**Python Programming**

**Lab Assignment-1**

**Team ID:** 5

**Partner 1:** Archana Reddy Basani, Class ID: 5

**Partner 2:** Sree Prathyusha Nomula, Class ID: 21

**Introduction:**

In this lab assignment, we have worked on:

1. Searching a string and finding the first non-repeated characters in the string
2. Removing the content in file1 which is inside file2
3. The list of students attending “python” but not “web applications”
4. Programming for the hospital admission system
5. Programming a code which downloads the html webpage containing a table using Request library and then parsing the page using the beautifulsoup library.

**Objectives:**

Applying dictionaries, arrays, and iterations in strings for finding the first non-repeated characters in the string.

Applying splitting, reading, looping for the files in order to remove the content in file 1 which is inside file 2.

Applying subtraction rule to get list of students in one class.

Applying special built-in methods to create the Hospital Admission system

**Approaches:**

We used classes, constructors, inheritance and multiple inheritance and also created instances for all the classes and created a relationship between them in creating the hospital admission system.

We have downloaded the html webpage which contains a table and parsed it using beautifulsoup library.

**Workflow:**

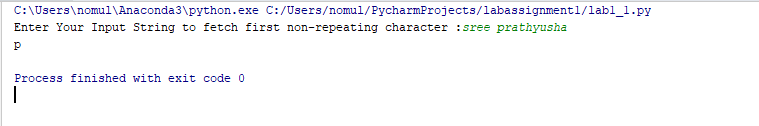
Question 1:

We used dictionary to store the counts of each character in string input and then an array to store the orders of characters. We will iterate the array to check the first character whose count is 1 and returning the first character if it’s count is 1. If there are no repeated characters, it will automatically return no repeated characters in the string.

Code snippet:

A screenshot of a social media post

Description generated with very high confidenceOutput:



Question 2:

Taking the two text files, opening the second file and splitting the file and reading it using file2WordsList = f.read().lower().split()

Also the words list in file 1 is given as

file1WordsList = f.read().split() and open it through the write access.

And then we use looping file 1 list and check whether file 1 is present in file 2

Code snippet:

A screenshot of a social media post

Description generated with very high confidence

Output:

A screenshot of a social media post

Description generated with very high confidence

Question 3:

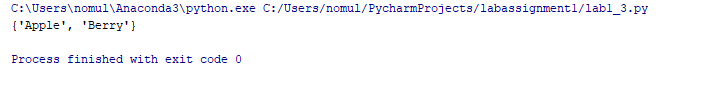
In this, taking the two lists of students who are attending ‘python, and ‘web applications’ and prints the list of students who are attending only ‘python’ using subtraction rule.

Code snippet:

A screenshot of a cell phone

Description generated with very high confidence

Output:



Question 4:

In this, we have taken 6 classes hospital, doctor, patient, clerk, nurse, book and also used \_\_init\_\_ constructor in all the classes. We took subclasses which inherits the main class hospital properties and calls the super class constructor and also used self in order to map the parameters. Used multiple inheritance for the class book by taking the classes patient and nurse. Finally, we defined the list and append the objects by creating instances of all the classes.

Code snippet:

A screenshot of a social media post

Description generated with very high confidence

A screenshot of a social media post

Description generated with very high confidence

Multiple inheritance

A screenshot of a social media post

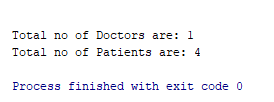
Description generated with very high confidence

A screenshot of a social media post

Description generated with very high confidence

Output:

A screenshot of a cell phone

Description generated with very high confidence

Question 5:

In this, we are importing the url using request library and assign a link to the variable and open the link and converts it in to html using

soup = BeautifulSoup(getLink, "html.parser") and prints the header for the wiki page and goes to the table and collects data from the class and loop all elements in tr tags.

Code snippet:

A screenshot of a social media post

Description generated with very high confidence

Output:

A screenshot of a cell phone

Description generated with very high confidence