**NAME :-** TAUSIF KHAN

**ROLL. NO :-** 12

**CLASS :-** SE COMPS B

**EXPERIMENT. NO : 04**

**AIM :-** Write a program to demonstrate use of control statements in python:

1. Write a program in python to implement FizzBuzz
2. A python program to display even numbers between m and n (m and n are taken from user)
3. Write a python program to swap two numbers and find whether the first number is positive or negative or equal to zero.
4. A python program to search for an element in the list of elements (The List must be entered through keyboard.

**Tools Used :-** Python 3.4.3, Terminal

**THEORY :-** In Python we have two types of Control Statement

1. Decision making
2. Loops
3. **DECISION MAKING :-**

Decision making is anticipation of conditions occurring while execution of the program and specifying actions taken according to the conditions.

Decision structures evaluate multiple expressions which produce TRUE or FALSE as outcome. You need to determine which action to take and which statements to execute if outcome is TRUE or FALSE otherwise.

Following is the general form of a typical decision making structure found in most of the programming languages –

Python programming language assumes any non-zero and non-null values as TRUE, and if it is either zero or null, then it is assumed as FALSE value.

Python programming language provides following types of decision making statements. Click the following links to check their detail.



There three types of decision making statements

1. **If Statements :-** An if statement consists of a boolean expression followed by one or more statements.
2. **If – else if Statements (if – elif) :-** An if statement can be followed by an optional else statement, which executes when the boolean expression is FALSE.
3. **Nested If else :-** You can use one if or else if statement inside another if or else if statement(s).
4. **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 for 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 –



Python programming language provides following types of loops to handle looping requirements.

Python has three types of loop

1. **While Loop :-** Repeats a statement or group of statements while a given condition is TRUE. It tests the condition before executing the loop body.
2. **For Loop :-** Executes a sequence of statements multiple times and abbreviates the code that manages the loop variable.
3. **Nested Loop :-** You can use one or more loop inside any another while, for or do..while loop.

**CODES & OUTPUTS :-**

**Write a program in python to implement FizzBuzz.**

CODES :-

x = int(input("Enter the starting value "))

y = int(input("Enter the ending value "))

for i in range(x,y):

if(i%3==0 and i%5==0):

print("Fizz Bizz")

elif(i%3==0):

print("Fizz")

elif(i%5==0):

print("Bizz")

else:

print(i)

'''OUTPUT:-

student@lab:~/Desktop/shreyash$ python3 numfor.py

Enter the starting value 1

Enter the ending value 50

1

2

Fizz

4

Bizz

Fizz

7

8

Fizz

Bizz

11

Fizz

13

14

Fizz Bizz

16

17

Fizz

19

Bizz

Fizz

22

23

Fizz

Bizz

26

Fizz

28

29

Fizz Bizz

31

32

Fizz

34

Bizz

Fizz

37

38

Fizz

Bizz

41

Fizz

43

44

Fizz Bizz

46

47

Fizz

49

student@lab:~/Desktop/shreyash$ '''

**A python program to display even numbers between m and n (m and n are taken from user)**

CODES :-

x = int(input("Enter the starting value "))

y = int(input("Enter the ending value "))

while(x<=y):

if(x%2==0):

print(x)

x=x+1

'''OUTPUT:-

student@lab:~/Desktop/shreyash$ python3 num-x-y.py

Enter the starting value 200

Enter the ending value 250

200

202

204

206

208

210

212

214

216

218

220

222

224

226

228

230

232

234

236

238

240

242

244

246

248

250

student@lab:~/Desktop/shreyash$ '''

**Write a python program to swap two numbers and find whether the first number is positive or negative or equal to zero.**

CODES :-

x = int(input("Enter the first number "))

y = int(input("Enter the second number "))

x,y = y,x

if(x == 0):

print("Number is neither positive nor negative")

elif(x > 0):

print("Number is a positive number")

else:

print("Number is a negative number")

'''OUTPUT :-

student@lab:~/Desktop/shreyash$ python3 swap.py

Enter the first number 3

Enter the second number -9

Number is a negative number

student@lab:~/Desktop/shreyash$ python3 swap.py

Enter the first number 4

Enter the second number 7

Number is a positive number

student@lab:~/Desktop/shreyash$'''

**A python program to search for an element in the list of elements (The List must be entered through keyboard.**

CODES :-

x = [int(x)for x in input(" Enter the elements of list ").split()]

y = int(input("Enter any one number "))

flag = 0

for i in x:

if(i == y):

flag =1;

if(flag ==1):

print("Entered number is present in the list ")

else:

print("Entered number is not present in the list ")

'''OUTPUT:-

student@lab:~/Desktop/shreyash$ python3 list.py

Enter the elements of list 45 78 98 34 23

Enter any one number 34

Entered number is present in the list

student@lab:~/Desktop/shreyash$ python3 list.py

Enter the elements of list 65 98 67 12 56 34 90

Enter any one number 21

Entered number is not present in the list

student@lab:~/Desktop/shreyash$ '''

**CONCLUSION :-**

Thus we have studied various control statements (i.e if ,if – elif, while, for) and implemented it in programs