LAB 3

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

Ans: **Code:-**

HelloWorld.scala 42

```scala
object ForEach extends App {

  val numbers = List(1, 2, 3, 4, 5,6,7)


  numbers.foreach(num => println(num))
  //vaswati//
}
```

OUTPUT:

NEW SCALA RUN

STDIN

Input for the program (Optional)

Output:

```
1
2
3
4
5
6
7
```

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

Ans: **Code:-**

LAB 3

HelloWorld.scala

```scala
object Func{
    def comp(a:Int, b:Int): Int={
  if(a>b){
     return a;
  }
  else{
     return b;
  }
    }
    def main(args: Array[String]):Unit={
  println("Enter the two nos.");
  var p=scala.io.StdIn.readLine().toInt
  var q=scala.io.StdIn.readLine().toInt
  println("The largest no. is " + comp(p,q));
    }
}
//vaswati//
```

OUTPUT:-

NEW SCALA RUN

STDIN

```
10
23
```

Output:

```
Enter the two nos.
```

NEW SCALA RUN

STDIN

23

Output:

```
Enter the two nos.
```

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

Ans: **Code:-**

HelloWorld.scala 423wat

```scala
object Night {
  def main(args: Array[String]): Unit = {
    greet()
    println("Enter name:")
    val name = scala.io.StdIn.readLine().toString()
    greet(name)
  }

  def greet(name: String="po"): Unit = {
    println("Hi, " + name + "!")
  }
}
//vaswati//
```

OUTPUT:-

NEW SCALA RUN

STDIN

vaswati

Output:

```
Hi, po!
Enter name:
Hi, vaswati!
```

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

Ans: **Code:-**

HelloWorld.scala 423wat7

```scala
object AnonymousFunctions {
  def main(args: Array[String]): Unit = {

    val add = (x: Int, y: Int) => x + y
    val sub = (x: Int, y: Int) => x - y
    val mul = (x: Int, y: Int) => x * y


    val resultAdd = add(10, 5)
    val resultSub = sub(10, 5)
    val resultMul = mul(10, 5)


    println("Addition result: " + resultAdd)
    println("Subtraction result: " + resultSub)
    println("Multiplication result: " + resultMul)
  }
}
//vaswati//
```

OUTPUT:-

NEW SCALA RUN

STDIN

Output:

```
Addition result: 15
Subtraction result: 5
Multiplication result: 50
```

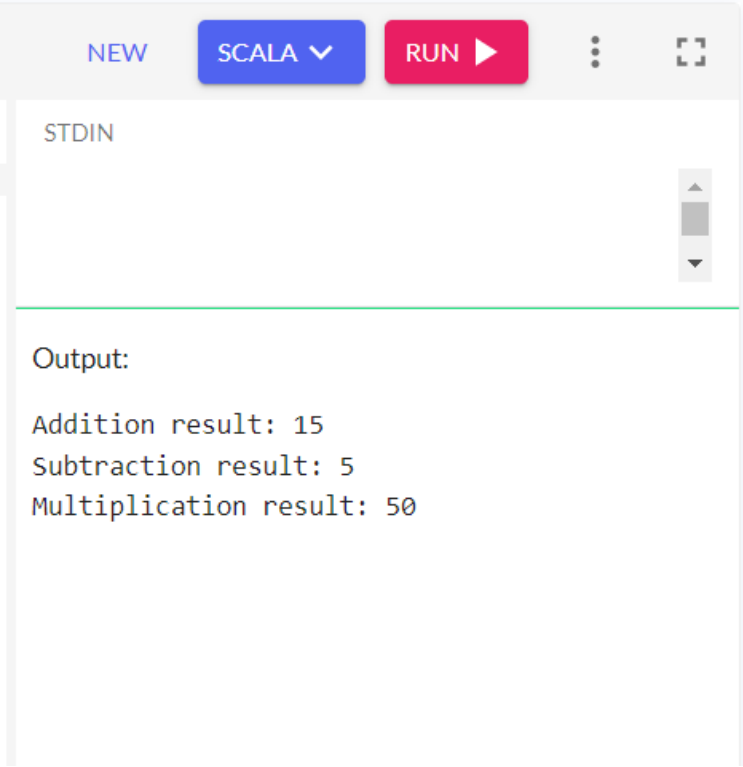
Q5. Scala code to create anonymous functions for add, sub, and mul with _ operator.

Ans: **Code:-**

HelloWorld.scala 423wat7

```scala
object AnonymousFunctions {
  def main(args: Array[String]): Unit = {

    val add = (_: Int) + (_: Int)
    val sub = (_: Int) - (_: Int)
    val mul = (_: Int) * (_: Int)


    val resultAdd = add(10, 5)
    val resultSub = sub(10, 5)
    val resultMul = mul(10, 5)


    println("Addition result: " + resultAdd)
    println("Subtraction result: " + resultSub)
    println("Multiplication result: " + resultMul)
  }
}

//vaswati//
```

OUTPUT:-


NEW
SCALA
RUN
STDIN
Output:
Addition result: 15
Subtraction result: 5
Multiplication result: 50