LAB # 05

ITERATION WITH LOOPS

OBJECTIVE

To get familiar with the different types of loops.

THEORY

- ✓ A loop can be used to tell a program to execute statements repeatedly.
- ✓ In other word, to keep a computer doing useful work we need repetition, looping back over the same block of code again and again.
- ✓ Loops are used to repeat actions efficiently.

Type of Loop Statements in Python:

The main types are For loops (counting through items) and While loops (based on conditions. They are,

1. for
2. while

The for loop:

- ✓ It is used to iterate over a sequence such as a list, tuple, string or range.
- ✓ It allow to execute a block of code repeatedly, once for each item in the sequence
- ✓ A Python **for** loop iterates through each value in a sequence.

Syntax:

In general, the syntax of a **for** loop is:

```
for var in sequence:
      # Loop body
```

For loop can be used to simplify the preceding loop:

```
a) for i in range(endValue):
        #Loop body

b) for i in range(initialValue, endValue):
       # Loop body

c) for i in range(initialValue, endValue,k):   #k=step value
       # Loop body
```

LAB # 05

How for loop works:

- A ‘sequence’ holds multiple items of data, stored one after the other.
- In sequence introduce strings and data storing techniques.
- They are sequence-type objects in Python.
- The variable ‘**Var’** takes on each successive value in the sequence.
- The statements in the body of the loop are executed once for each value.
- Generate a sequence of numbers using ‘**range() function’**, range(10) will generate numbers from 0 to 9 (10 numbers).

Example program for ‘for loop’:

```
#simple for loop using sequence(string)

for letter in 'Python':
  print ('Current Letter :', letter)
```

Output:

```
>>> %Run Lab_05.py

Current Letter : P
Current Letter : y
Current Letter : t
Current Letter : h
Current Letter : o
Current Letter : n

>>>
```

Example program for ‘for loop using range( )’:

```
#Simple for loop using range()
n = 4;
for i in range(0, n):
    print(i)
```

Output:

```
>>> %Run Lab_05.py

0
1
2
3

>>>
```

LAB # 05

Example program for ‘for loop using range( )’:

```
#Simple for loop using range()
p = 4;
for i in range(p):
   print(i)
```

Output:

```
>>> %Run Lab_05.py

0
1
2
3

>>>
```

Example program for using List, Tuple, and Dictionary.

```
#Simple for loop using range()

# Example 1: For loop with a LIST
li = ["apple", "banana", "cherry"]
print("List Example:")
for x in li:
   print(x)
print()  # for spacing

# Example 2: For loop with a TUPLE
tup = ("car", "bike", "bus")
print("Tuple Example:")
for x in tup:
   print(x)
print()

# Example 3: For loop with a DICTIONARY
d = {'a': 10, 'b': 20, 'c': 30}
print("Dictionary Example:")
for key in d:
   print(f"{key} : {d[key]}")
print()

# Example 4: For loop with a SET
set1 = {100, 200, 300}
print("Set Example:")
for x in set1:
   print(x)
```

Output:

```
>>> %Run Lab_05.py

List Example:
apple
banana
cherry

Tuple Example:
car
bike
bus

Dictionary Example:
a : 10
b : 20
c : 30

Set Example:
200
100
300

>>>
```

The break Statement

With the **break statement** we can stop the loop before it has looped through all the items:

Example program for ‘for loop using Break Statemnet’:

```
# List of fruits

fruits = ["apple", "banana", "cherry", "mango"]
print("---- Using 'break' ----")
for x in fruits:
    print(x)
    if x == "banana":
        break                                   # stops the loop completely
print()                                         # blank line for spacing
```

Output:

```
>>> %Run Lab_05.py
---- Using 'break' ----
apple
banana

>>>
```

LAB # 05

The continue Statement

With the continue statement we can stop the current iteration of the loop, and continue with the next:

Example program for ‘for loop using Break Statemnet’:

```
# List of fruits

fruits = ["apple", "banana", "cherry", "mango"]
print("---- Using 'continue' ----")
for x in fruits:
    if x == "banana":
        continue                    # skips this item, continues to next
    print(x)                        # blank line for spacing
```

Output:

```
>>> %Run Lab_05.py

---- Using 'continue' ----
apple
cherry
mango

>>>
```

Example program:

```
# Using If else with for loop
for x in range(6):  # Put value less than 3 then else will print
  if x == 3: break
  print(x)
else:
  print("Finally finished!")
```

Output:

```
>>> %Run Lab_05.py
0
1
2
>>>
```

Nested Loops

- ✓ A nested loop is a loop inside a loop.
- ✓ The "inner loop" will be executed one time for each iteration of the "outer loop":

LAB # 05

Example program:

```
# Lists
adj = ["red", "big", "tasty"]
fruits = ["apple", "banana", "cherry"]

print("------ Nested For Loop Example ------\n")

# Outer loop (adjectives)
for x in adj:
    print(f"Adjective: {x}")                 # show which adjective is being used
    print("----------------------")


    for y in fruits:
        print(f"  → {x} {y}")                 # print combination neatly

    print()                                    # empty line after each adjective group
```

Output:

```
>>> %Run Lab_05.py

------ Nested For Loop Example ------

Adjective: red
----------------------
  → red apple
  → red banana
  → red cherry

Adjective: big
----------------------
  → big apple
  → big banana
  → big cherry

Adjective: tasty
----------------------
  → tasty apple
  → tasty banana
  → tasty cherry

>>>
```

LAB # 05

EXERCISE

A. Point out the errors, if any, in the following Python programs.

1. Code

```
for(;;)
{
  printf("SSUET")
}
```

2. Code

```
for count in range(4 + 3, 12, 3):
    print(count)
```

3.Code

```
for v in range(4:8)
print(v)
```

B. What will be the output of the following programs:

1. Code

```
for i in range(1, 10):  # numbers from 1 to 9
    if i % 2 == 0:      # check if even
      print(i)
```

Output

LAB # 05

2. Code

```
for i in range(1, 6):
    print(i)
```

Output

3. Code

```
num = 3
for i in range(1, 6):
    print(f"{num} x {i} = {num*i}")
```

Output

C. Write Python programs for the following:

1. Write a program that prints the first 10 natural numbers and their sum using ‘for loop’.
   Sample output:
   The first 10 natural number are :
   1 2 3 4 5 6 7 8 9 10
   The Sum is : 55

2. Write a program to print the multiplication table of the number entered by the user. The table should get displayed in the following form.

```
    29 x 1 =  29
    29 x 2 =  58
    …
    29 x 10 = 290
```

3. Write a program that take vowels character in a variable named “vowels” then print each vowels characters in newline using ‘for loop’.