**ICP-3 Description**

In this ICP, we have learnt about the following concepts and making use of them in programming the below tasks.

* Classes
* Object Oriented Concepts
* Instances
* Init
* self
* Private, Protected, Public
* Inheritance
* Scientific Python
* Numpy Package
* Web scraping

**Software Requirements:**

* Python version 3
* Pycharm
* Anaconda
* Github

**Objectives of this ICP:**

**(1) Create a class Employee and then do the following**

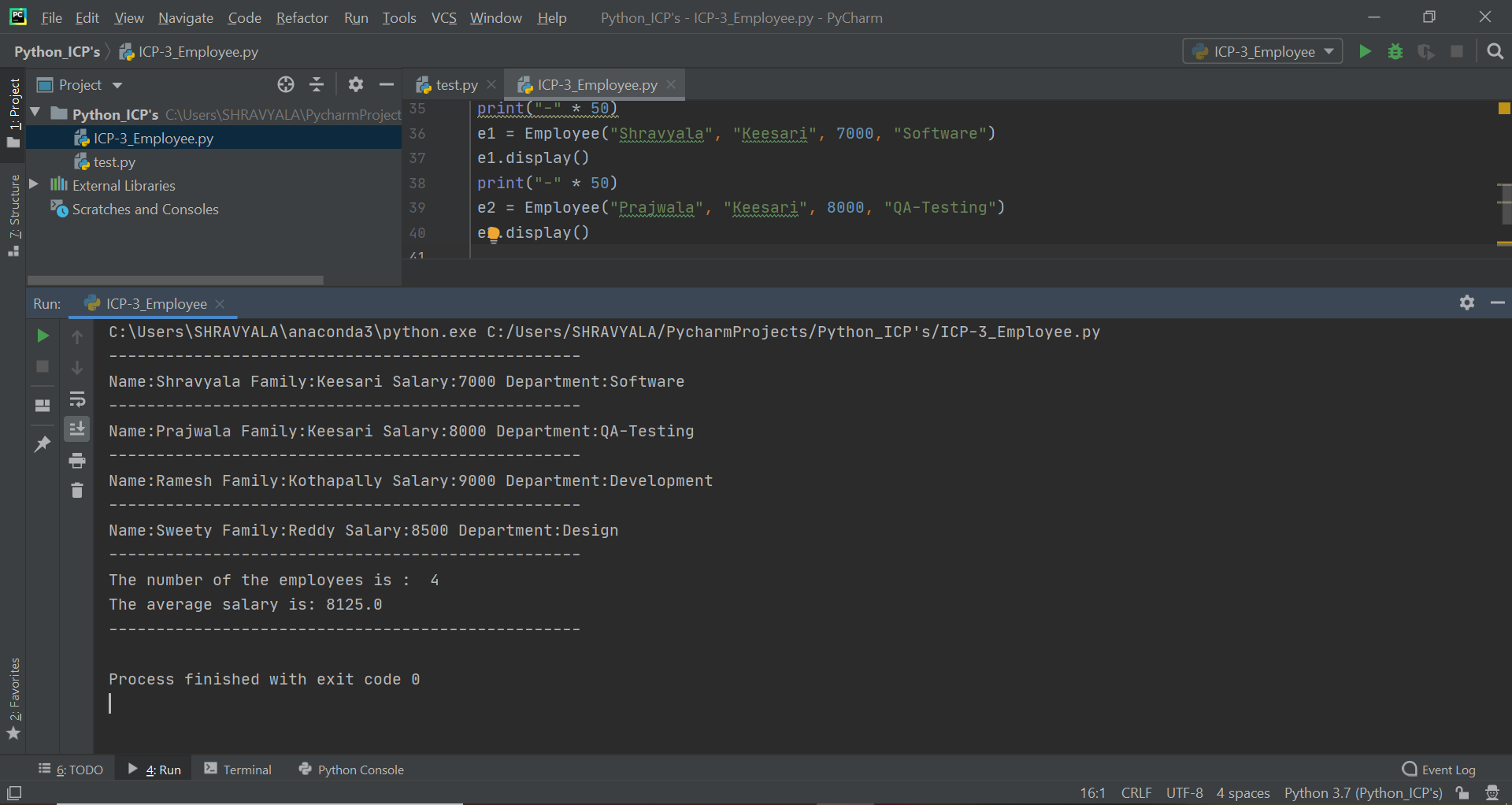
* **Create a data member to count the number of Employees**
* **Create a constructor to initialize name, family, salary, department**
* **Create a function to average salary**
* **Create a Fulltime Employee class and it should inherit the properties of Employee class**
* **Create the instances of Fulltime Employee class and Employee class and call their member functions.**

For the above case, python code is written in such a way that:

1. A Class is defined with the name “Employee”
2. Data members are created to count the no of employees and total salary of them

numofEmp = 0  
total\_salary = 0

1. A constructor is created to initialise the information of the employees such as name, family, salary, department.
2. Function **“avg\_sal”** is created to calculate the average salary of the employees.
3. Function is defined to display the information of the employees with “**def display(self):”**
4. Now a class **“FulltimeEmployee”** is created and inherited the data of the main class Employee.
5. 2 Instances are created in each of the class
6. Employee information and calculated average salary are displayed with print function
7. Hence the resultant output is displayed below.



**(2) Web scraping**

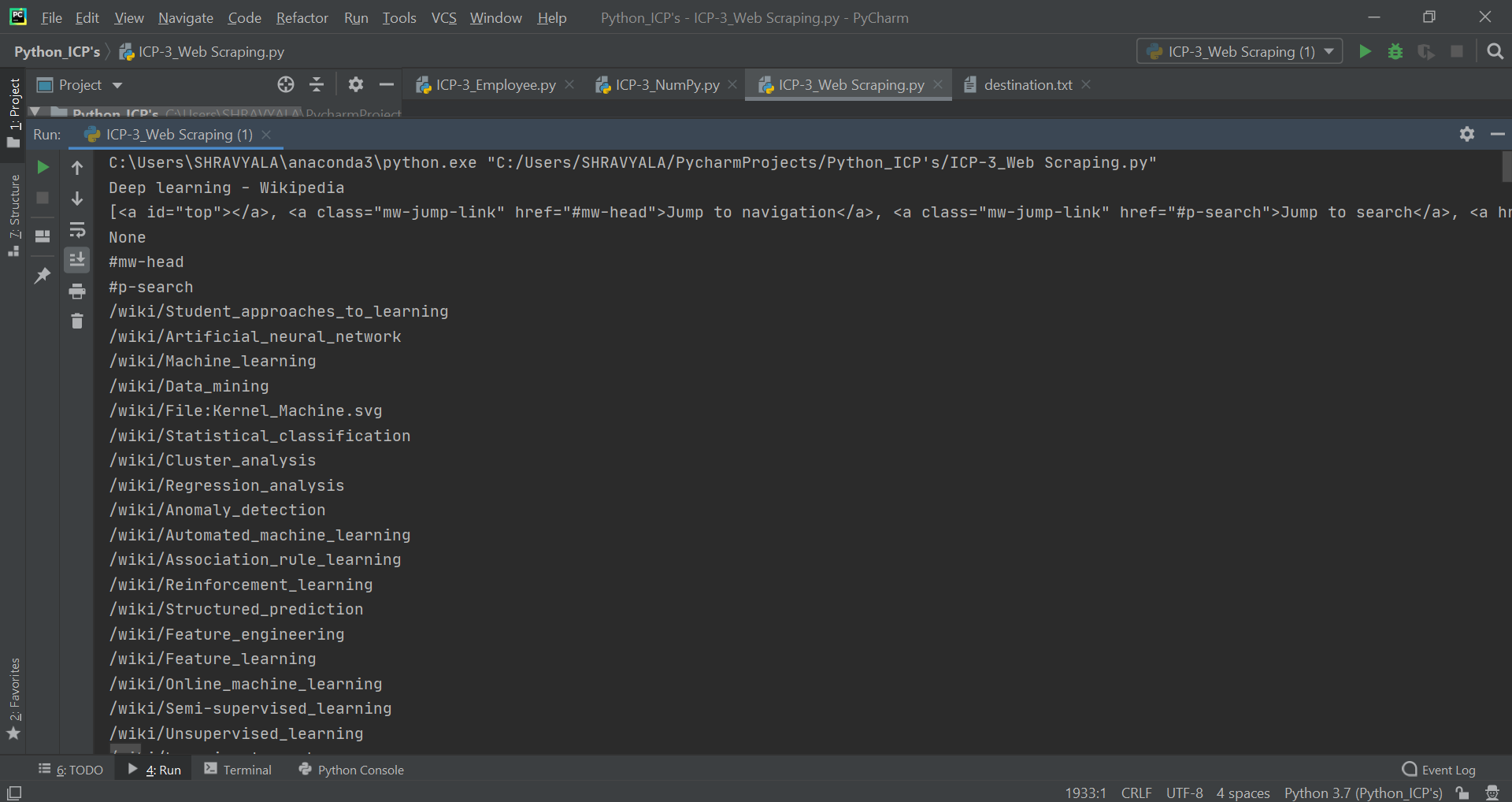
**Write a simple program that parse a Wiki page**

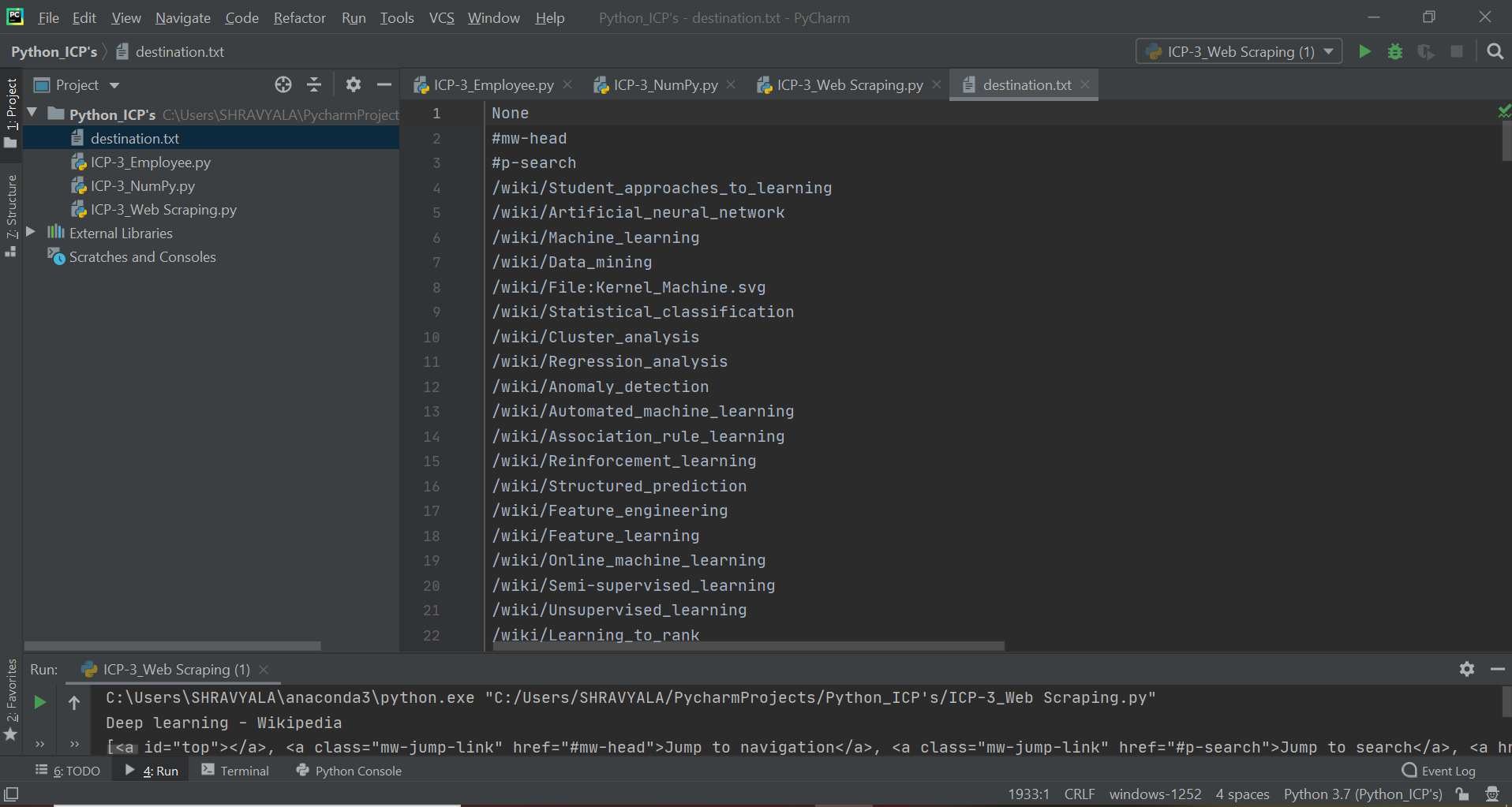
**mentioned below and follow the instructions:**

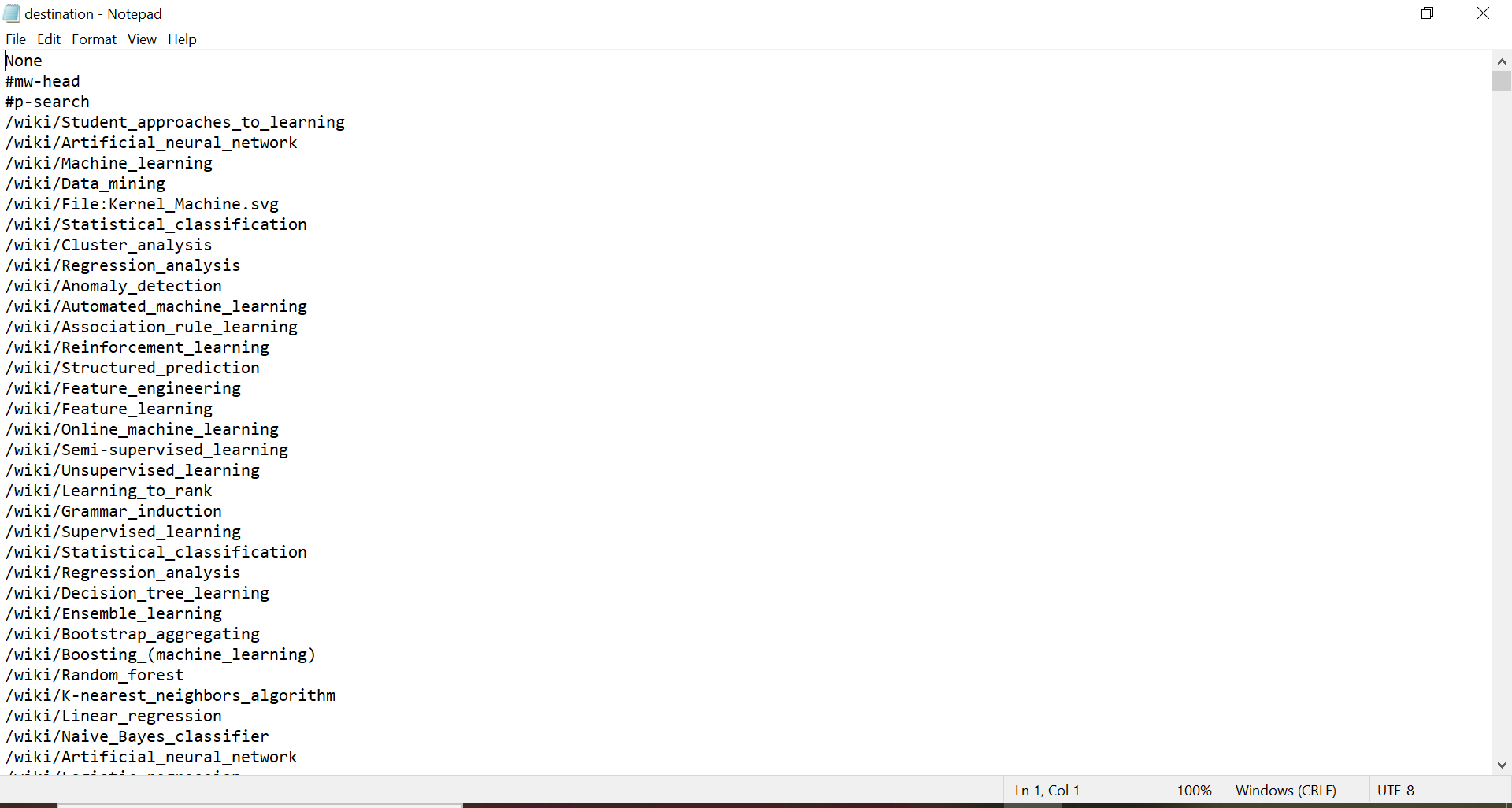
**https://en.wikipedia.org/wiki/Deep\_learning**

* **Print out the title of the page**
* **Find all the links in the page (‘a’ tag)**
* **Iterate over each tag(above) then return the link using attribute "href" using get**
* **Save all the links in the file**

1. Web Scraping is a computer software technique for extracting information from websites
2. It take’s URL’s to scrap the data
3. It transforms the unstructured data (HTML format) on the web into structured data.
4. “BeautifulSoup” library is used for extracting data from html and xml files
5. In this program, the **“BeatifulSoup”** library is imported first to extract the data from the given website [**https://en.wikipedia.org/wiki/Deep\_learning**](https://en.wikipedia.org/wiki/Deep_learning)
6. **“html.parser”** is used to parse the content from the website.
7. The title of the web page is printed
8. All the links in the page are found using the ‘a’ tag which gives all the hyperlinks
9. ‘href’ attribute indicates the link destination and is printed at the output.
10. All the links are saved in the file “destination.txt”
11. Hence the resultant output is shown below.







**(3) Numpy**

**Using NumPy create random vector of size 15 having only Integers in the range 1-20.**

**Then reshape the array to 3 by 5**

**Then replace the max in each row by 0**

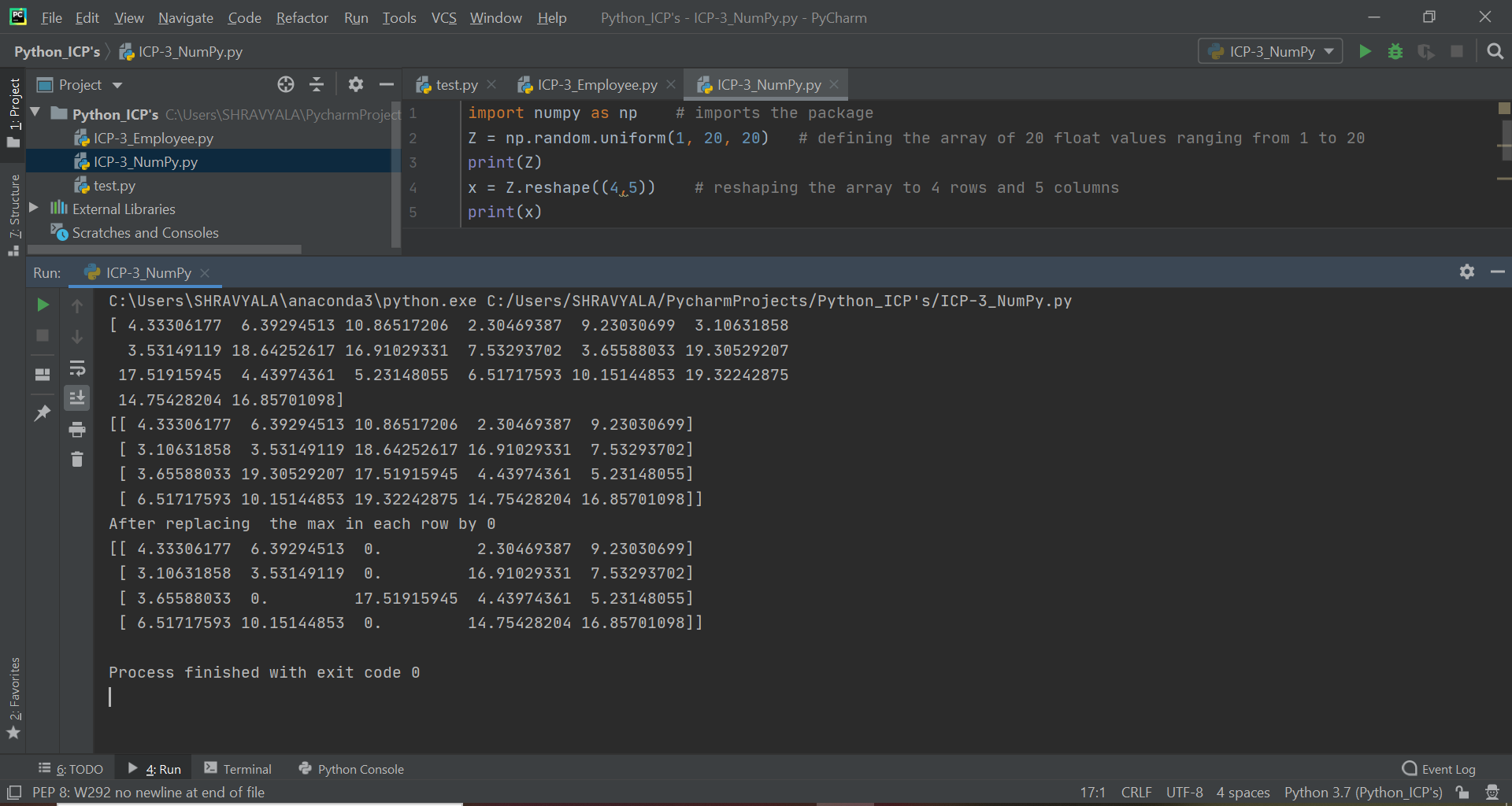
**(you can NOT implement it via for loop. You need to use np.where, reshape)**

**An input text file “word” is created with the lines**

1. First ensure that “numpy” package is installed of latest version.
2. “NumPy” is very useful for N-dimensional array objects.
3. Numpy is imported first and later the array is defined with 20 float values which ranges from 1 to 20.
4. Array is printed
5. Next the array size is reshaped to 4:5 which means it rearranges the values from the above list to 4 rows and 5 columns.
6. The highest or max value in each row is replaced with “0” using the below command

y = np.where(x == np.amax(x, axis=1, keepdims=True), 0, x)

1. Hence the resultant output is shown below.



**Video Link:**