|  |  |
| --- | --- |
| **rogramming 1 (PRG1)**  Diploma in IT / DS / CSF / IM / CICTP  Year 1 (2023/24) Semester 1 | Week **3** |
|  |
| **Activities : String Manipulation, List and Functions** | |

**OBJECTIVES**

At the end of this exercise, students should be able to:

* Access characters in strings
* Manipulate strings
* Create Lists
* Process Lists using List operators and functions
* Create user-defined function and call it

**IMPORTANT**

* Create a folder, **Week03**, on your hard disk.
* Save this word document as **Week03-YourName.docx** in the **Week03** folder created above.
* For each question, type your answer into the box provided in the question.
* For the questions that require you to write Python program, create the Python program with the given file name in the **Week03** folder created above. Do add the description, your name and student ID as comments at the beginning of each program.
* At the end of the session, save this word document, compress all the files in your **Week03** folder (i.e. the Python program files and this word document) and submit the zip file in POLITEMall.

**PART 1 - Lecture 3-1 (String Manipulation)**

Activity 1

Given **message = 'Baby Shark do do do'** , determine the output for the given Python code :

|  |  |
| --- | --- |
| **Python code** | **Output** |
| >>> len(message) | 19 |
| >>> message.find('do') | 11 |
| >>> message.find('do’, 0, 12) | -1 |
| >>> message.replace('o ', 'ie') | Baby shark diediedo |
| >>> message[5:10].isalpha() | True |
| >>> 'shark' in message | False #because its Shark not shark |
| >>> message[22] | IndexError |

Activity 2

Assume that string **s1 = 'Programming 1'** & **s2 = 'PRG1'**, write the python code to achieve the objectives. The expected output for each objective is provided for your reference and comparison.

|  |  |  |
| --- | --- | --- |
| **Objective** | **Python code** | **Expected Output** |
| Determine the length of string s1 | print(len(s1)) | 13 |
| Check if s1 has even number of characters | print(len(s1)) if: s1 % 2 = 0 print(True)  else:  print(False) | False |
| Replace every character 'g' in string **s1** with character 'z'. | s1.replace(‘g’, ‘z’) | 'Prozramminz 1' |
| Append the string **s2** to the string **s1,** with a space in between | s1 = 'Programming 1'  s2 = 'PRG1'  print(s1 + ' ' + s2) | 'Programming 1 PRG1' |
| Check if string **s1** starts and ends with the same character | if s1[1] == s1[-1]:  print("True")  else:  print("False") | False |

Activity 3

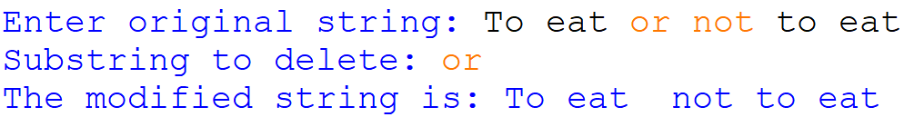
You are given a string s that contains three integers separated by semi-colons. For example, **s** may be **'10;20;30'**. Write Python code to achieve the objectives. The expected output for each objective is provided for your reference and comparison.

|  |  |  |
| --- | --- | --- |
| **Objective** | **Python code** | **Expected Output** |
| Split the string s into substrings using separator ';' | s ='10;20;30'  sub = s.split(';') | ['10', '20', '30'] |
| Retrieve the first integer from string s | index = sub.find(‘;’) s[:index] | 2  '10' |
| Retrieve the second integer from string s | temp = s[:index]  index = temp.find(‘;’)  print(index)  print(Temp[:index]) | '20;30’  2  '20' |

Activity 4 (modifyString.py)

Write a Python program to

1. Prompt user to input the original string and substring to delete
2. Remove the first occurrence of the substring from the original string
3. Display the modified string



|  |
| --- |
| string = str(input("Enter original string: "))  new\_string = string.replace('or', '',1)  print(new\_string) |

**PART 2 - Lecture 3-2 (Lists)**

Activity 1

Given the lists **first\_list** and **second\_list** are created as follows:

**first\_list = [ 1, 2, 3, 4, 5 ]**

**second\_list = [ 2, 3, 4 ]**

Evaluate the output in the following:

|  |  |
| --- | --- |
| **Program Text** | **Output** |
| print(first\_list[1]) | 2 |
| print(first\_list[-1]) | 5 |
| print(first\_list[5]) | IndexError: Index out of range |
| print(first\_list[1:3]) | [2,3] |
| print(first\_list[:3]) | [1,2,3] |
| print(first\_list[3:]) | [3,4,5] |
| print(first\_list) | [1,2,3,4,5] |
| print(len(first\_list)) | 5 |
| print(min(first\_list[1:3]) | 2 |
| print(first\_list != second\_list) | True |
| print(first\_list[1:4] == second\_list) | True |
| thirdList = first\_list + second\_list  print(thirdList) | [1, 2, 3, 4, 5, 2, 3, 4] |
| print(4 in first\_list) | True |
| print(4 not in second\_list) | False |
| print(second\_list in first\_list) | True |
| print(second\_list[1] in first\_list[1:3]) | True |

Activity 2 (friends.py)

Write a Python program to

1. Create a list of strings friends and initialize with the following string elements "Izzat", "Bryan", "Dalton", "Ethan", "Isaac"
2. Prompt the user to input a new name. Add this new name to friends and print the new friends list
3. Prompt the user to input an existing name.
4. Show the index of this name in the list
5. Remove the name from friends
6. Print out the new friends list

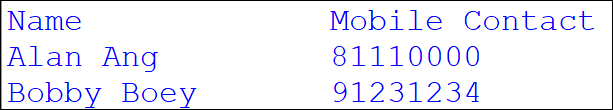
|  |
| --- |
| friends = ["Izzat", "Bryan", "Dalton", "Ethan", "Isaac"]  new\_friend = str(input("Enter the name of the new friend: "))  friends.append(new\_friend)  print(friends)  delete\_friend = str(input("Enter the name of friend to delete: "))  delete\_friend in friends  index = friends.index(delete\_friend)  print(str("{} is at index {} in the friendslist".format(delete\_friend, index)))  friends.pop(index)  print(friends) |

Activity 3 (phone.py)

Copy the text file, StudentData.txt (from Week 2 Exercise Activity 9), that contains 5 of your friends’ names together with their mobile numbers as shown in screenshot below.

|  |
| --- |
| path = 'C://pythonresource//'  datafile = open(path + 'StudentData.txt', 'r')  print(‘Name’ + ‘\t’ + ‘Mobile Contact’) #1st student  student = datafile.read()  list = student.split(‘,’)  student.strip(‘\n’)  print(list[0], ‘\t’, list[1]) #2nd student  student = datafile.read() student.strip(‘\n’)  list = student.split(‘,’)  print(list[0], ‘\t’, list[1])  file.close |

Write a program that reads the first 2 lines of data from the file and displays in the format shown.



**Hint**: Use String *split()* method to split a string into a list using a specified separator *string*.split(separator).

E.g. line = " Alan Ang,81110000"

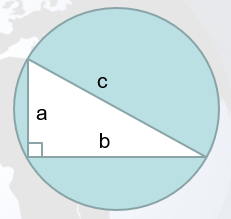
line.split(',') [' Alan Ang','81110000']

**PART 3 - Lecture 3-3 (Intro to Functions)**

Activity 1

Rewrite the code (in Week 2 Activity 5) to define a function to pass in the length values a and b and return the area of the circle.

The function should have the following heading:

calculate\_circle(a, b)

Call the function to calculate the area of the circle and display the value.

Sample output (values underlined are the user input):

|  |
| --- |
| Enter the length of a: 2.5  Enter the length of b: 4.6  The area of the circle is 21.52776365872405  import math  x = float(input("Enter the length of a: "))  y = float(input("Enter the length of b: "))  def calculate\_circle(a, b):  c = math.sqrt(a\*\*2 + b\*\*2) \* 0.5  area = math.pi\*(c\*\*2)  return(area)  area = calculate\_circle(x,y)  print (area) |

-- End of Exercise --