MODULE 9: FOR LOOP
WEEK 9

Learning Outcomes:

After completing this course you are expected to demonstrate the following:

The student will have used For Loops to iterate over a block of code as long as the test expression (condition) is true.

A. Engage

Memes Station

Figure 8.0: Friend to the rescue.

Loop a series of instructions that is repeated until a terminating condition is reached.

B. Explain

In computer science, a loop is a programming structure that repeats a sequence of instructions until a specific condition is met. Programmers use loops to cycle through values, add sums of numbers, repeat functions, and many other things.

Loops are supported by all modern programming languages, though their implementations and syntax may differ. Two of the most common types of loops are the **while loop** and the **for loop**.

C. Explore

Video Title:

1. Python 3 Programming Tutorial - For loop

YouTube Link:

1. https://www.youtube.com/watch?v=dxwT6cq8tlM

Module Video Filename:

1. WEEK9_Python 3 Programming Tutorial - For loop

D. Elaborate

Introduction

Python for loop

Like most other languages, Python has for loops, but it differs a bit from other like C or Pascal. In Python for loop is used to iterate over the items of any sequence including the Python list, string, tuple etc. The for loop is also used to access elements from a container (for example list, string, tuple) using built-in function range().

Syntax:

```
for variable_name in sequence:
   statement_1
   statement_2 ....
```

Name	Description
variable_name	It indicates target variable which will set a new value for each iteration of the loop.
sequence	A sequence of values that will be assigned to the target variable variable_name. Values are provided using a list or a string or from the built-in function range().
statement_1, statement_2	Block of program statements.

Example: Python for loop

```
>>> #The list has four elements, indices start at 0 and end at 3
>>> color_list = ["Red", "Blue", "Green", "Black"]
>>> for c in color_list:
      print(c)

 Red
 Blue
 Green
 Black
>>>
```

In the above example color_list is a sequence contains a list of various color names. When the for loop executed the first item (i.e. Red) is assigned to the variable c. After this, the print statement will execute and the process will continue until we reach the end of the list.

Python for loop and range() function

The range() function returns a list of consecutive integers. The function has one, two or three parameters where last two parameters are optional. It is widely used in for loops. Here is the syntax.

range(a)
range(a,b)
range(a,b,c)

range(a): Generates a sequence of numbers from 0 to a, excluding a, incrementing by 1.

Syntax:
for <variable> in range(<number>):

Example:

```
>>> for a in range(4):
 print(a)

 0
 1
 2
 3
>>>
```

range(a,b): Generates a sequence of numbers from a to b excluding b, incrementing by 1.

Syntax:
for "variable" in range("start_number", "end_number"):

Example:

```
>>> for a in range(2,7):
 print(a)

 2
 3
 4
 5
 6
>>>
```

range(a,b,c): Generates a sequence of numbers from a to b excluding b, incrementing by c.

Example:

```
>>> for a in range(2,19,5):
 print(a)

 2
 7
 12
 17
>>>
```

Python for loop: Iterating over tuple, list, dictionary

Example: Iterating over tuple

The following example counts the number of even and odd numbers from a series of numbers.

```
numbers = (1, 2, 3, 4, 5, 6, 7, 8, 9) # Declaring the tuple
count_odd = 0
count_even = 0
for x in numbers:
    if x % 2:
        count_odd+=1
    else:
        count_even+=1
print("Number of even numbers :",count_even)
print("Number of odd numbers :",count_odd)
```

Output:

```
Number of even numbers:4
Number of odd numbers: 5
```

In the above example a tuple named numbers is declared which holds the integers 1 to 9. The best way to check if a given number is even or odd is to use the modulus operator (%). The operator returns the remainder when dividing two numbers.

Modulus of 8 % 2 returns 0 as 8 is divided by 2, therefore 8 is even and modulus of 5 % 2 returns 1 therefore 5 is odd.

The **for loop** iterates through the tuple and we test modulus of x % 2 is true or not, for every item in the tuple and the process will continue until we rich the end of the tuple. When it is true count_even increase by one otherwise count_odd is increased by one. Finally, we print the number of even and odd numbers through print statements.

Example: Iterating over list

In the following example for loop iterates through the list "datalist" and prints each item and its corresponding Python type.

```
datalist = [1452, 11.23, 1+2j, True, 'w3resource', (0, -1), [5, 12],
{"class":'V', "section":'A'}]
for item in datalist:
  print ("Type of ",item, " is ", type(item))

Type of  1452  is  <class 'int'>
Type of  11.23  is  <class 'float'>
Type of  (1+2j)  is  <class 'complex'>
Type of  True  is  <class 'bool'>
Type of  w3resource  is  <class 'str'>
Type of  (0, -1)  is  <class 'tuple'>
Type of  [5, 12]  is  <class 'list'>
Type of  {'section': 'A', 'class': 'V'}  is  <class 'dict'>
```

Example: Iterating over dictionary

In the following example for loop iterates through the dictionary "color" through its keys and prints each key.

```
>>> color = {"c1": "Red", "c2": "Green", "c3": "Orange"}
>>> for key in color:
  print(key)

c2
c1
c3
>>>
```

Following for loop iterates through its values:

```
>>> color = {"c1": "Red", "c2": "Green", "c3": "Orange"}
>>> for value in color.values():
  print(value)

Green
Red
Orange
>>>
```

You can attach an optional else clause with for statement, in this case, syntax will be –

```
for variable_name in sequence:
   statement_1
   statement_2
   ....
else :
   statement_3
   statement_4
   ....
```

The else clause is only executed after completing the for loop. If a break statement executes in first program block and terminates the loop then the else clause does not execute.

E. Evaluate

ASSESSMENT:

Instructions: You may write your answer on the Answer Sheet (AS) provided on this module.

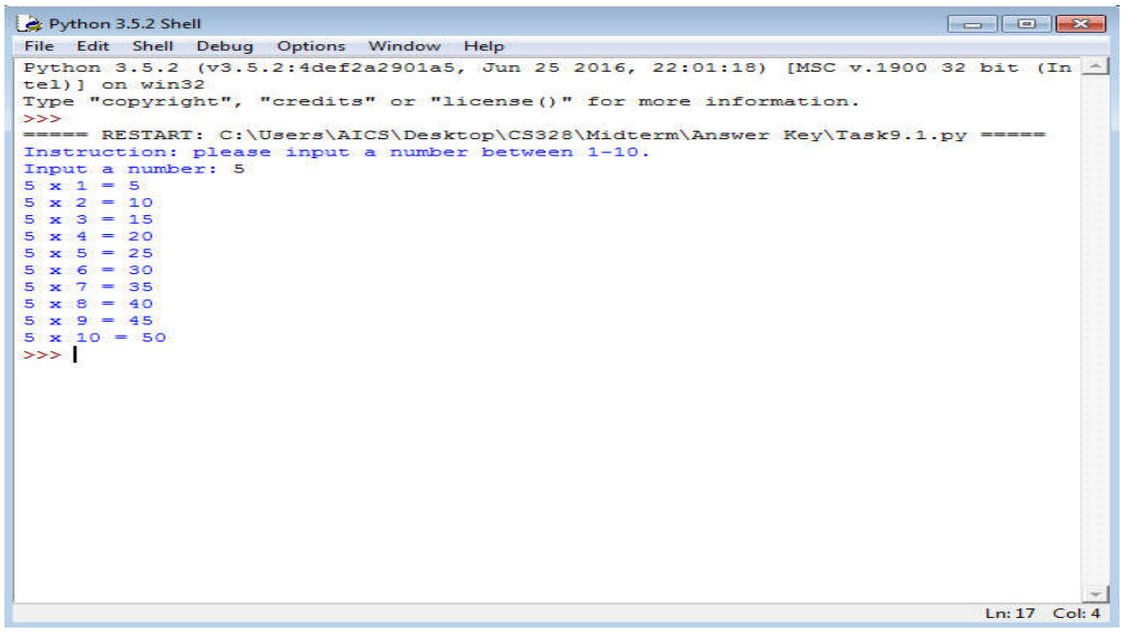
I. Matching Type (2-Point Each):

1. Block of program statements.
2. It is used to iterate over the items of any sequence including the Python list, string, tuple etc.
3. It indicates target variable which will set a new value for each iteration of the loop.
4. The function that returns a list of consecutive integers.
5. Generates a sequence of numbers from a to b excluding b, incrementing by 1.
6. It is values that will be assigned to the target variable variable_name.
7. The best way to check if a given number is even or odd is to use the?
8. Generates a sequence of numbers from a to b excluding b, incrementing by c.
9. it is true when increase by one otherwise count_odd is increased by one.
10. Generates a sequence of numbers from 0 to a, excluding a, incrementing by 1.

A. sequence
B. modulus operator
C. for loop
D. variable_name
E. range(a,b)
F. range(a,b,c)
G. count_even
H. statement_1, statement_2 .....
I. range()
J. range(a)

Hands-On Activity 9.1

Write a program in script mode that prompts the user to enter a number then it will create an multiplication table (from 1 to 10) of a number.



References:

1. **Python For Loops–**https://www.w3resource.com/python/python-for-loop.php

Facilitated By:		
Name	:	
MS Teams Account (email)	:	
Smart Phone Number	:	