Topics:

- **UNIT – II [14 Lectures]**
- **Data Structures:** Sequence, Lists, Tuple, Sets, Dictionaries
- **Strings and its operations:** Concatenating, Appending, Multiplying strings, Built-in String methods and functions, Slice Operation, Iterating String, String Module
- **Modules:** Importing module, The from..import statement, Name of Module, Making your own modules, The dir()function, The Python Module,Math module, OS Module, Sys Module, Random module
- **Introduction to Functions:** Declaration and Definition, Variable Scope and Lifetime, Return Statements, Types of Arguments, Lambda function, Recursion
- **Functional Programming:** filter() function, map()function, reduce()function

Topics:

- **Strings and its operations**
  - **Concatenating**
  - **Appending**
  - **Multiplying strings**
  - **Built-in String methods and functions**
  - **Slice Operation**
  - **Iterating String**
  - **String Module**

Strings

- In Python, string is a sequence of characters, where a character could be a letter,digit,whitespace or any other symbol
- **Create a string**
- Strings can be created by enclosing characters inside a single quote(‘), double quotes (“) and triple code (‘ ‘ ‘ or “ “ “)
- It must start and end with same type of quote
- Tripe quotes are used to span string across multiple lines
- Python has built-in string class named “str” that has many useful features
- **The index of the first character is 0 and the index of the last character is n-1, where n is the number of characters in the string.** Can be accessed using negative indices. Last character will start with -1 and traverses from right to left

```
#Syntax for Creating a sting
name="RGUKT"
location='''''Basar Temple'''''
Dist='Nirmal'
mandal=str("Basar,")
print(name, location, Dist,mandal)


word = 'Python Programming'
sentence = "Object Oriented Programming"
paragraph ='''''Python is a Object Oriented Programming Language'''''
feedback=str('It is a Biginner's Language')
print(word,sentence,paragraph,feedback)

my_string = 'Hello'
print(my_string)
my_string1 = input("Enter a string")
print(my_string1)
my_string2 = """Hello, welcome to the world of Python"""
print(my_string2)
```

- **access characters in a string**
- We can access individual characters using indexing,using the subscript([ ]) operator and a range of characters using slicing. Index starts from 0. Trying to access a character out of index range(below 0 or above n-1) will raise an IndexError.
- The index must be an integer. We can't use float or other types, this will result into TypeError.
- Python also allows negative indexing for its sequences.

Example

```
str = 'program'
i=2
print('str = ', str)
#first character
print('str[0] = ', str[0])
#last character
print('str[-1] = ', str[-1])
#slicing 2nd to 5th character
print('str[1:5] = ', str[1:5])
print( str[i])
print(str[i*2+2])
```

- **change or delete a string**
- Strings are immutable. This means that elements of a string cannot be changed once it has been assigned. We can simply reassign different strings to the same name.
- We cannot delete or remove characters from a string. But deleting the string entirely is possible using the keyword del.
- **Example**

```
my_string = 'perl'
my_string = 'Python'
print(my_string)
del my_string
print(my_string)
```

- **Concatenation of Two or More Strings**
- Joining of two or more strings into a single one is called concatenation. The + operator does this in Python.
- The * operator can be used to repeat the string for a given number of times.
- **Example**

```
# using +
str1 = 'Hello'
str2 ='World!'
str3=str1+str2
print("The Concatenated string is:", str3)
# using *
str="Hello hw R U"
print('str * 3 =', str * 3)
```

- Append mean to add something at the end. In Python you can add one string at the end of another string using the += operator

```
str="Hello, "
name=input("\n Enter Your name:")
str+=name
str+=" Welcome to Python Programming"
print(str)
```

- The **str( )** function is used to convert values of any other type into string type. This helps the programmer to concatenate a string with any other data which is otherwise not allowed

```
str1="Hello"
var=7
str2=str1+var
print("str2",str2)
```

```
str1="hello"
var=7
str2=str1+str(var)
print("str2",str2)
```

- The print statement prints one or more literals or values in a new line. If you don't want to print on a new line then, add end statement with a separator like whitespace, comma etc.

```
print("Hello")
print("world")
```

```
print("Hello",end='')
print("world")
```

Strings are Immutable

- Python strings are immutable which means that once created they cannot be changed. Whenever you try to modify an existing string variable, a new string is created
- Every object in python is stored in memory. You can find out whether two variables are referring to the same object or not by using id( ).
- The id( ) returns the memory address of that object. As both str1 and str2 points to same memory location, they both point to the same object

```
# Program to demonstrate id() function

str1="Hello"
print("str1 is :",str1)
print("Id of str1 is:",id(str1))

str2="World"
print("str2 is :",str2)
print("id of str2 is:",id(str2))
str1+=str2
print("str1 after concatenation is:",str1)
print("id of str1:",id(str1))
str3=str1
print("str3=",str3)
print("id of str3 is:",id(str3))
```

```
str="hai"
str[0]='o'
print(str)


str1="Hai"
new_str="O"
print("old string",str1)
print("New string", new_str)
```

For Details Contact Me @ :

9247448766

ravikanth27787@gmail.com

python

Programming Language