SCALA TUTORIAL 2

Q1,Q2).

Scala Code

```
object Question_1_2_Scala {
  def main(args: Array[String]): Unit = {
    var k,i,j:Int= 2
    var m,n:Int = 5
    var f:Float = 12.0f
    var g:Float = 4.0f
    var c:Char = 'X'

    println("K + 12 * m = "+ (k + 12 * m))
    println("m / j = "+ (m / j))
    println("n % j = "+ (n % j))
    println("m / j * j = "+ (m / j * j))
    println("f + 10 * 5 + g = "+ (f + 10 * 5 + g))
    println("++i * n = "+ (prefixIncrement(i) * n))

  }

  def prefixIncrement(value:Int):Int=
  {
      value + 1
  }

}
```

Java Code

```
public class Qusetion_1_2_Java {
    public static void main(String[] args)
    {
        int k,i,j;
        k = i = j = 2;
        int m,n;
        m = n = 5;
        float f = 12f;
        float g = 4f;
        char c = 'x';


        System.out.println("K + 12 * m = "+(k + 12 *
m));
        System.out.println("m / j = "+ (m / j));
        System.out.println("n % j = "+ (n % j));
        System.out.println("m / j * j = "+(m / j * j));
        System.out.println("f + 10 * 5 + g = "+ (f + 10 * 5 + g));
        System.out.println("++i * n = "+(++i * n));

    }
}
```

Q3).

Scala Code

```
object Question_3 {
  def main(args: Array[String]): Unit = {
    var a:Int=2
    var b:Int = 3
    var c:Int = 4
    var d:Int = 5
    var g = 4.0f
    var k = 4.3f


   println( "--b*a+c*d-- :" + prefixDecrement(b) * a + c *postfixDecrement(d))
   println("a++ :" + postfixIncrement(a))
   println("(-2) * ( g - k ) +c :" + (-2) * (g-k) + c)
   println("c=c++ :" + postfixIncrement(c));
   println("c=++c*a++ :" + prefixIncrement(c)*postfixIncrement(a));



  def prefixIncrement(value:Int):Int = value + 1
  def prefixDecrement(value:Int):Int = value - 1

  def postfixDecrement(value:Int):Int = value
  def postfixIncrement(value:Int):Int = value

  }
}
```

Java Code

```
public class Question_3_java {
    public static void main(String[] args) {

        int a = 2,b = 3,c = 4, d = 5;
        float g = 4.0f,k = 4.3f;

        System.out.println( "--b*a+c*d-- :" + (--b) * a + c * (d--));
        System.out.println("a++ :" + (a++));
        System.out.println("(-2) * ( g - k ) +c :" + (-2) * (g-k) + c);
        System.out.println("c=c++ :" + (c++));
        System.out.println("c=++c*a++ :" + (++c)*(a++));
     }
}
```

Q4) => a

```
object Question_4_a {
  def main(args: Array[String]): Unit = {

    def normalEarn(normalHours:Int):Int = normalHours * 250

    def OTearn(OThour:Int):Int = OThour * 85

    def salaryWithoutTax(normalHours:Int,OThour:Int):Int=
normalEarn(normalHours) + OTearn(OThour)


    def tax(salaryWithoutTax:Int): Double =
    {
      salaryWithoutTax * 0.12
    }

    def netSalary(normalHours:Int,OThours:Int):Double=
    {
      salaryWithoutTax(normalHours,OThours) -
tax(salaryWithoutTax(normalHours,OThours))
    }

    println()
    print("Enter the Employee Working Hours:")
    var normalHours = scala.io.StdIn.readInt()
    print("Enter the Employee OT Hours:")
    var OThours = scala.io.StdIn.readInt()

    println("Net salary of Employee :"+netSalary(normalHours, OThours))

  }
}
```

Q4) => b

```
object Question_4_b {

  def main(args: Array[String]): Unit = {

    def attendance(ticketPrice:Int):Int= ticketPrice
match
    {
        case x if x > 15 && x > 0 =>
          if((120 + (-20/5) * (x - 15)) > 0)
              (120 + (-20/5) * (x - 15))
          else
              0

        case x if x < 15 && x > 0 =>
          if(120 + (20/5) * (15 - x) > 0)
              120 + (20/5) * (15 - x)
          else
              0

        case x if x == 15 => 120
    }


    def ticketRevenue(ticketPrice:Int):Int=
    {
        ticketPrice * attendance(ticketPrice)
    }


    def ownerCost(ticketPrice:Int):Int=
    {
        500 + 3 * attendance(ticketPrice)
    }
```

```
    def netProfit(ticketPrice:Int):Int=
    {
        ticketRevenue(ticketPrice) - ownerCost(ticketPrice)
    }


    var profitCompareMap = Map[Int,Int]()
    var i = 1
    while(i > 0)
      {
        print("Enter the ticket price :")
        var ticketPrice = scala.io.StdIn.readInt()
        var profit = netProfit(ticketPrice)
        profitCompareMap += ticketPrice->profit

        profitCompareMap.keys.foreach{ i=>
          println("Ticket Price Rs."+i+"  =>  "+"Profit
Rs."+profitCompareMap(i))
        }

         println("Best Ticket Price is
:Rs."+profitCompareMap.maxBy(_._2))

         println()
       }
   }
}
```

Output of 4 => b

Enter the ticket price :26

Ticket Price Rs.10 => Profit Rs.480

Ticket Price Rs.20 => Profit Rs.1200

Ticket Price Rs.21 => Profit Rs.1228

Ticket Price Rs.22 => Profit Rs.1248

Ticket Price Rs.15 => Profit Rs.940

Ticket Price Rs.24 => Profit Rs.1264

Ticket Price Rs.25 => Profit Rs.1260

Ticket Price Rs.26 => Profit Rs.1248

Ticket Price Rs.23 => Profit Rs.1260

Best Ticket Price is :Rs.(24,1264)

Enter the ticket price :

*****