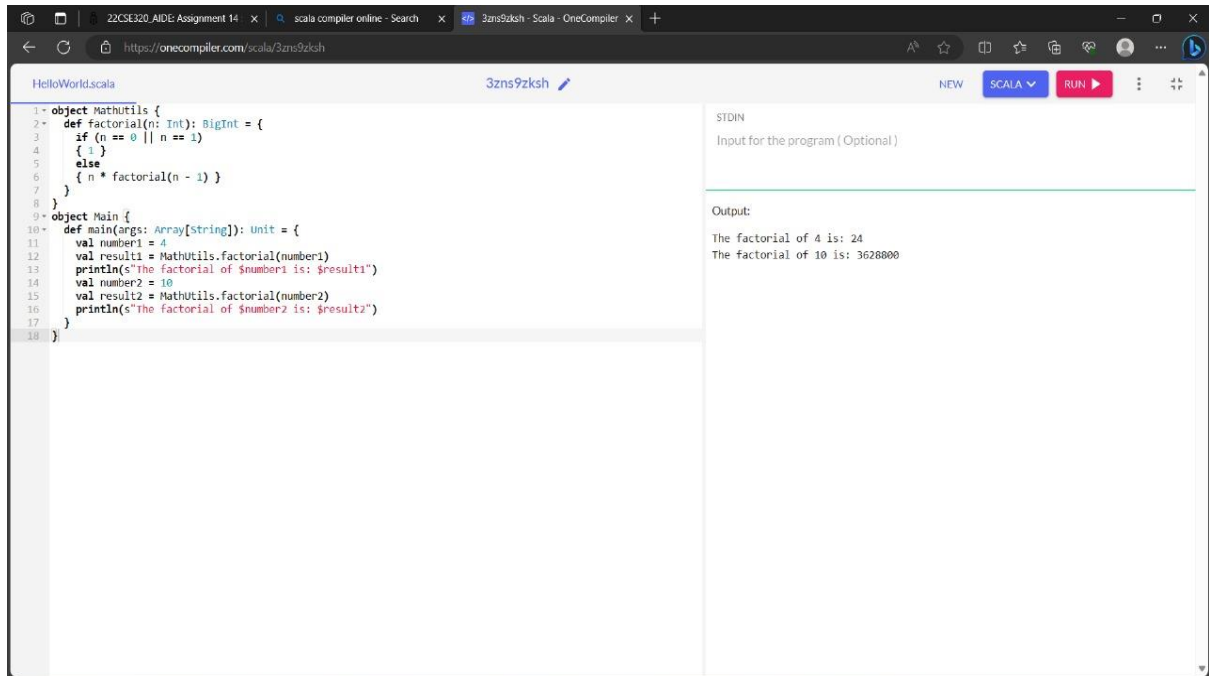


LAB-12

ABHINANDAN SINGH PARMAR

22BTRAD001

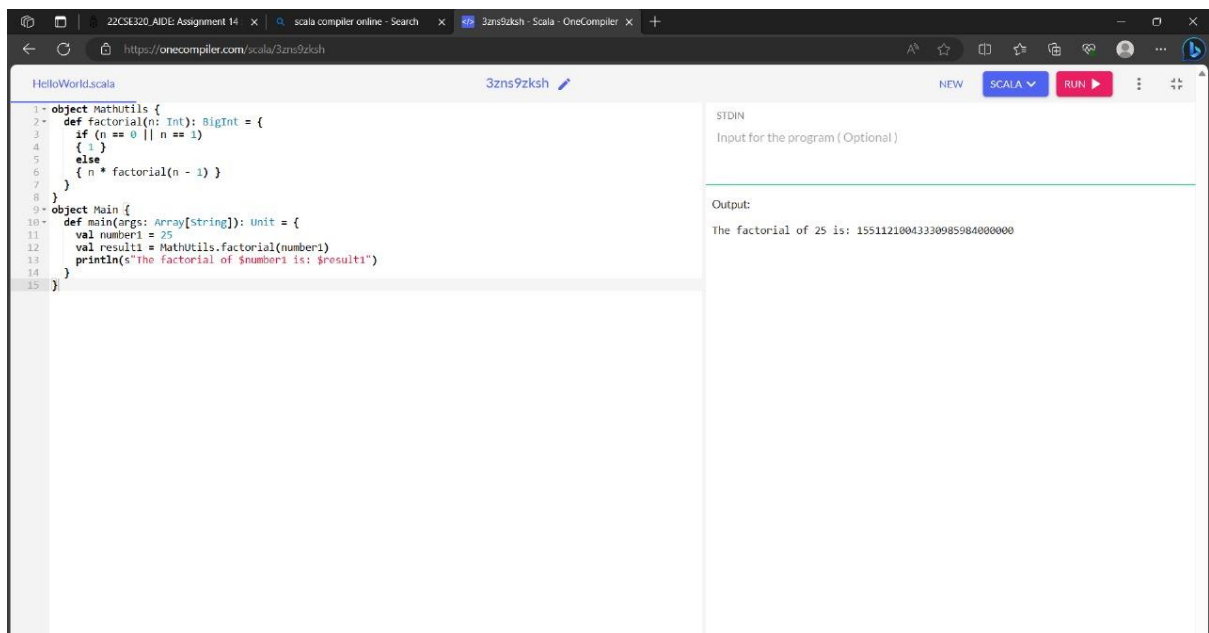


The screenshot shows the Scala OneCompiler interface. The code editor on the left contains the following Scala code:

```
1 object MathUtils {  
2   def factorial(n: Int): BigInt = {  
3     if (n == 0 || n == 1)  
4       { 1 }  
5     else  
6       { n * factorial(n - 1) }  
7   }  
8 }  
9  
10 object Main {  
11   def main(args: Array[String]): Unit = {  
12     val number1 = 4  
13     val result1 = MathUtils.factorial(number1)  
14     println(s"The factorial of $number1 is: $result1")  
15     val number2 = 10  
16     val result2 = MathUtils.factorial(number2)  
17     println(s"The factorial of $number2 is: $result2")  
18   }  
19 }
```

The right panel shows the output of the program:

Output:
The factorial of 4 is: 24
The factorial of 10 is: 3628800



The screenshot shows the Scala OneCompiler interface with the same code as the first image, but the output is for a larger number:

```
1 object MathUtils {  
2   def factorial(n: Int): BigInt = {  
3     if (n == 0 || n == 1)  
4       { 1 }  
5     else  
6       { n * factorial(n - 1) }  
7   }  
8 }  
9  
10 object Main {  
11   def main(args: Array[String]): Unit = {  
12     val number1 = 25  
13     val result1 = MathUtils.factorial(number1)  
14     println(s"The factorial of $number1 is: $result1")  
15   }  
16 }
```

Output:
The factorial of 25 is: 15511210043330985984000000

Code:

```
object MathUtils {  
  def factorial(n: Int): BigInt = {  
    if (n == 0 || n == 1)  
      { 1 }  
    else  
      { n * factorial(n - 1) }  
  }  
}  
  
object Main {  
  def main(args: Array[String]): Unit = {  
    val number1 = 4  
    val result1 = MathUtils.factorial(number1)  
    println(s"The factorial of $number1 is: $result1")  
    val number2 = 10  
    val result2 = MathUtils.factorial(number2)  
    println(s"The factorial of $number2 is: $result2")  
  }  
}
```

Output:

```
The factorial of 4 is: 24  
The factorial of 10 is: 3628800
```