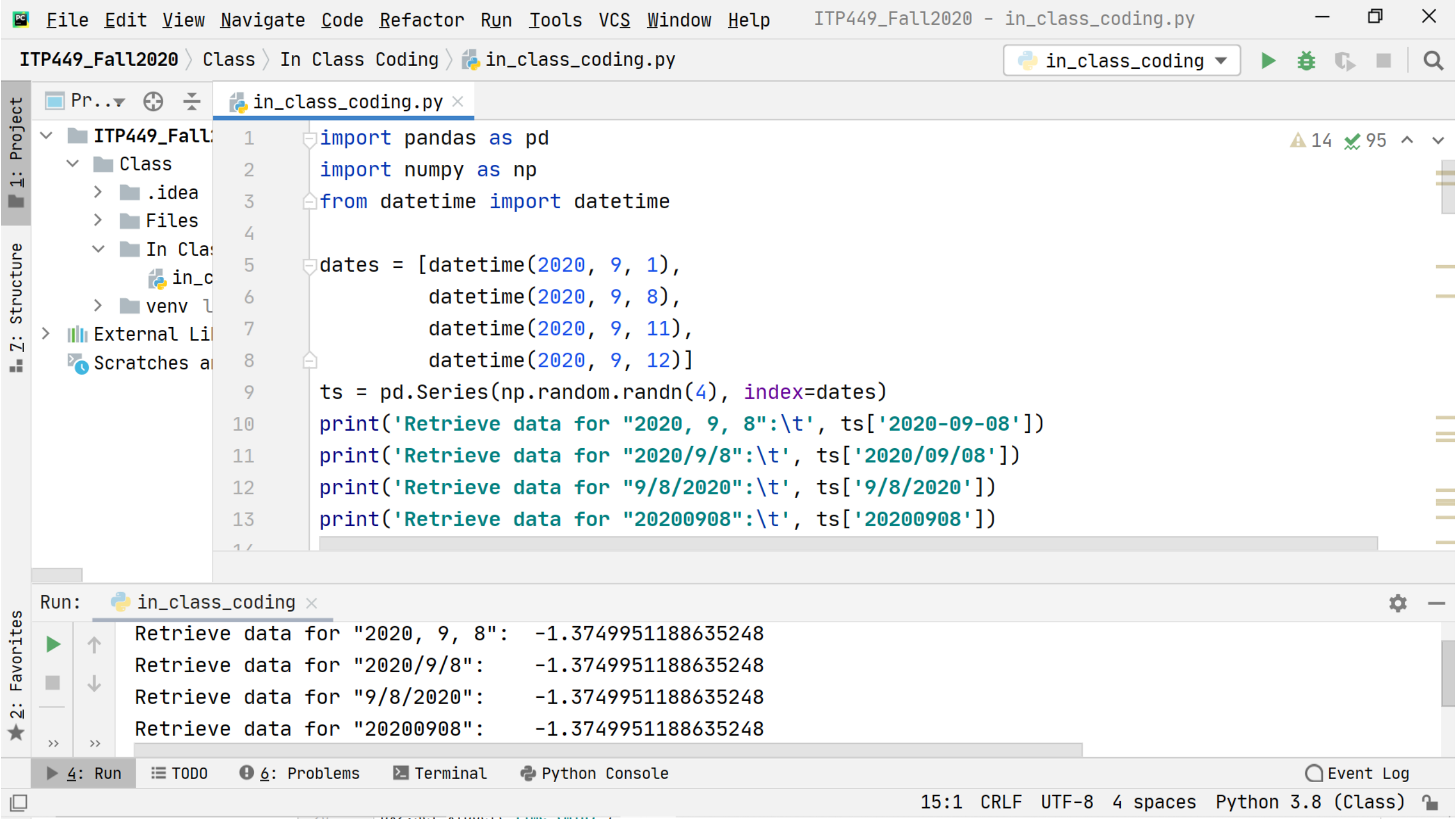
Run: in_class_coding

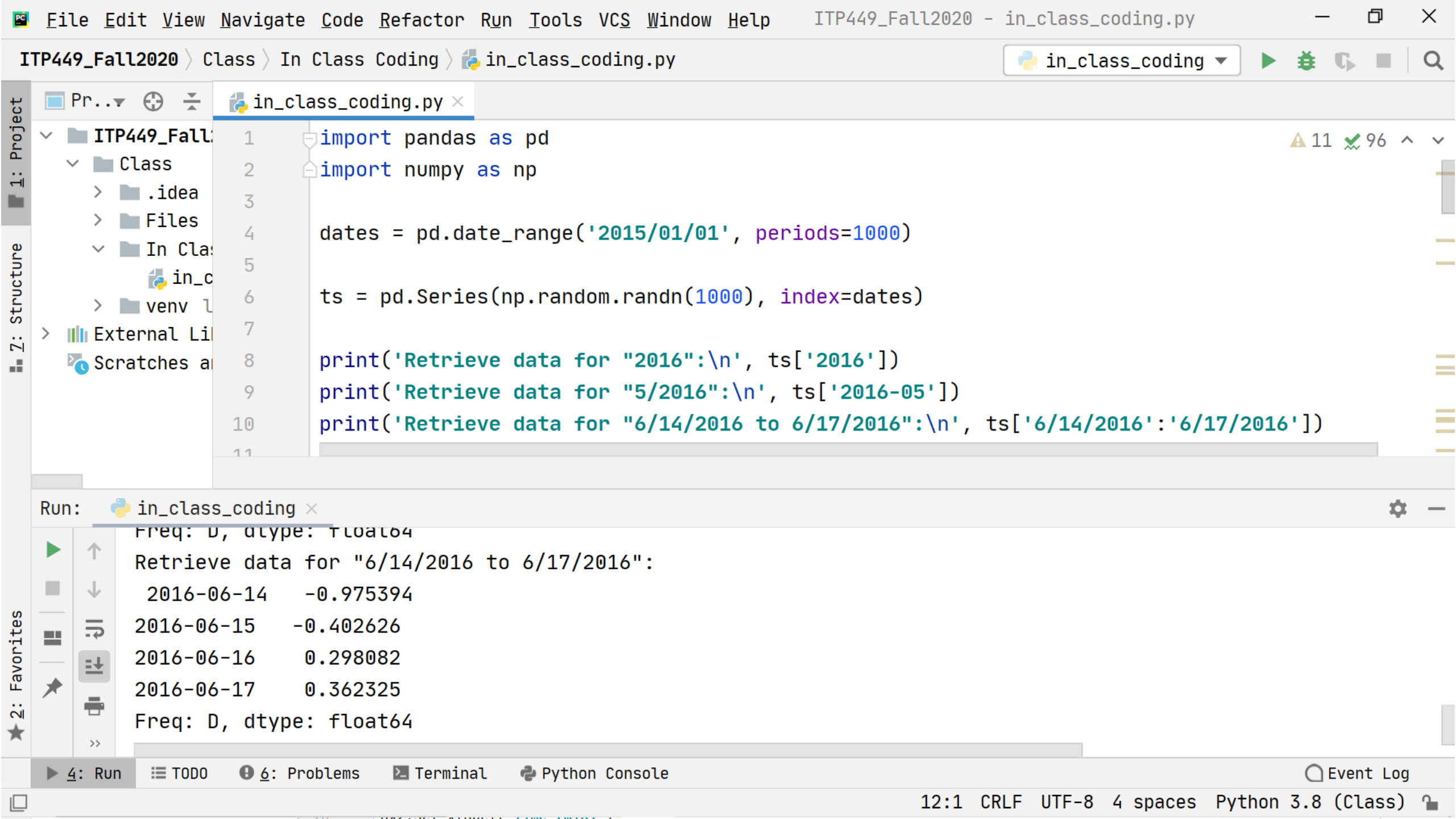
```
Timestamp as Series Index:
2020-09-01   -0.613781
2020-09-08    1.111112
2020-09-11   -0.773008
2020-09-12    0.525666
dtype: float64
```

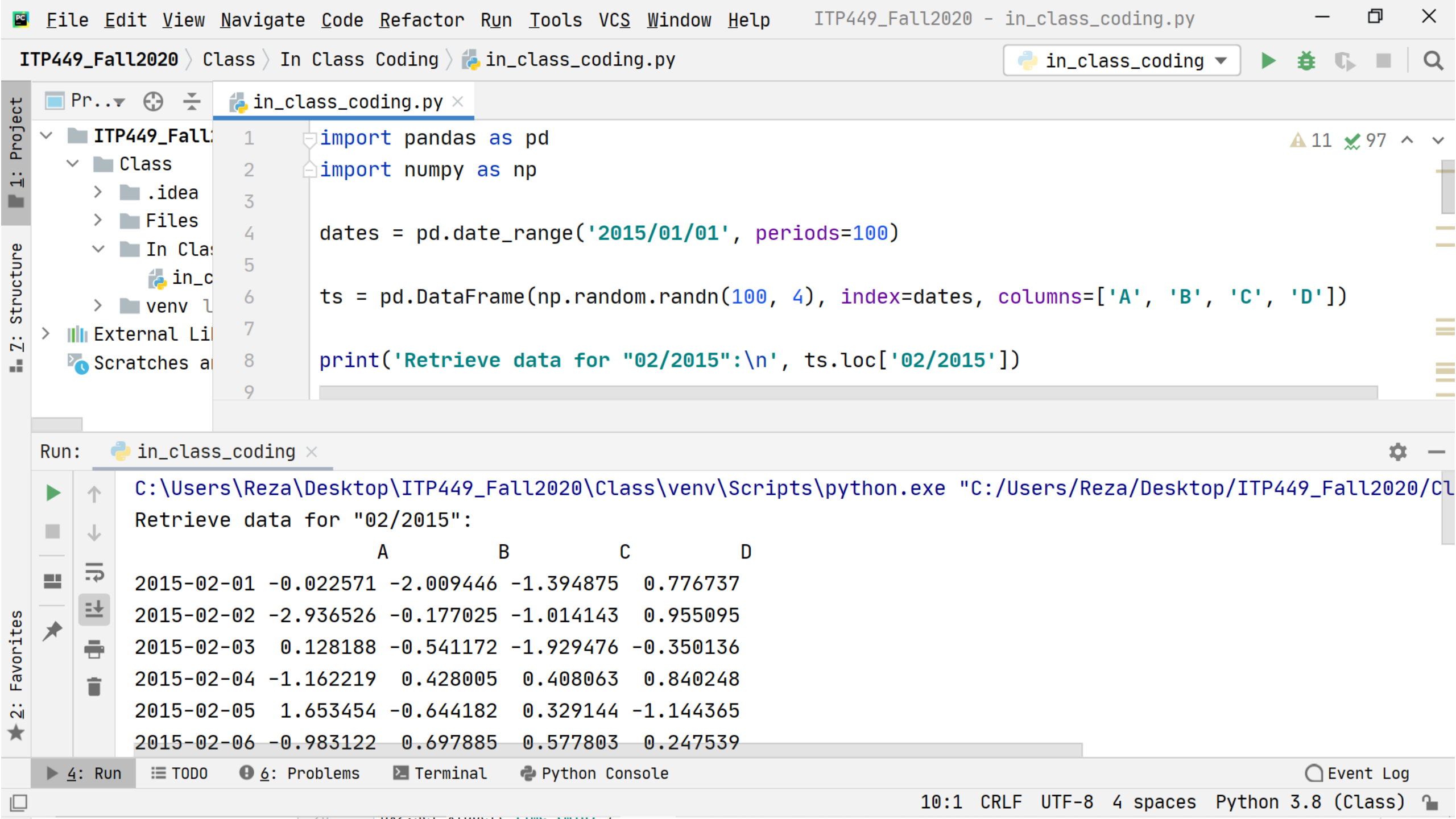
2: Favorites

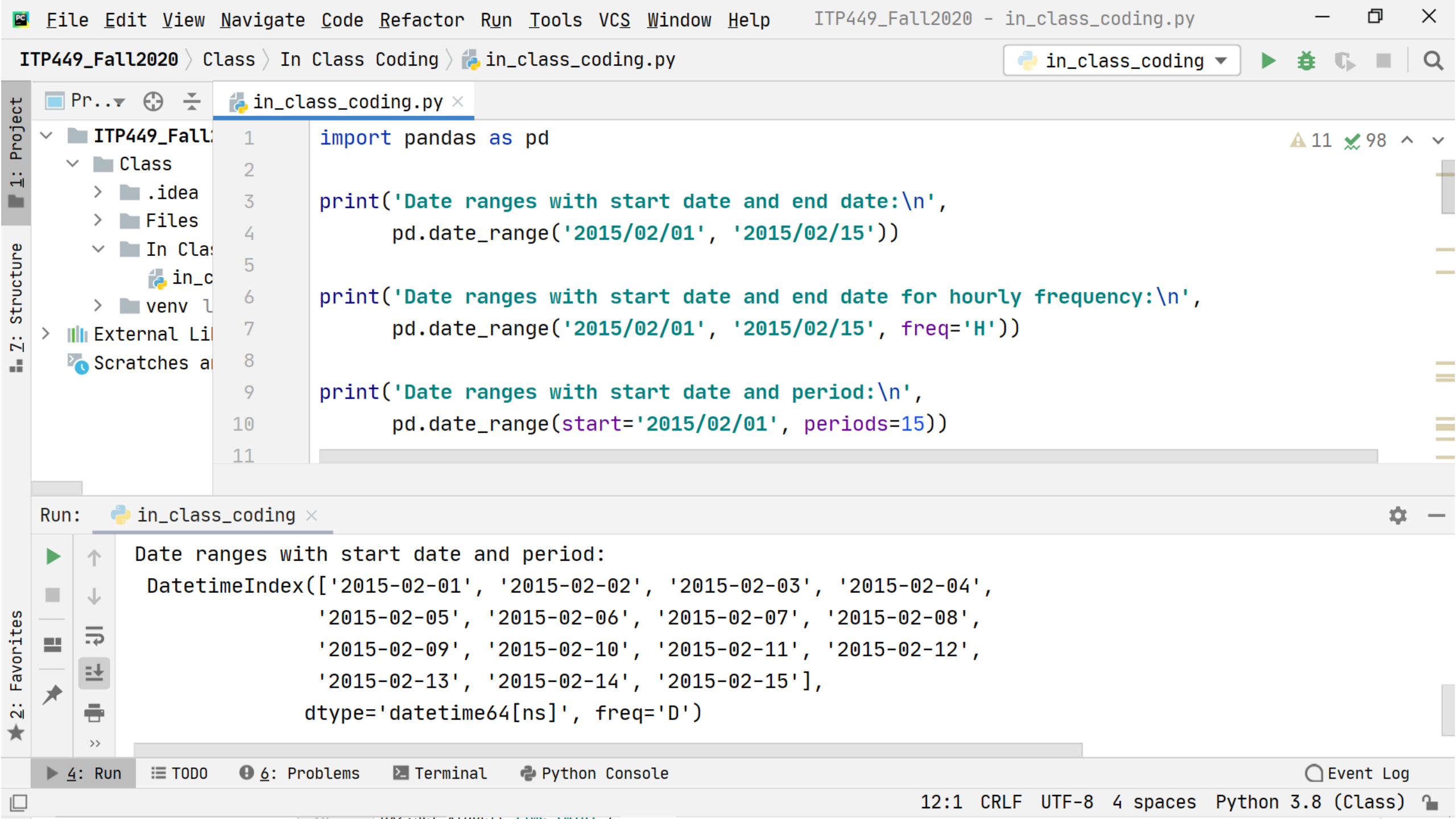
4: Run TODO 6: Problems Terminal Python Console Event Log

12:1 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
7: Structure
2: Favorites
in_class_coding.py
ITP449_Fall
Class
.idea
Files
In Cla
in_c
venv
External Li
Scratches a
14 95
1 import pandas as pd
2 import numpy as np
3 from datetime import datetime
4
5 dates = [datetime(2020, 9, 1),
6 datetime(2020, 9, 8),
7 datetime(2020, 9, 11),
8 datetime(2020, 9, 12)]
9 ts = pd.Series(np.random.randn(4), index=dates)
10 print('Retrieve data for "2020, 9, 8":\t', ts['2020-09-08'])
11 print('Retrieve data for "2020/9/8":\t', ts['2020/09/08'])
12 print('Retrieve data for "9/8/2020":\t', ts['9/8/2020'])
13 print('Retrieve data for "20200908":\t', ts['20200908'])
Run: in_class_coding
Retrieve data for "2020, 9, 8": -1.3749951188635248
Retrieve data for "2020/9/8": -1.3749951188635248
Retrieve data for "9/8/2020": -1.3749951188635248
Retrieve data for "20200908": -1.3749951188635248
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
15:1 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...
in_class_coding.py
ITP449_Fall
Class
.idea
Files
In Cla
in_c
venv
External Li
Scratches a
1 import pandas as pd
2 import numpy as np
3
4 dates = pd.date_range('2015/01/01', periods=1000)
5
6 ts = pd.Series(np.random.randn(1000), index=dates)
7
8 print('Retrieve data for "2016":\n', ts['2016'])
9 print('Retrieve data for "5/2016":\n', ts['2016-05'])
10 print('Retrieve data for "6/14/2016 to 6/17/2016":\n', ts['6/14/2016':'6/17/2016'])
11
11
96
Run:
in_class_coding
Freq: D, dtype: float64
Retrieve data for "6/14/2016 to 6/17/2016":
2016-06-14 -0.975394
2016-06-15 -0.402626
2016-06-16 0.298082
2016-06-17 0.362325
Freq: D, dtype: float64
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)


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: Project
7: Structure
2: Favorites
ITP449_Fall
Class
.idea
Files
In Cla
in_c
venv
External Li
Scratches a
1 import pandas as pd
2 import numpy as np
3
4 dates = pd.date_range('2015/01/01', periods=100)
5
6 ts = pd.DataFrame(np.random.randn(100, 4), index=dates, columns=['A', 'B', 'C', 'D'])
7
8 print('Retrieve data for "02/2015":\n', ts.loc['02/2015'])
9
11
97
Run: in_class_coding
C:\Users\Reza\Desktop\ITP449_Fall2020\Class\venv\Scripts\python.exe "C:/Users/Reza/Desktop/ITP449_Fall2020/Cl
Retrieve data for "02/2015":
A B C D
2015-02-01 -0.022571 -2.009446 -1.394875 0.776737
2015-02-02 -2.936526 -0.177025 -1.014143 0.955095
2015-02-03 0.128188 -0.541172 -1.929476 -0.350136
2015-02-04 -1.162219 0.428005 0.408063 0.840248
2015-02-05 1.653454 -0.644182 0.329144 -1.144365
2015-02-06 -0.983122 0.697885 0.577803 0.247539
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
in_class_coding.py
1: Project
7: Structure
2: Favorites
ITP449_Fall
Class
.idea
Files
In Cla
in_c
venv
External Li
Scratches a
1 import pandas as pd
2
3 print('Date ranges with start date and end date:\n',
4 pd.date_range('2015/02/01', '2015/02/15'))
5
6 print('Date ranges with start date and end date for hourly frequency:\n',
7 pd.date_range('2015/02/01', '2015/02/15', freq='H'))
8
9 print('Date ranges with start date and period:\n',
10 pd.date_range(start='2015/02/01', periods=15))
11
11
98
Run: in_class_coding
Date ranges with start date and period:
DatetimeIndex(['2015-02-01', '2015-02-02', '2015-02-03', '2015-02-04',
'2015-02-05', '2015-02-06', '2015-02-07', '2015-02-08',
'2015-02-09', '2015-02-10', '2015-02-11', '2015-02-12',
'2015-02-13', '2015-02-14', '2015-02-15'],
dtype='datetime64[ns]', freq='D')
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
12:1 CRLF UTF-8 4 spaces Python 3.8 (Class)

Alias	Offset type	Description
`D`	`Day`	Calendar daily
`B`	`BusinessDay`	Business daily
`H`	`Hour`	Hourly
`T` or `min`	`Minute`	Minutely
`S`	`Second`	Secondly
`L` or `ms`	`Milli`	Millisecond (1/1,000 of 1 second)
`U`	`Micro`	Microsecond (1/1,000,000 of 1 second)
`M`	`MonthEnd`	Last calendar day of month
`BM`	`BusinessMonthEnd`	Last business day (weekday) of month
`MS`	`MonthBegin`	First calendar day of month
`BMS`	`BusinessMonthBegin`	First weekday of month