**Lab#9**

**Questions**

Q # 1: Write a program that takes a sentence as input from the user and uses the split

() function to split the sentence into a list of words. Print the list.

Input:

sentence = input("Enter a sentence: ")

word\_list = sentence.split()

print(word\_list)

Output:

Enter a sentence: This is a sample sentence.

['This', 'is', 'a', 'sample', 'sentence.']

Q # 2: "lab9\_2.py" to ask the user for a specific word. Use the replace() function to replace all occurrences of that word with "REPLACED". Print the modified sentence.

Input:

sentence = input("Enter a sentence: ")

word\_to\_replace = input("Enter a word to replace: ")

modified\_sentence = sentence.replace(word\_to\_replace, "REPLACED")

print(modified\_sentence)

Output:

Enter a sentence: This is a sample sentence.

Enter a word to replace: sample

This is a REPLACED sentence.

Q # 3: Write a program that takes a string containing comma-separated values (CSV)

as input and uses the split() function to separate the values into a list. Then, print the list.

Input:

csv\_string = input("Enter comma-separated values: ")

values\_list = csv\_string.split(',')

print(values\_list)

Output:

Enter comma-separated values: apple,banana,orange,grape

['apple', 'banana', 'orange', 'grape']

Q # 4: Write a program that takes a string as input from the user and uses the split()

function to split the string into words. Then, use a dictionary to count the frequency of each word in the string.

Input:

sentence = input("Enter a sentence: ")

word\_list = sentence.split()

word\_frequency = {}

for word in word\_list:

if word in word\_frequency:

word\_frequency[word] += 1

else:

word\_frequency[word] = 1

print(word\_frequency)

Output:

Enter a sentence: This is a sample sentence. This sentence has sample words.

{'This': 2, 'is': 1, 'a': 1, 'sample': 2, 'sentence.': 1, 'sentence': 1, 'has': 1, 'words.': 1}

Q # 5: To check for a duplicate key before saving a new item in the dictionary. In

case an item is already available in the students\_dict, then the program should warn the user.

Input:

students\_dict = {}

while True:

student\_name = input("Enter student's name: ")

if student\_name in students\_dict:

print("Warning: Student already exists in the dictionary!")

else:

student\_score = input("Enter student's score: ")

students\_dict[student\_name] = student\_score

another = input("Add another student? (yes/no): ")

if another.lower() != 'yes':

break

print(students\_dict)

Output:

Enter student's name: John

Enter student's score: 90

Add another student? (yes/no): yes

Enter student's name: Alice

Enter student's score: 85

Add another student? (yes/no): yes

Enter student's name: John

Warning: Student already exists in the dictionary!

Enter student's score: 88

Add another student? (yes/no): no

{'John': '90', 'Alice': '85'}

Q # 6: Write a program that takes a sentence as input and uses the split() function to

split the sentence into words. Then, reverse the order of words in the sentence and print the modified sentence.

Input:

sentence = input("Enter a sentence: ")

word\_list = sentence.split()

reversed\_sentence = ' '.join(word\_list[::-1])

print(reversed\_sentence)

Output:

Enter a sentence: This is a sample sentence.

sentence. sample a is This

**QUESTIONS:**

Q # 1: What is the purpose of the split() function in Python?

Ans: The purpose of the split() function in Python is to split a string into a list of substrings based on a specified delimiter. By default, the delimiter is whitespace, but it can be customized to split the string based on other characters as well.

Q # 2: Explain the difference between the replace() and split() functions in Python.

Ans: The replace() function is used to replace occurrences of a specified substring within a string with another substring. It returns a new string with the replacements made. On the other hand, the split() function is used to split a string into a list of substrings based on a specified delimiter. It does not modify the original string but returns a list of substrings.

Q # 3: How can you split a string into a list of words using the split() function with a custom Delimiter?

Ans: To split a string into a list of words using the split() function with a custom delimiter, you can pass the custom delimiter as an argument to the split() function.

Name: Jawaria

Roll : 23BS(AI)39

Date: 3-march-2024