

# Scala Programming

( 20/09/2023)

Name: Niranjana J

USN: 22BTRAD027

Branch: CSE- AI& DE

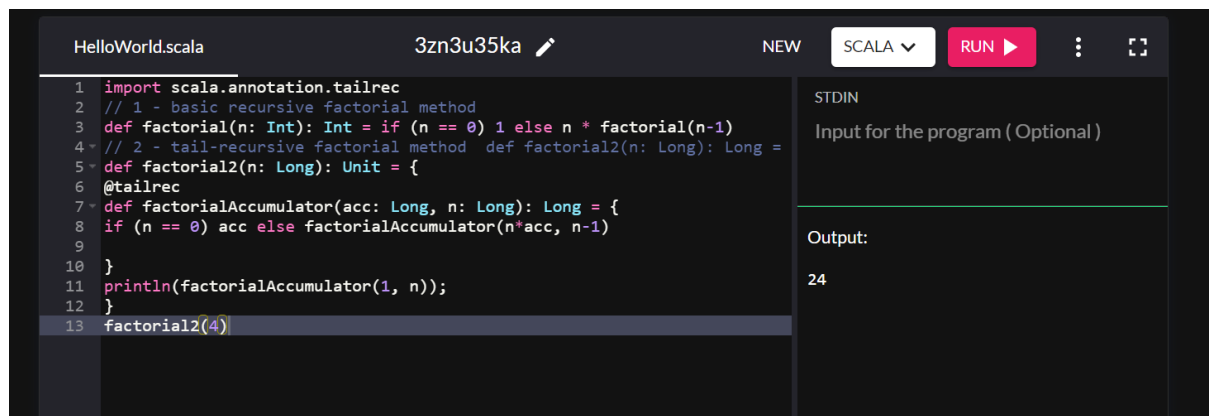
## Question:

Factorial using recursion

## Code:

```
import scala.annotation.tailrec
import scala.io.StdIn
// 1 - basic recursive factorial method
def factorial(n: Int): Int = if (n == 0) 1 else n * factorial(n-1)
// 2 - tail-recursive factorial method def factorial2(n: Long): Long = {
def factorial2(n: Long): Unit = {
  @tailrec
  def factorialAccumulator(acc: Long, n: Long): Long = {
    if (n == 0) acc else factorialAccumulator(n*acc, n-1)
  }
  println(factorialAccumulator(1, n));
}
val a = scala.io.StdIn.readInt()
factorial2(a)
```

## Output:

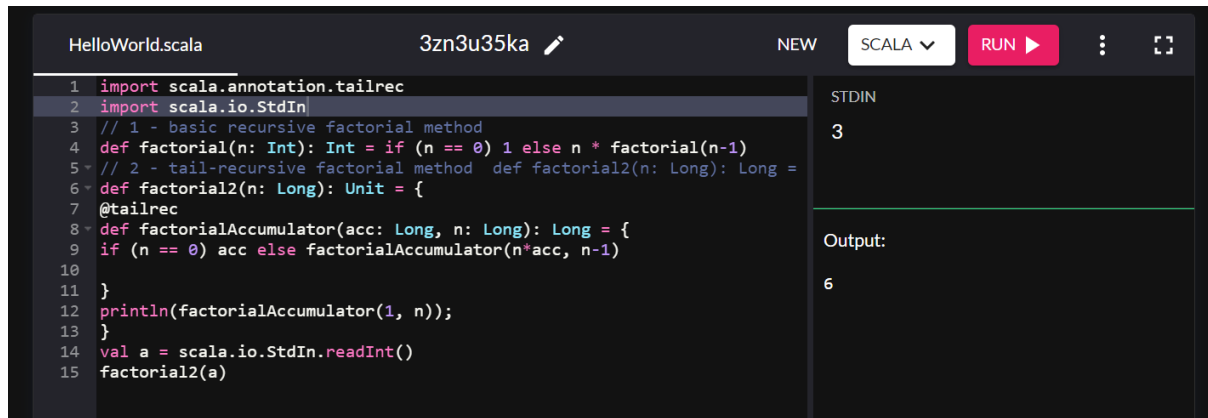
A screenshot of a Scala IDE interface. The top bar shows the file name 'HelloWorld.scala', a user identifier '3zn3u35ka', and buttons for 'NEW', 'SCALA', and 'RUN'. The main editor area contains the Scala code for calculating factorials using both a basic recursive method and a tail-recursive method. The code is as follows:

```
1 import scala.annotation.tailrec
2 // 1 - basic recursive factorial method
3 def factorial(n: Int): Int = if (n == 0) 1 else n * factorial(n-1)
4 // 2 - tail-recursive factorial method def factorial2(n: Long): Long = {
5 def factorial2(n: Long): Unit = {
6   @tailrec
7   def factorialAccumulator(acc: Long, n: Long): Long = {
8     if (n == 0) acc else factorialAccumulator(n*acc, n-1)
9   }
10  }
11  println(factorialAccumulator(1, n));
12 }
13 factorial2(4)
```

The right-hand side of the IDE shows the 'STDIN' input field with the text 'Input for the program ( Optional )' and an 'Output' field displaying the result '24'.

In this program, the value is passed in the code itself.

User input: In this program, a user input is accepted from the user. For this, the `scala.io.StdIn` library is imported. Using `readInt()` function within the library, the user passes a value to the function.



```
1 import scala.annotation.tailrec
2 import scala.io.StdIn
3 // 1 - basic recursive factorial method
4 def factorial(n: Int): Int = if (n == 0) 1 else n * factorial(n-1)
5 // 2 - tail-recursive factorial method def factorial2(n: Long): Long =
6 def factorial2(n: Long): Unit = {
7   @tailrec
8   def factorialAccumulator(acc: Long, n: Long): Long = {
9     if (n == 0) acc else factorialAccumulator(n*acc, n-1)
10  }
11 }
12 println(factorialAccumulator(1, n));
13 }
14 val a = scala.io.StdIn.readInt()
15 factorial2(a)
```

STDIN

3

Output:

6

**Github link:**

<https://github.com/niranjana628/Scala-Programming>