CHRIST THE KING BOYS MATRIC HR. SEC. SCHOOL, KUMBAKONAM – 612 001

CHAPTER – 8

STRINGS AND STRING MANIPULATION

1. What are Strings?

- ☆ String is a data type in python, which is used to handle **array of characters.**
- ☆ String is a sequence of Unicode characters that may be a combination of letters, numbers, or special symbols enclosed within **single, double or even triple quotes.**
- ☆ Strings are immutable in Python.
- ☆ **Eg:**

  'Welcome to learning Python'

  "Welcome to learning Python"

  '''Welcome to learning Python'''

2. Define Subscript and its types

- ☆ Index values are otherwise called as **subscript** which are used to access and manipulate the strings.
- ☆ The subscript can be **positive** or **negative** integer numbers.
- ☆ The positive subscript **0** is assigned to the first character and **n-1** to the last character, where n is the number of characters in the string.
- ☆ The negative index assigned from the last character to the first character in reverse order begins with **-1**.
- ☆ **Eg:**

String	C	O	M	P	U	T	E	R
+ve Index	0	1	2	3	4	5	6	7
-ve Index	-8	-7	-6	-5	-4	-3	-2	-1

3. Write the General format of replace function with example?

- ☆ **General Format:**

  replace("char1", "char2")

  The replace function replaces all occurrences of char1 with char2.
- ☆ **Eg:**

```
str1="How are you"
print (str1)
```

- ☆ **Output:**

How are you

print (str1.replace("o", "e"))

☆ **Output:**

Hew are yeu

4. What are the String Operators used in Python?

☆ Concatenation Operator

☆ Append Operator

☆ Repeating Operator

☆ String Slicing Operator

☆ Stride when Slicing String Operator

5. Define Concatenation Operator with Example

☆ **Joining** of two or more strings is called as **Concatenation**.

☆ The **plus (+)** operator is used to concatenate strings in python.

☆ **Eg:**

"welcome" + "Python"

☆ **Output:**

'welcomePython'

6. Define Append Operator with Example

☆ Adding more strings at the end of an existing string is known as **Append.**

☆ The operator **+= is used to append a new string** with an existing string.

☆ **Eg:**

str1="Welcome to"

str1+="Learn Python"

print (str1)

☆ **Output:**

Welcome to Learn Python

7. Define Repeating Operator with Example

☆ The **Multiplication Operator (*)** is used to display a string in multiple number of times.

☆ **Eg:**

```
str1="Welcome"
print (str1*4)
```

☆ **Output:**

```
Welcome Welcome Welcome Welcome
```

8. Define String Slicing Operator with Example

☆ **Slice** is a substring of a main string.

☆ A substring can be taken from the original string by using **[ ] operator** and index or subscript values. Thus, [ ] is also known as **Slicing Operator.**

☆ Using slice operator, you have to slice one or more substrings from a main string.

☆ **General format of slice operation:**

str[start:end]

☆ Where **start** is the beginning index and **end** is the last index value of a character in the string. Python takes the end value less than one from the actual index specified.

☆ **Eg:** if you want to slice first 4 characters from a string, you have to specify it as 0 to 5. Because, python consider only the end value as n-1.

9. Define Stride when Slicing String Operator with Example

☆ When the slicing operation, you can specify a third argument as the stride, which refers to the number of characters to move forward after the first character is retrieved from the string. The default value of stride is 1.

☆ **Eg:**

```
str1 = "Welcome to learn Python"
print (str1[10:16])
learn
print (str1[10:16:4])
r
print (str1[10:16:2])
er
print (str1[::3])
Wceoenyo
```

10. Write the Syntax and Example for String Formatting?

☆ **Syntax:**

("String to be display with %val1 and %val2" %(val1, val2))

☆ **Eg:**

```
name = "Rajarajan"
mark = 98
print ("Name: %s and Marks: %d" %(name,mark))
```

☆ **Output**

Name: Rajarajan and Marks: 98

11. List some String Formatting Characters with Usage?

FORMAT CHARACTERS	USAGE
%c	Character
%d (or) %i	Signed decimal integer
%s	String
%u	Unsigned decimal integer
%o	Octal integer

12. Define Escape Sequence with Python

☆ Escape sequences starts with a **backslash** and it can be interpreted differently.

☆ When you have use single quote to represent a string, all the **single quotes** inside the string must be **escaped**. Similar is the case with double quotes.

☆ **Eg:**

```
print ('They said, "What\'s there?"')
```

They said, "What's there?"

13. List some of the Escape Sequences in Python?

ESCAPE SEQUENCE	DESCRIPTION

\newline	Backslash and newline ignored
\\	Backslash
\'	Single quote
\"	Double quote
\a	ASCII Bell

14. Define format() Function with Example

☆ The format( ) function used with strings is very versatile and powerful function used for formatting strings.

☆ The curly braces { } are used as placeholders or replacement fields which get replaced along with format( ) function.

☆ **Eg:**

```
num1=int (input("Number 1: "))
num2=int (input("Number 2: "))
print ("The sum of { } and { } is { }".format(num1, num2,(num1+num2)))
```

⊕ **Output:**

```
Number 1: 34
Number 2: 54
The sum of 34 and 54 is 88
```

15. Define the following Built – in String Functions:

a) len() **b) capitalize()**

a) len():

⊕ It is used to return the length of the string.

⊕ **Eg:**

```
A=”Computer”
print(len(A))
```

☆ **Output:**

```
8
```

b) capitalize():

☆ It is used to capitalize the first character of the string.

☆ **Eg:**

A=”computer”

print(A.capitalize())

☆ **Output:**

Computer

16. Define the following Built – in String Functions:

a) isalnum() **b) isalpha()** **c) isdigit()**

a) isalnum()

☆ It is used to return **true** if the string contains only letter and digits. It returns **false** if the string contains any special characters.

☆ **Eg 1:**

A=”computer”

print(A.isalnum())

☆ **Output:**

True

☆ **Eg 2 :**

A=”#computer”

print(A.isalnum())

☆ **Output:**

False

b) isalpha()

☆ It is used to return **true** if the string contains only letters otherwise it returns **false.**

☆ **Eg 1:**

A=”computer”

print(A.isalpha())

☆ **Output:**

True

⊕ **Eg 2:**

A=”#computer”

print(A.isalnum())

☆ **Output:**

False

c) isdigit()

☆ It is used to return **true** if the string contains only numbers otherwise it returns **false.**

☆ **Eg 1:**

```
A=”computer123”
print(A.isalpha())
```

☆ **Output:**

```
True
```

⊕ **Eg 2:**

```
A=”computer”
print(A.isalnum())
```

☆ **Output:**

```
False
```

17. Define the following Built – in String Functions:

a) isupper() **b) islower()**

a) isupper()

☆ It is used to return **true** if the string is in Uppercase.

☆ **Eg 1:**

```
A=”computer”
print(A.isupper())
```

☆ **Output:**

```
False
```

⊕ **Eg 2:**

```
A=”COMPUTER”
print(A.isupper())
```

☆ **Output:**

```
True
```

b) islower()

☆ It is used to return **true** if the string is in Lowercase.

☆ **Eg 1:**

```
A=”computer”
print(A.islower())
```

☆ **Output:**

True

⊕ **Eg 2:**

A=”COMPUTER”

print(A.islower())

☆ **Output:**

False

18. Define the following Built – in String Functions:

a) upper() **b) lower()**

a) upper()

☆ It is used to return the given string in Uppercase.

⊕ **Eg:**

A=”computer”

print(A.upper())

☆ **Output:**

COMPUTER

b) lower()

☆ It is used to return the given string in Lowercase.

⊕ **Eg:**

A=”COMPUTER”

print(A.lower())

☆ **Output:**

computer

19. Define the following Built – in String Functions:

a) title() **b) swapcase()**

a) title()

☆ It is used to return the given string in Title Case.

⊕ **Eg:**

A=”computer science”

print(A.title())

☆ **Output:**

Computer Science

b) swapcase()

☆ It is used to change the case of every character to its opposite case vice versa.

⊕ **Eg:**

```
A=”cOmPuTeR”
print(A.swapcase())
```

☆ **Output:**

```
CoMpUtEr
```

20. Define the following Built – in String Functions:

a) ord() **b) chr()**

a) ord()

☆ It is used to return the ASCII code of the character.

⊕ **Eg:**

```
A=’A’
print(ord(A))
```

☆ **Output:**

```
65
```

b) chr()

☆ It is used to return the character represented by a ASCII code.

⊕ **Eg:**

```
A=97
print(chr(A))
```

☆ **Output:**

```
a
```

21. Define Membership Operators

⊕ The ‘**in**’ and ‘**not in**’ operators can be used with strings to determine whether a string is present in another string.

⊕ These operators are called as **Membership Operators.**

⊕ **Eg:**

```
str1=input ("Enter a string: ")
str2="chennai"
if str2 in str1:
```

```
        print ("Found")
    else:
        print ("Not Found")
```

⊕ **Output :**

Enter a string: Chennai G HSS, Saidapet

Found

22. Write a Python Program to check whether the given string is palindrome or not

⊕ **Program:**

```
str1 = input ("Enter a string: ")
str2 = ' '
index=-1
for i in str1:
    str2 += str1[index]
    index -= 1
print ("The given string = { } \n The Reversed string = { }".format(str1, str2))
if (str1==str2):
    print ("Hence, the given string is Palindrome")
else:
    print ("Hence, the given is not a palindrome")
```

⊕ **Output 1:**

Enter a string: malayalam

The given string = malayalam

The Reversed string = malayalam

Hence, the given string is Palindrome

⊕ **Output 2:**

Enter a string: welcome

The given string = welcome

The Reversed string = emoclew

Hence, the given string is not a palindrome

23. Write a Python Program to display the following pattern

```
*
* *
* * *
* * * *
* * * * *
```

⊕ **Program:**

```
str1=' * '
i=1
while i<=5:
    print (str1*i)
    i+=1
```

⊕ **Output:**

```
*
* *
* * *
* * * *
* * * * *
```

24. Write a Python Program to display the number of vowels and consonants in the given string

⊕ **Program:**

```
str1=input ("Enter a string: ")
str2="aAeEiIoOuU"
v,c=0,0
for i in str1:
    if i in str2:
        v+=1
    else:
        c+=1
print ("The given string contains { } vowels and { } consonants".format(v,c))
```

⊕ **Output:**

Enter a string: Tamilnadu School Education

The given string contains 11 vowels and 15 consonants

25. **Write a Python Program to create an Abecedarian series. (Abecedarian refers list of elements appear in alphabetical order)**

⊕ **Program:**

```
str1="ABCDEFGH"
str2="ate"
for i in str1:
        print ((i+str2),end='\t')
```

⊕ **Output**

Aate Bate Cate Date Eate Fate Gate Hate

26. **Write a Python Program that accept a string from the user and display the same after removing vowels from it**

⊕ **Program:**

```
def rem_vowels(s):
    temp_str=''
    for i in s:
        if i in "aAeEiIoOuU":
            pass
        else:
            temp_str+=i
    print ("The string without vowels: ", temp_str)
str1= input ("Enter a String: ")
rem_vowels (str1)
```

⊕ **Output**

Enter a String: Mathematical foundations of Computer Science

The string without vowels: Mthmtcl fndtns f Cmptr Scnc

27. Write a Python Program that count the occurrences of a character in a string

⊕ **Program:**

```
def count(s, c):
    c1=0
    for i in s:
        if i == c:
            c1+=1
    return c1
str1=input ("Enter a String: ")
ch=input ("Enter a character to be searched: ")
cnt=count (str1, ch)
print ("The given character {} is occurs {} times in the given string".format(ch,cnt))
```

⊕ **Output:**

```
Enter a String: Software Engineering
Enter a character to be searched: e
The given character e is occurs 3 times in the given string
```