

SCALA PROGRAMMING

20/9/2023

Name: Arjun Unnikrishnan USN: 22BTRAD004

Tail Recursion

Code

```
HelloWorld.scala 3zn3u4tba ✎  
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   @tailrec  
6   def factorialAccumulator(acc: Long, n: Long): Long = {  
7     if (n == 0) acc else factorialAccumulator(n*acc, n-1)  
8   }  
9   println(factorialAccumulator(1,3));
```

Output

Output:

6

Code


```
HelloWorld.scala 3zn3u4tba ✎  
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  
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  println(factorialAccumulator(1,n))  
11 }  
12 factorial2(5)  
13
```

Output

Output:

120

Code

```
HelloWorld.scala 3zn3u4tba 
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
6 // 2 - tail-recursive factorial method
7 def factorial2(n: Long): Unit = {
8   @tailrec
9   def factorialAccumulator(acc: Long, n: Long): Long = {
10     if (n == 0) acc else factorialAccumulator(n * acc, n - 1)
11   }
12
13   println(factorialAccumulator(1,n))
14 }
15
16 println("Enter number:")
17 val n = StdIn.readLong()
18 println("Factorial of "+n+":")
19 factorial2(n)
```

Output

STDIN

12

Output:

Enter number:

Factorial of 12:

479001600