|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| C:\Users\Admin\Desktop\download.jpg | USMAN INSTITUTE OF TECHNOLOGY | | | | | |  |
|  |  |  |  |  |  |  |  |
|  | Department of Computer Science  CS121 Object Oriented Programming | | | | | |  |
|  |  | Lab # 04  Loop Statements in Python | | | |  |  |
|  | Objective:  This experiment introduces the students to the concept of Iterations and Loop Statements in Python programming language. Different loop statements of Python including **while,** and **for** statements are covered. Concept of nested loops is also introduced | | | | | |  |
|  | **Name of Student:**  **Roll No: Sec.**  **Date of Experiment:** | | | | | | 7 |
|  | **Marks Obtained/Remarks:**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Signature:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | | | | |  |

**Lab 04: Loop Statements in Python**

# Loops

In general, statements are executed sequentially − The first statement in a function is executed first, followed by the second, and so on. There may be a situation when you need to execute a block of code several number of times.

Programming languages provide various control structures that allow more complicated execution

paths.

A loop statement allows us to execute a statement or group of statements multiple times. The following diagram illustrates a loop statement.

# While Loop Statements

A while loop statement in Python programming language repeatedly executes a target statement as long as a given condition is true.

count = 0

while (count < 9):

print ('The count is:', count)

count = count + 1

print ("Good bye!")

# For Loop Statements

The for statement in Python has the ability to iterate over the items of any sequence, such as a list or a string.

>>> for var in list(range(5)):

print (var)

for letter in 'Python': # traversal of a string sequence

print ('Current Letter :', letter)

print()

fruits = ['banana', 'apple', 'mango']

for fruit in fruits: # traversal of List sequence

print ('Current fruit :', fruit)

print ("Good bye!")

Current Letter : P

Current Letter : y

Current Letter : t

Current Letter : h

Current Letter : o

Current Letter : n

Current fruit : banana

Current fruit : apple

Current fruit : mango

Good bye!

# Student Exercise

1. Write a program that prompts the user to enter three integers and displays them in increasing order.

**Code:**

a = int(input('Enter first number: '))

b = int(input('Enter second number: '))

c = int(input('Enter third number: '))

if a>b and a>c:

if b>c:

print(c,b,a)

else:

print(b,c,a)

elif b>a and b>c:

if a>c:

print(c,a,b)

else:

print(a,c,b)

else:

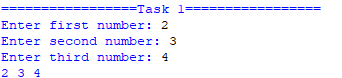
if a>b:

print(b,a,c)

else:

print(a,b,c)

**Output:**



1. Write a program that reads an unspecified number of integers, determines how many positive and negative values have been read, and computes the total and average of the input values (not counting the zero). The program ends with the input 0. Display the average as a floating-point number

**Code:**

i = int( input ("Enter an interger, the input ends if it is 0: "))

count\_pos = 0

count\_neg = 0

total = 0

if (i != 0):

while (i != 0):

if (i > 0):

count\_pos += 1

elif (i < 0):

count\_neg += 1

total += i

i = int( input ("Enter an interger, the input ends if it is 0: "))

count = count\_pos + count\_neg

average = total / count

print ("The number of positives is", count\_pos)

print ("The number of negatives is", count\_neg)

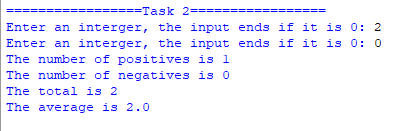
print ("The total is", total)

print ("The average is", float(average))

else:

print ("You didn't enter any number.")

**Output:**



1. Write a program that displays the following table. (Note: 1 kilogram is 2.2 pounds)

|  |  |
| --- | --- |
| Kilograms | Pounds |
| 1 | 2.2 |
| 3 | 6.6 |
| … | … |
| 197 | 433.4 |
| 199 | 437.8 |

**Code:**

print('Kilograms\tPounds')

for i in range(1,200,2):

print(f'{i}\t\t{i\*2.2}')

**Output:**

=================Task 3=================

Kilograms Pounds

1 2.2

3 6.6000000000000005

5 11.0

7 15.400000000000002

9 19.8

11 24.200000000000003

13 28.6

15 33.0

17 37.400000000000006

19 41.800000000000004

21 46.2

23 50.6

25 55.00000000000001

27 59.400000000000006

29 63.800000000000004

31 68.2

33 72.60000000000001

35 77.0

37 81.4

39 85.80000000000001

41 90.2

43 94.60000000000001

45 99.00000000000001

47 103.4

49 107.80000000000001

51 112.2

53 116.60000000000001

55 121.00000000000001

57 125.4

59 129.8

61 134.20000000000002

63 138.60000000000002

65 143.0

67 147.4

69 151.8

71 156.20000000000002

73 160.60000000000002

75 165.0

77 169.4

79 173.8

81 178.20000000000002

83 182.60000000000002

85 187.00000000000003

87 191.4

89 195.8

91 200.20000000000002

93 204.60000000000002

95 209.00000000000003

97 213.4

99 217.8

101 222.20000000000002

103 226.60000000000002

105 231.00000000000003

107 235.4

109 239.8

111 244.20000000000002

113 248.60000000000002

115 253.00000000000003

117 257.40000000000003

119 261.8

121 266.20000000000005

123 270.6

125 275.0

127 279.40000000000003

129 283.8

131 288.20000000000005

133 292.6

135 297.0

137 301.40000000000003

139 305.8

141 310.20000000000005

143 314.6

145 319.0

147 323.40000000000003

149 327.8

151 332.20000000000005

153 336.6

155 341.0

157 345.40000000000003

159 349.8

161 354.20000000000005

163 358.6

165 363.00000000000006

167 367.40000000000003

169 371.8

171 376.20000000000005

173 380.6

175 385.00000000000006

177 389.40000000000003

179 393.8

181 398.20000000000005

183 402.6

185 407.00000000000006

187 411.40000000000003

189 415.8

191 420.20000000000005

193 424.6

195 429.00000000000006

197 433.40000000000003

199 437.8

1. Revise the scissor, rock, paper game to let the user play continuously until either the user or the computer wins more than two times

**Code:**

import random

user\_won=0

comp\_won=0

tie = 0

while True:

bot = random.randint(1,3)

if bot == 1:

bot = "rock"

elif bot == 2:

bot = "paper"

elif bot == 3:

bot = "scissors"

user=input("Enter rock paper or scissors: ")

print("Computer Chose: Rock paper or Scissors ",bot)

if user == bot:

result = "Tie"

tie += 1

else:

if user == "rock" and bot == "scissors":

result = "Bot Won, User Lose"

user\_won += 1

elif user == "paper" and bot == "rock":

result = "Bot Won, User Lose"

user\_won += 1

elif user == "scissors" and bot == "paper":

result = "Bot Won, User Lose"

user\_won += 1

elif bot == "rock" and user == "scissors":

result = "Bot Won, User Lose"

comp\_won += 1

elif bot == "paper" and user == "rock":

result = "Bot Won, User Lose"

comp\_won += 1

elif bot == "scissors" and user == "paper":

result = "Bot Won, User Lose"

comp\_won += 1

print(result)

print("User Won:",user\_won)

print("Bot Won:",comp\_won)

print("Tie:", tie)

if user\_won > 2:

print("!!!USER WON!!!")

break

elif comp\_won > 2:

print("!!!BOT WON!!!")

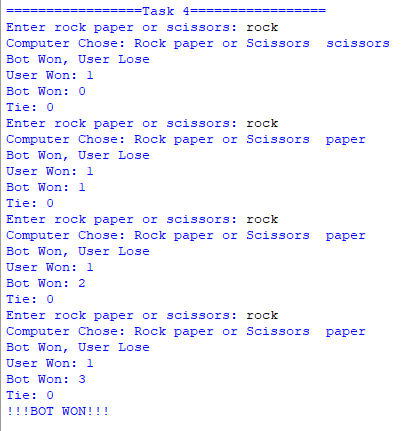
break

elif tie > 2:

print ("!!!TIE!!!")

break

**Output:**

****

1. Write a program that reads integers, finds the largest of them, and counts its occurrences. Assume that the input ends with number 0. Suppose you entered 3 5 2 5 5 5 0; the program finds that the largest number is 5 and the occurrence count for 5 is 4.

**Code:**

num = int(input('Enter a number: '))

max = num

count = 0

while num!=0:

if num>max:

max = num

count = 1

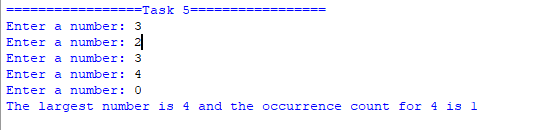
elif num==max:

count+=1

num = int(input('Enter a number: '))

print(f'The largest number is {max} and the occurrence count for {max} is {count}')

**Output:**



1. Use nested loops that display the following pattern

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 |  |  |  |  |  |
| 1 | 2 |  |  |  |  |
| 1 | 2 | 3 |  |  |  |
| 1 | 2 | 3 | 4 |  |  |
| 1 | 2 | 3 | 4 | 5 |  |
| 1 | 2 | 3 | 4 | 5 | 6 |

**Code:**

for i in range(1,7):

for j in range(1,i+1):

print(j,end='\t')

print()

**Output:**

