

selection statements

[Document subtitle]



November 14, 2023

**AACHAL MOHITE**

In Python, the selection statements are also known as decision making statements or branching statements. The selection statements are used to select a part of the program to be executed based on a condition. Python provides the following selection statements.

* if statement
* if-else statement
* nested-if statement
* elif statement

**IF STATEMENT:**

The if statement is the most simple decision-making statement. It is used to decide whether a certain statement or block of statements will be executed or not.

Here, the condition after evaluation will be either true or false. if the statement accepts boolean values – if the value is true then it will execute the block of statements below it otherwise not.

As we know, python uses indentation to identify a block. So the block under an if statement will be identified as shown

SYNTAX-

If condition:

Statement(s)

If condition:

Statement(s)

EXAMPLE-

#Pgm to check num is positive

num=int(input(‘enter the number:’)

if(num>=0)

print(‘Num is positive’)

OUTPUT-

Enter the number:2

Num is positive

**IF-ELSE STATEMENT:**

We use the if-else statement to test a condition and pick the execution of a block of statements out of two blocks based on that condition result. The if-else statement checks the given condition then decides which block of statements to be executed based on the condition result. If the condition is True, then the true block of statements is executed and if it is False, then the false block of statements is executed.

SYNTAX-

If condition:

Statement(s)

else:

statement(s)

EXAMPLE-

#Pgm to check the num is positive or not

num=int(input(‘enter the number:’)

if(num>=0)

print(‘Num is positive’)

else:

print(‘Num is not positive’)

OUTPUT-

Enter the number:5

Num is not positive

**NESTED-IF STATEMENT:**

A nested if is an if statement that is the target of another if statement. Nested if statements mean an if statement inside another if statement. Yes, Python allows us to nest if statements within if statements. i.e., we can place an if statement inside another if statement.

All the If conditions will be executed one by one.

SYNTAX-

If (condition1):

If (condition2):

Statement1(s)

else:

Statement2(s)

else:

Statement3(s)

EXAMPLE-

#Pgm to check num is positive or negative,if positive check num is even or odd

num=int(input(‘enter the number:’)

if(num>=0)

if(num%2==0)

print(‘Num is positive and even’)

else:

print(‘Num is positive and odd’)

else:

print(‘Num is negative’)

OUTPUT-

Enter the number:8

Num is positive and even

**ELIF CONDITION:**

Here, a user can decide among multiple options. The if statements are executed from the top down. As soon as one of the conditions controlling the if is true, the statement associated with that if is executed, and the rest of the ladder is bypassed. If none of the conditions is true, then the final else statement will be executed. Number of elif is dependent on the number of conditions to be checked.

SYNTAX-

If condition:

Statement(s)

elif condition:

Statement(s)

elif condition:

Statement(s)

else:

Statement(s)

EXAMPLE-

#Pgm to check num is positive ,negative or zero

num=int(input(‘enter the number:’)

if(num>=0):

print(‘Num is positive’)

elif(num<0)

print(‘Num is negative’)

else:

print(‘Num is zero’)

OUTPUT-

Enter the number:-5

Num is negative