Atira Scrap zip file has two folders in it.

1. Webcrawler
2. DataAnalyser

Aside of both of them there is **requirements.txt** file and output snapshot, which is the snapshot of output file. Output file is also included with the name **data.json**. There is a snapshot of barchart in DataAnalyser folder. This documents will explain how to set up the environment and how to run the code. I will assume that python is already installed on the system on which this code is to be run, otherwise please go online and install **python3**. I have used **python3** in my projects.

# **Setting up environment:**

Before running code it is crucial that you setup your environment.

1. First create any folder and inside it create a virtual environment. I have used **virtualenv** because some packages were giving error when I was installing them in an virtual environment of **venv**. If you do not have virtualenv then install it using **pip** using the command given below

**Command: pip install virtualenv**

1. Activate your environment. After that put both folders and requirements.txt in the folder you created your virtual environment.
2. Use the command given bellow to installed all the packages listed in requirements.txt. **Please make sure you have activated your environment**.

**Command: pip install -r requirements.txt**

If you have followed these steps correctly then now your environment is setup and code can be run.

# **WebCrawler Folder:**

WebCrawler folder has a **scrapy project** by the name of **AtiraSpider** with configuration file. For crawling I have used **scrapy**, which is the framework designed for scraping websites. In **Webcrawler > AtiraSpider > spiders** there is a file by the name of **AtiraCrawler.py**. This file has all the code to scrap from website.

The code can be run by using the command given below

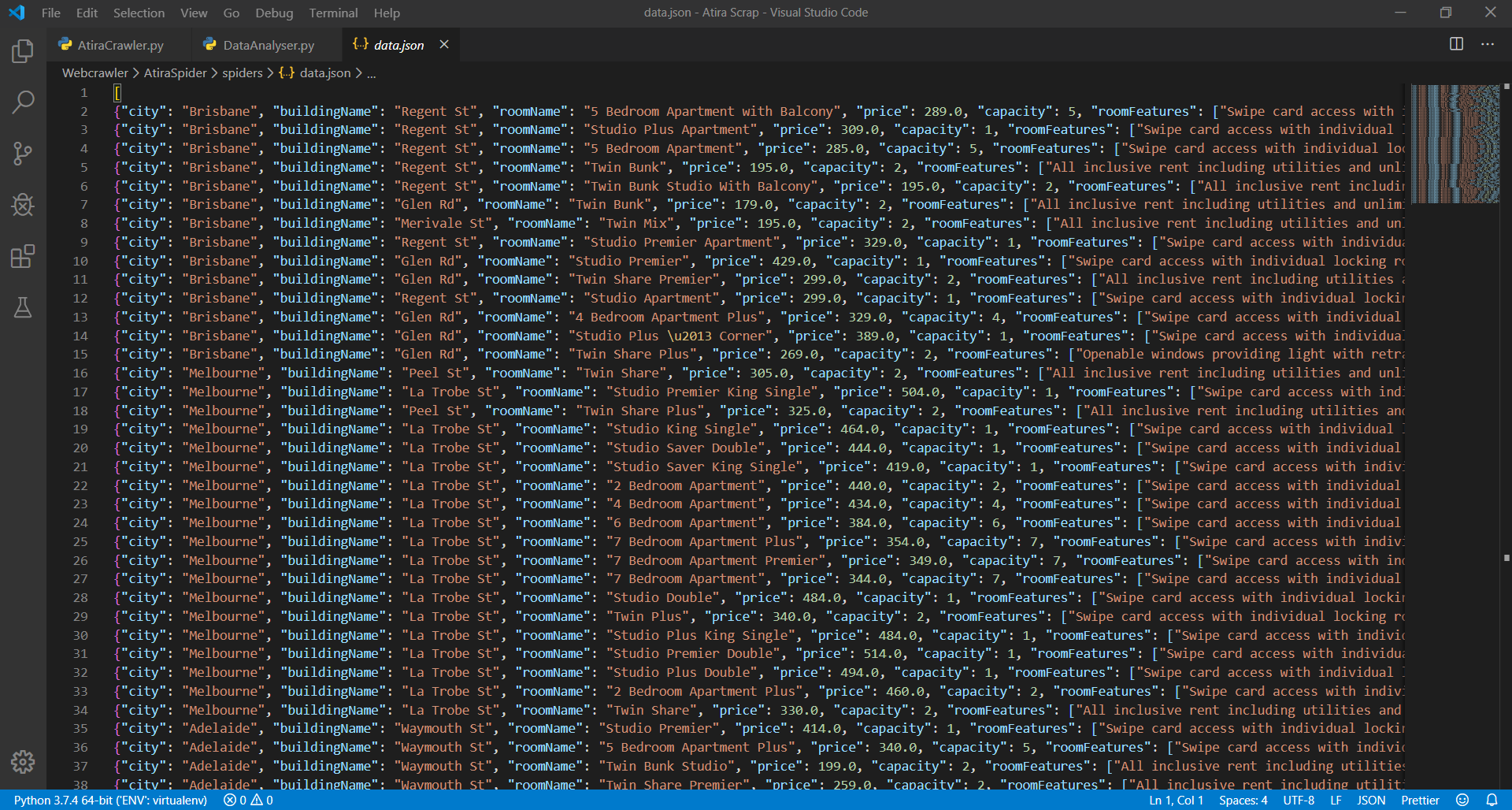
**Command: scrapy crawl AtiraCrawler -o data.json**

Here **data.json** is the output file in which all the scraped data will be saved in json format.

**Important: You can give any name to output file but it is important that after you create it, you do not change its position as it will be created in the same directory as AtiraCrawler.py. DataAnalyser which I will talk later will look for the file in the same directory as AtiraCrawler.py so It is really important that you follow these steps.**

**Important: After running the crawler once and generating the output file, if you want to run the crawler again then it is important that the name of output file is different than the name you used first time. Because if the name is same then scrapy will append the data of second time in the same file. So either use a different name each time you run the crawler or If you want to use the same name of output file then delete it or move it before running again.**

After this it will take about 10 – 15 min for crawler to crawl the website. The scraped data is now saved in the output file alongside **AtiraCrawler.py**. Here is a snap shot of my output file.

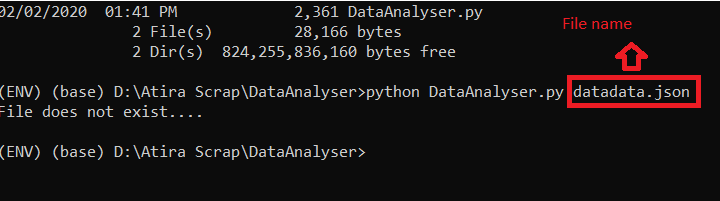


**As mentioned before do not move this file** as DataAnalyser will look for it in the same directory as AtiraCrawler.py.

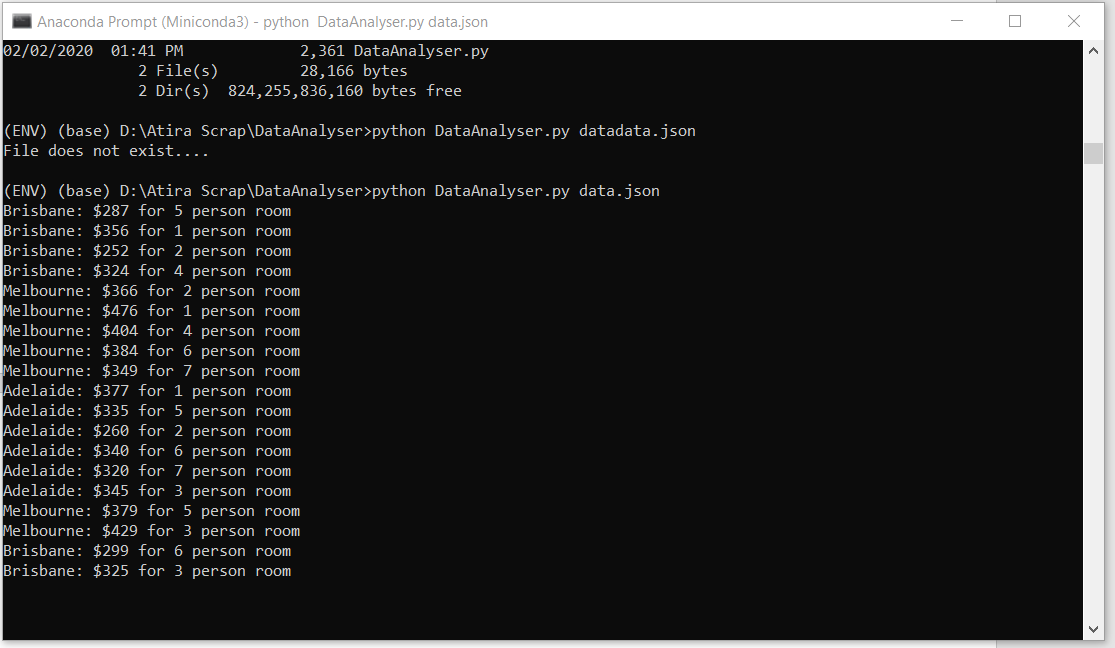
# **DataAnalyser Folder:**

DataAnalyser has a file with name **DataAnalyser.py**. This file has the code to open the output file and analyze the data. To run the code use the command given below

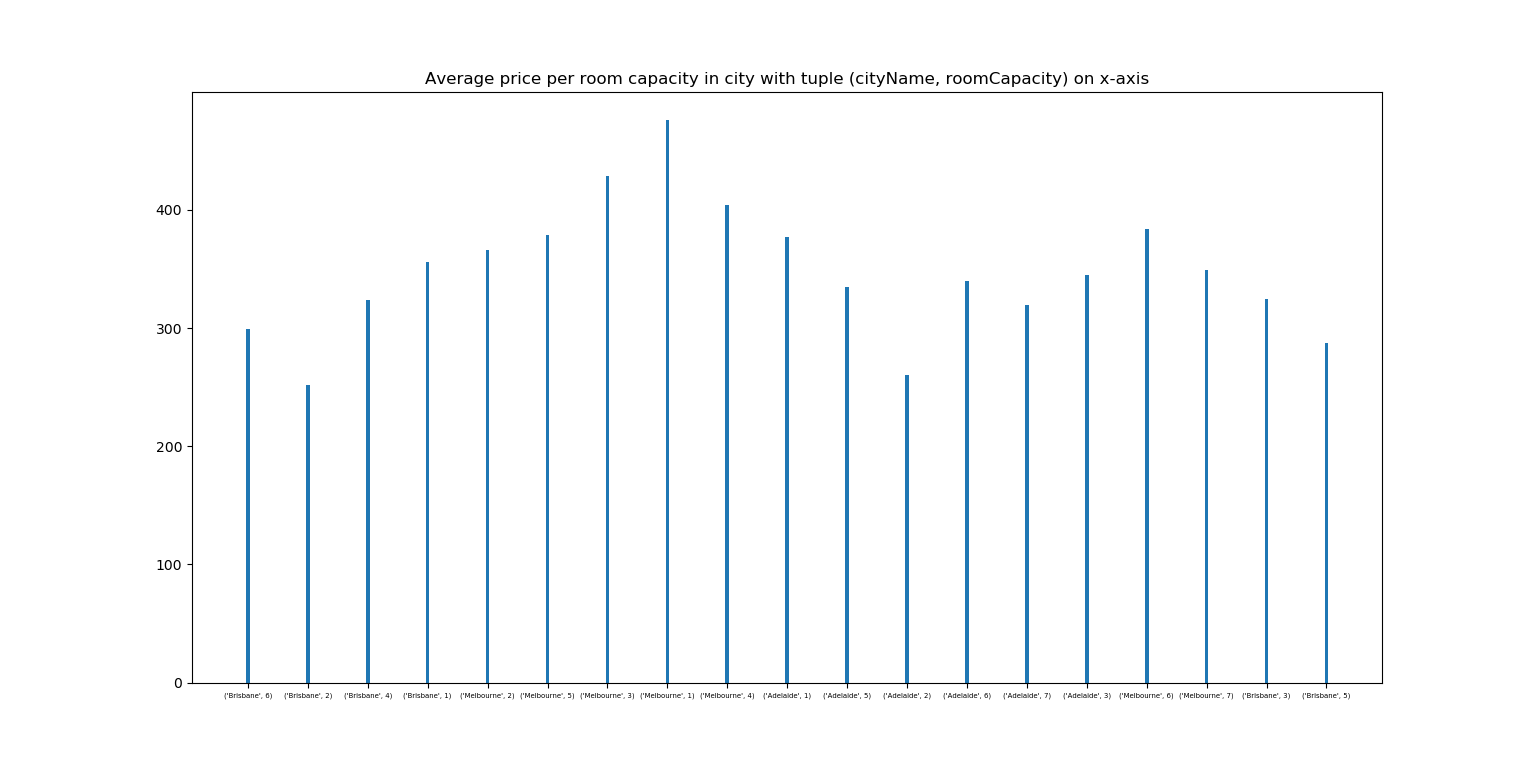
**Command: python DataAnalyser.py data.json**

Here **data.json** is the name of my output file. You will write the name of your file. If file is moved or does not exist then DataAnalyser will tell you that file does not exist as shown below

Once you run the command then on the command line you can see following results



I have also used **matplotlib.pyplot** library to create a bar chart of this data. **Once you run the command then the bar chart screen will open up, Please click on full screen button and maximize it otherwise on the x-axis values will overlap**. Here is the snapshot if you maximize your screen.



After it is in full screen you will be able to see x-axis clearly.