DEPARTMENT OF COMPUTER SCIENCE & TECHNOLOGY

Course: B.Tech CSE/AIML/CSTI/FSD
Subject: Python Programming (CSH108B-T) & (CSH108B-P)

Lab 4

***Course Outcome*:**

CSW108B.1: To **impart** understanding of basic programming concepts in python language.

CSW108B.2: To enable the student to articulate given program scenario and **apply** different programming constructs.

Learning outcome:

Students will be able to do hands-on practice of Strings in Python

***Blooms Taxonomy Level**: BT1, BT2, BT3*

***Blooms Taxonomy Level**: BT2, BT3*

1. WAP to demonstrate Slicing Operations in Strings
   Ans:

```
# Defining a string
text = "Hello, Python!"

# Slicing operations

# Slicing from index 0 to 5 (not including 5)
print("Slicing from 0 to 5:", text[0:5])

# Slicing from index 7 to the end
print("Slicing from 7 to the end:", text[7:])

# Slicing from the beginning to index 5 (not including 5)
print("Slicing from beginning to 5:", text[:5])

# Slicing the entire string
print("Slicing the entire string:", text[:])

# Slicing with a step value (every 2nd character)
print("Slicing with step of 2:", text[::2])

# Slicing with a negative index (from the end)
print("Slicing from -6 to -1:", text[-6:-1])
```

```
# Reversing the string using slicing
print("Reversed string:", text[::-1])
```

2. WAP to demonstrate built in functions of Strings.

Ans:

```
# Defining a string
text = "  Hello, Python World!  "

# 1. Convert to Uppercase
print("Uppercase:", text.upper())

# 2. Convert to Lowercase
print("Lowercase:", text.lower())

# 3. Strip leading and trailing spaces
print("Stripped:", text.strip())

# 4. Replace a substring
print("Replace 'Python' with 'Java':", text.replace("Python", "Java"))

# 5. Split the string into a list
print("Split into words:", text.split())

# 6. Find the index of a substring (returns the index of the first occurrence)
print("Index of 'Python':", text.find("Python"))

# 7. Count occurrences of a substring
print("Count of 'o':", text.count("o"))

# 8. Check if the string starts with a specific substring
print("Starts with '  Hello':", text.startswith("  Hello"))

# 9. Check if the string ends with a specific substring
print("Ends with 'World!  ':", text.endswith("World!  "))

# 10. Check if the string is alphanumeric (contains only letters and numbers)
alphanumeric_text = "Python123"
print("Is alphanumeric:", alphanumeric_text.isalnum())

# 11. Check if the string contains only alphabetic characters
print("Is alphabetic:", "Python".isalpha())

# 12. Check if the string contains only digits
digit_text = "12345"
print("Is digit:", digit_text.isdigit())

# 13. Join a list of strings with a separator
words = ["Python", "is", "great"]
print("Joined words:", " ".join(words))
```

```
# 14. Check the length of the string
print("Length of the string:", len(text))

# 15. Convert the string to title case
print("Title Case:", text.title())
```

3. WAP to check weather a given string is palindrome or not.

Ans:

```
# Input: Getting a string from the user
text = input("Enter a string: ")

# Remove spaces and convert to lowercase for a case-insensitive comparison
cleaned_text = text.replace(" ", "").lower()

# Check if the string is equal to its reverse
if cleaned_text == cleaned_text[::-1]:
    print(f"'{text}' is a palindrome.")
else:
    print(f"'{text}' is not a palindrome.")
```

4. WAP to capitalize the first and last character of each word in a string.

Ans:

```
# Input: Getting a string from the user
text = input("Enter a string: ")

# Function to capitalize the first and last character of each word
def capitalize_first_last(word):
    if len(word) > 1:
        return word[0].upper() + word[1:-1] + word[-1].upper()
    elif len(word) == 1:
        return word.upper()
    else:
        return word

# Split the text into words, capitalize first and last character of each word, and join them back
modified_text = ' '.join([capitalize_first_last(word) for word in text.split()])

# Output: Display the modified string
print("Modified string:", modified_text)
```

5. WAP to accept two strings from the user and display the common words.

Ans:

```
# Accept two strings from the user
string1 = input("Enter the first string: ")
string2 = input("Enter the second string: ")

# Split the strings into lists of words
words1 = set(string1.split())
words2 = set(string2.split())

# Find the common words between the two sets
common_words = words1.intersection(words2)
```

```
# Display the common words
if common_words:
    print("Common words:", ", ".join(common_words))
else:
    print("No common words found.")
```

6. WAP to accept a string and count the frequency of each vowel.

   Ans:

```
# Accept a string from the user
text = input("Enter a string: ")

# Initialize a dictionary to store the frequency of vowels
vowels = {'a': 0, 'e': 0, 'i': 0, 'o': 0, 'u': 0}

# Convert the string to lowercase to make the counting case-insensitive
text = text.lower()

# Loop through the string and count vowels
for char in text:
    if char in vowels:
        vowels[char] += 1

# Display the frequency of each vowel
print("Vowel frequencies:")
for vowel, count in vowels.items():
    print(f"{vowel.upper()}: {count}")
```

7. WAP to display the smallest word from the string.

ans:

```
# Accept a string from the user
text = input("Enter a string: ")

# Split the string into words
words = text.split()

# Find the smallest word based on length
smallest_word = min(words, key=len)

# Display the smallest word
print("The smallest word is:", smallest_word)
```