**Practical-2**

**Aim:** Program to demonstrate the use of if, if-else, while, for, break and continue statements.

🡪Decision making is required when we want to execute a code only if a certain condition is satisfied.

The if, elif, else statement is used in Python for decision making.

1. **The if, if-else, elif, statements:**

The if, elif, else statement is used in Python for decision making.

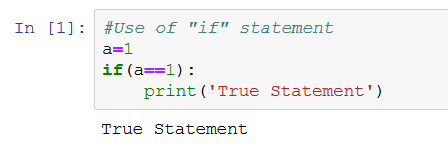
1. **Use of "if" statement:** the program evaluates the test expression and will execute statement(s) only if the text expression is True. If the text expression is False, the statement(s) is not executed.

**Example code:**

a=1

if(a==1):

print('True Statement')



1. **Use of "if-else" statement:** The if-else statement evaluates test expression and will execute body of **if** only when test condition is True. If the condition is False, body of **else** is executed.

**Example code:**

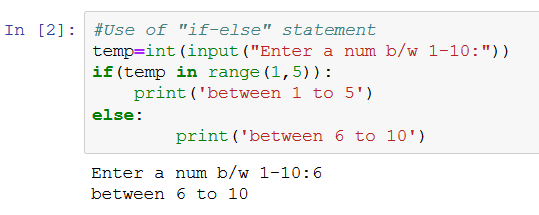
temp=int(input("Enter a num b/w 1-10:"))

if(temp in range(1,5)):

print('between 1 to 5')

else:

print('between 6 to 10')



1. **Use of “if...elif...else” statement:** The elif is short for else if. It allows us to check for multiple expressions. If the condition for if is False, it checks the condition of the next elif block and so on. If all the conditions are False, body of else is executed.

**Example code:**

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

if num>0:

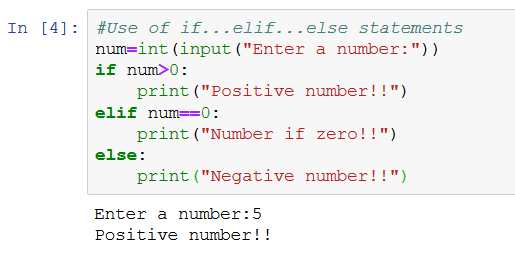
print("Positive number!!")

elif num==0:

print("Number if zero!!")

else:

print("Negative number!!")



1. **The “while” and “for” statements:**
2. **Use of “while” loop:** The while loop in Python is used to iterate over a block of code as long as the test expression (condition) is true. We generally use this loop when we don't know beforehand, the number of times to iterate.

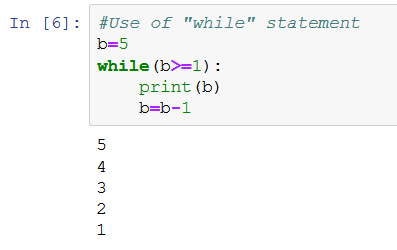
**Example code:**

b=5

while(b>=1):

print(b)

b=b-1



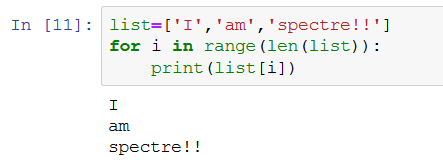
1. **Use of “for” loop:** The for loop in Python is used to iterate over a sequence (list, tuple, string) or other iterable objects. Iterating over a sequence is called traversal.

**Example code:**

list=['I','am','spectre!!']

for i in range(len(list)):

print(list[i])



1. **The “break” and “continue” statements:**  In Python, break and continue statements can alter the flow of a normal loop. Loops iterate over a block of code until test expression is false, but sometimes we wish to terminate the current iteration or even the whole loop without checking test expression. The break and continue statements are used in these cases.
2. **Use of “break” statement:** The break statement terminates the loop containing it. Control of the program flows to the statement immediately after the body of the loop.

**Example code:**

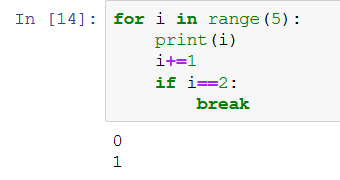
for i in range(5):

print(i)

i+=1

if i==2:

break



1. **Use of “continue” statement:** The continue statement is used to skip the rest of the code inside a loop for the current iteration only. Loop does not terminate but continues on with the next iteration.

**Example code:**

c=10

while(c>=1):

print(c)

c=c-1

if(c>4):

continue

else:

break

