Sagnik Roy
500109927
Big Data-Batch 2

Advanced Functional Thinking Lab
Experiment-3

Q1. Scala program to implement the foreach loop on a list of numbers.

Code-

```scala
//SagnikRoy_500109927
object ForEach {
  def main(args: Array[String]): Unit = {
    val numbers = List(1, 2, 3, 4, 5)
    numbers.foreach(num => println(num))
  }
}
```

Output-

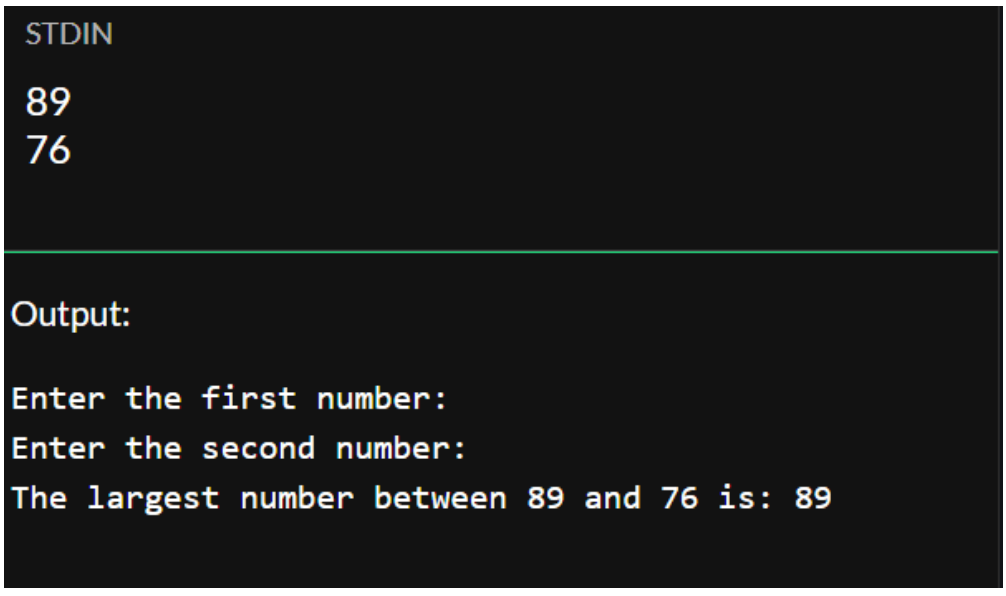
```
Output:

1
2
3
4
5
```

Q2. Scala program to create a user define function to return largest number among two numbers entered by user.

Code-

```scala
//SagnikRoy_500109927
object LargestNumber{
  def main(args: Array[String]): Unit = {
    println("Enter the first number:")
    val num1 =scala.io.StdIn.readInt()
    println("Enter the second number:")
    val num2 =scala.io.StdIn.readInt()
    val largest = findLargest(num1, num2)
    println(s"The largest number between $num1 and $num2 is: $largest")
  }
  def findLargest(num1: Int, num2: Int): Int = {
    if (num1 > num2) {
      num1
    }else {
      num2
    }
  }
}
```

Output-



Q3. Scala code to create a function with default arguments. Wish good morning to the person.

Code-

```
//SagnikRoy_500109927
object Greeting {
  def main(args: Array[String]):Unit = {
    greetPerson()
    greetPerson()
    greetPerson()
    greetPerson("Roy")
  }
def greetPerson(name:String="Friend"):Unit={
  println("Good morning, "+name+"!")
  }
}
```

Output-

```
Output:

Good morning, Friend!
Good morning, Friend!
Good morning, Friend!
Good morning, Roy!
```

Q4. Scala code to create anonymous functions for add, sub, and mul with => operator.

Code-

```
//SagnikRoy_500109927
object Anonymous{
  def main(args: Array[String]): Unit = {
  println("Enter the first number:")
  val p= scala.io.StdIn.readInt()

  println("Enter the second number:")
  val q= scala.io.StdIn.readInt()

  val add =(x: Int, y: Int) => x + y

  val sub= (x: Int, y: Int) => x - y

  val mul =(x: Int, y: Int) => x * y

  println("Addition result:" +add(p,q));
  println("Subtraction result:" +sub(p,q));
  println("Multiplication result:" +mul(p,q));
  }
}
```

Output-

```
STDIN
15
10

Output:

Enter the first number:
Enter the second number:
Addition result:25
Subtraction result:5
Multiplication result:150
```

Q5. Scala code to create anonymous functions for add, sub, and mul with _ operator.

Code-

```
//SagnikRoy_500109927
object Anonymous{
  def main(args: Array[String]): Unit = {
  println("Enter the first number:")
  val x= scala.io.StdIn.readInt()

  println("Enter the second number:")
  val y= scala.io.StdIn.readInt()

  val add = (_: Int)+(_: Int)

  val sub = (_: Int)-(_: Int)

  val mul = (_: Int)*(_: Int)

  println("Addition result:" +add(x,y));
  println("Subtraction result:" +sub(x,y));
  println("Multiplication result:" +mul(x,y));
  }
}
```

Output-

```
STDIN
12
9

Output:

Enter the first number:
Enter the second number:
Addition result:21
Subtraction result:3
Multiplication result:108
```