

How to run Final_Project_Code_IS590PR.py file

These instructions are with respect to PyCharm as the IDE but we believe that they should be relevant to other IDE's as well.

1. Download the file from the GitHub repository -
https://raw.githubusercontent.com/rahulrohri/final_project_2020Sp/master/Final_Project_Code_IS590PR.py
2. Download the necessary datasets from the google drive link
https://drive.google.com/open?id=1g_StaWiaWQyNjNOu3wIFKG2dsIJZjyF
3. Open PyCharm and load this file in your environment.
4. Update the corresponding file paths in the program code (lines 47 – 50)
5. Click on the run command that will initiate the process of running the **Final_Project_Code_IS590PR.py** file.
6. Next you will be shown the following message in the run terminal assuming you have loaded the correct files and not made any changes to the base data.
The columns necessary for analysis are present in the EMS data file
The columns necessary for analysis are present in the Complaints data file
The columns necessary for analysis are present in the Arrests data file
7. Since our program is designed in a way to input only the necessary columns for analysis and save memory and RAM , you will be prompted 5 times to input the relevant columns. Click in the run terminal section when prompter and perform the following:
 - a. In the First prompt , enter the following column names
enter your column name 1 and press enter: **CMPLNT_NUM**
enter your column name 2 and press enter: **CMPLNT_FR_DT**
enter your column name 3 and press enter: **BORO_NM**
 - b. In the Second prompt , enter the following column names
enter your column name 1 and press enter: **ARREST_BORO**
enter your column name 2 and press enter: **ARREST_DATE**
enter your column name 3 and press enter: **ARREST_KEY**
 - c. In the Third prompt , enter the following column names
enter your column name 1 and press enter: **VIC_RACE**
enter your column name 2 and press enter: **OFNS_DESC**
enter your column name 3 and press enter: **CMPLNT_NUM**
After a few seconds you will see a message on the screen that says **"The correlation between Population Density and the crime per capita is : 0.7275769206466038"**
 - d. In the Fourth prompt , enter the following column names
enter your column name 1 and press enter: **OFNS_DESC**
enter your column name 2 and press enter: **BORO_NM**
enter your column name 3 and press enter: **CMPLNT_NUM**

- e. In the Fifth prompt , enter the following column names
 enter your column name 1 and press enter: **INCIDENT_RESPONSE_SECONDS_QY**
 enter your column name 2 and press enter: **INCIDENT_DATETIME**
 enter your column name 3 and press enter: **BOROUGH**

8. After this wait for a few minutes and the output will be shown on screen. The message will be
"The correlation coefficient between Population Density and the Incident response time is : 0.4639141825587108"

You should also get some plots on your screen assuming the packages are installed in your environment.

Additional Note – if you choose to run the doctests as well, then after all the above 8 steps are completed and run successfully, the doctests will run and at a point for the **get_file function** it will be waiting for the input from you. Since this is not a jupyter notebook , the input may not be visible . Just click on the run terminal Screen and type **"AircraftHex"** and press **"enter key"**. The type **"SessionID"** and press **"enter key"**. The reason is that we have fed a dummy csv file to check if the function is running and thus it is waiting for the user to input the 2 necessary columns for analysis. After that all doctests should pass and you should get a screen similar to the image below

```

Run: Doctests in Final_Project_Code_ISS90PR
Tests passed: 49 of 49 tests - 4 ms

Test Results
  ✓ Final_Project_Code_ISS90PR.EMS_details 0 ms
    ✓ ems_sample_csv = 'https://raw.githubusercontent.com 0 ms
    ✓ ems_sample_df = pd.read_csv(ems_sam 0 ms
    ✓ Pop_density_csv = 'https://raw.githubusercontent.com 0 ms
    ✓ Pop_density_df = pd.read_csv(Pop_dens 0 ms
    ✓ ans = EMS_details(ems_sample_df) 0 ms
    ✓ ans.iloc[0]['Population'] 0 ms
  ✓ Final_Project_Code_ISS90PR.corr_coeff 0 ms
    ✓ sample_csv = 'https://raw.githubusercontent.com 0 ms
    ✓ sample_df = pd.read_csv(sample_csv) 0 ms
    ✓ corr_coeff(sample_df['Age'],sample_df['t 0 ms
    ✓ data_dummy = {'Weight': [55,66,77,88,9 0 ms
    ✓ df_dummy = pd.DataFrame(data_dumr 0 ms
    ✓ corr_coeff(df_dummy['Weight'],df_dumr 0 ms
  ✓ Final_Project_Code_ISS90PR.dataset_validati 0 ms
    ✓ NYPD_Arrests = 'https://raw.githubusercontent.com 0 ms
    ✓ Complaints = 'https://raw.githubusercontent.com 0 ms
    ✓ EMS_incident = 'https://raw.githubusercontent.com 0 ms
    ✓ dataset_validation() 0 ms
  ✓ Final_Project_Code_ISS90PR.extract_year_mc 0 ms
    ✓ sample_csv = 'https://raw.githubusercontent.com 0 ms
    ✓ sample_df = pd.read_csv(sample_csv) 0 ms
    ✓ answer = extract_year_month(sample_df 0 ms
    ✓ answer.iloc[0]['Population'] #doctest: +N 0 ms

Testing started at 5:14 PM ...
C:\Users\rahul\Anaconda3\python.exe "C:\Program Files\JetBrains\PyCharm 2019.3.2\plugins\python\helpers\coverage_runner\run_coverage

The columns necessary for analysis are present in the EMS data file
The columns necessary for analysis are present in the Complaints data file
The columns necessary for analysis are present in the Arrests data file
enter your column name 1 and press enter:enter your column name 2 and press enter:enter your column name 3 and press enter:enter yo
enter your column name 1 and press enter:enter your column name 2 and press enter:enter your column name 3 and press enter:enter yo
25 items had no tests:
__main__
__main__.DocTestRunner
__main__.DocTestRunner.DocTestRunner__run
__main__.DocTestRunner.__init__
__main__.DocTestRunner.addTest
__main__.DocTestRunner.addTests
__main__.DocTestRunner.countTests
__main__.DocTestRunner.start
__main__.TeamcityDocTestResult
__main__.TeamcityDocTestResult.__TeamcityDocTestResult__getDuration
__main__.TeamcityDocTestResult.addError
__main__.TeamcityDocTestResult.addFailure
__main__.TeamcityDocTestResult.getSuiteLocation
__main__.TeamcityDocTestResult.getSuiteName
__main__.TeamcityDocTestResult.getTestId
__main__.TeamcityDocTestResult.getTestName
  
```

If running with coverage feature of pycharm, then you should get something similar to the below image

The screenshot displays the PyCharm IDE interface with the following components:

- Top Panel:** Shows the project structure and a search bar. The file `Final_Project_Code_IS590PR.py` is open, displaying code with line numbers 599 to 609. The code includes comments and function calls like `pd.read_csv` and `pd.read_csv`.
- Right Panel:** Displays the coverage report for `Doctests in Final_Project_Code_IS590PR`. It shows that 50% of files and 96% of lines are covered. A table lists the elements and their coverage statistics:

Element	Statistics, %
idea	
Final_Project_Code_IS590PR.py	96% lines c...
Final_Project_Code_IS590PR_Functions.py	not covered

- Bottom Panel:** Shows the test results for `Doctests in Final_Project_Code_IS590PR`. It indicates that 49 tests passed out of 49 tests, taking 4 ms. The test results are listed in a tree view:

- Final_Project_Code_IS590PR.EMS_details (0 ms)
- ems_sample_csv = 'https://raw.githubusercontent.com/rahulrohri/final_project_2020Sp/master/DocTest%20Dummy%20Files/NYC_getple_NYC_csv' (0 ms)
- ems_sample_df = pd.read_csv(ems_sam (0 ms)
- Pop_density_csv = 'https://raw.githubusercontent.com/rahulrohri/final_project_2020Sp/master/DocTest%20Dummy%20Files/Dummynsity_csv' (0 ms)
- ans = EMS_details(ems_sample_df) (0 ms)
- ans.iloc[0]['Population'] (0 ms)
- Final_Project_Code_IS590PR.corr_coef (0 ms)
- sample_csv = 'https://raw.githubusercontent.com/rahulrohri/final_project_2020Sp/master/DocTest%20Dummy%20Files/Dummynsity_csv' (0 ms)

The test results also show the output of the tests, including the path to the coverage runner and the message "Tests passed: 49".

End of program