Labsheet 5

Name: Aniketh Vijesh
Roll Number: AM.EN.U4AIE22009

Questions

Sr. No		Name of Experiment / Program
1	a	Write a Scala program to compute the sum of the two given integer values. If the two values are the same, then return triples their sum.
	b	Write a Scala program to check two given integers, and return true if one of them is 22 or if their sum is 32.
2	a	Write a Scala program to remove the character in a given position of a given string. The given position will be in the range 0...string length -1 inclusive.
	b	Write a Scala program to create a new string taking the first 5 characters of a given string and return the string with the 5 characters added at both the front and back.
3	a	Write a Scala program to print the multiplication table of a given number using a for loop.
	b	Write a Scala program to find the largest element in an array using pattern matching
4	a	Write a Scala function to calculate the product of digits in a given number
	b	Write a Scala function to check if a given number is a perfect square

Solutions

1.a

```
object HelloWorld {

    def specsum(a:Int, b:Int): Int = {
      if (a==b) 3*(a+b) else a+b
    }

    def main(args: Array[String]): Unit = {
    println(specsum(3,3))
    }
}
```

```
Output:

18
```

1.b

```
object HelloWorld {

    def specsum(a:Int, b:Int): Boolean = {

      if (a == 22 || b == 22 || a+b ==32) true else false
    }

    def main(args: Array[String]): Unit = {
    println(specsum(3,3))
    }
}
```

```
Output:

false
```

2.a

```
object HelloWorld {

def removeCharAt(str: String, pos: Int): String = {
  if (pos < 0 || pos >= str.length) str else str.substring(0, pos) +
str.substring(pos + 1)
}

    def main(args: Array[String]): Unit = {
    println(removeCharAt("Something", 4))
    }
}
```

```
Output:

Somehing
```

2.b

```
object HelloWorld {

def customString(str: String): String = {
  val first_chars = if (str.length >= 5) str.substring(0,5) else str

  val new_string = first_chars+str+first_chars
  println(first_chars)
  new_string

}

    def main(args: Array[String]): Unit = {
    println(customString("Aniketh"))
    }
}
```

```
Output:

Anike
AnikeAnikethAnike
```

3.a

```
object HelloWorld {

    def multiplicationTable(n: Int):Unit={
      for(i <- 1 to 20){
        println(s"$n x $i = ${n*i}")
      }
    }

    def main(args: Array[String]): Unit = {
    multiplicationTable(3)
    }
}
```

```
Output:

3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24
3 x 9 = 27
3 x 10 = 30
3 x 11 = 33
3 x 12 = 36
3 x 13 = 39
3 x 14 = 42
3 x 15 = 45
3 x 16 = 48
3 x 17 = 51
3 x 18 = 54
3 x 19 = 57
3 x 20 = 60
```

3.b

```scala
object HelloWorld {

  def findLargest(arr: Array[Int]): Int = {
    arr.max
  }

  def main(args: Array[String]): Unit = {
    val numbers = Array(2, 3, 4, 24, 5, 5, 52, 587, 70, 66, 788, 8)
    println(findLargest(numbers))
  }
}
```

```
Output:

788
```

4.a

```
object HelloWorld {

def productOfDigits(n: Int): Int = {
  n.toString.map(_.asDigit).product
}

  def main(args: Array[String]): Unit = {
   println(productOfDigits(324))
  }
}
```

```
Output:

24
```

4.b

```
object HelloWorld {

def isPerfectSquare(n: Int): Boolean = {
  val sqrt = Math.sqrt(n).toInt
  sqrt * sqrt == n
}

  def main(args: Array[String]): Unit = {
   println(isPerfectSquare(9))
  }
}
```

```
Output:

true
```

for the input 45

```
Output:

false
```