

Scala Conditional Expressions

Scala provides if statement to test the conditional expressions. It tests boolean conditional expression which can be either true or false. Scala use various types of if else statements.

If statement

If-else statement

Nested if-else statement

If-else-if ladder statement

Scala if statement

The scala if statement is used to test condition in scala. If block executes only when condition is true otherwise execution of if block is skipped.

Syntax

```
if(condition){  
    // Statements to be executed  
}
```

Scala If-Else Statement

The scala if-else statement tests the condition. If the condition is true, if block executes otherwise else block executes.

Syntax

```
if(condition){  
    // If block statements to be executed  
} else {  
    // Else block statements to be executed
```

```
}
```

Scala If-Else-If Ladder Statement

The scala if-else-if ladder executes one condition among the multiple conditional statements.

Syntax

```
if (condition1){  
    //Code to be executed if condition1 is true  
}  
else if (condition2){  
    //Code to be executed if condition2 is true  
}  
else if (condition3){  
    //Code to be executed if condition3 is true  
}  
...  
else {  
    //Code to be executed if all the conditions are false  
}
```

Scala If Statement as better alternative of Ternary Operators

In scala, you can assign if statement result to a function. Scala does not have ternary operator concept like C/C++ but provides more powerful if which can return value. Let's see an example

Example

```
object MainObject {
```

```
def main(args: Array[String]) {  
    val result = checkIt(-10)  
    println (result)  
}  
  
def checkIt (a:Int) = if (a >= 0) 1 else -1 // Passing a if expression value to function  
}
```

Output:

-1