22001212 – Practical 08

01)

22001212_q1.scala > {} CaesarCipher > cipher

```scala
object CaesarCipher {

  def encryption(msg: String, key: Int):String = {
    var encryptedText = "";
    for (char <- msg) {
      if (char.isLetter) {
        val offset = if (char.isUpper) 'A' else 'a';
        val encryptedChar = ((char - offset + key) % 26 + offset).toChar;
        encryptedText += encryptedChar;
      }
      else {
        encryptedText += char;
      }
    }
    return encryptedText;
  }

  def decryption(msg: String, key: Int):String = {
    encryption(msg, 26 - key);
  }

  def cipher(text: String, shift: Int, func: (String, Int) => String): String = {
    func(text, shift);
  }

  def main(args: Array[String]): Unit = {
    println("Enter your choice : \n1.Encryption \n2.Decryption");
    var choice=scala.io.StdIn.readInt();
    if(choice==1){
        println("Enter the text : ");
        var text=scala.io.StdIn.readLine();
        println("Enter the shifts : ");
        var shift=scala.io.StdIn.readInt();
        println("Encrypted text : "+ cipher(text,shift,encryption));
    }
    else if(choice==2){
        println("Enter the text : ");
        var text=scala.io.StdIn.readLine();
        println("Enter the shifts : ");
        var shift=scala.io.StdIn.readInt();
        println("Decrypted text : "+ cipher(text,shift,decryption));
    }
    else{
        println("Invalid Choice");
    }
  }
}

```

```
PS C:\Users\User\Desktop\UCSC\2nd Year Sem-01\SCS 2204 - Functional Programming\Practical 08> sc
Enter your choice :
1.Encryption
2.Decryption
1
Enter the text :
Hello Sri Lanka
Enter the shifts :
1
Encrypted text : Ifmmp Tsj Mbolb
PS C:\Users\User\Desktop\UCSC\2nd Year Sem-01\SCS 2204 - Functional Programming\Practical 08> sc
Enter your choice :
1.Encryption
2.Decryption
2
Enter the text :
bcdef
Enter the shifts :
1
Decrypted text : abcde
PS C:\Users\User\Desktop\UCSC\2nd Year Sem-01\SCS 2204 - Functional Programming\Practical 08>
```

02)

22001212_q2.scala > {} Categorizer

```scala
object Categorizer{

  def main(args: Array[String]): Unit = {
    if (args.isEmpty) {
      println("Please provide an integer input.\n");
    }

    val input = args(0).toInt;

    val multipleOfThree: Int => Boolean = _ % 3 == 0;
    val multipleOfFive: Int => Boolean = _ % 5 == 0;

    val result = (multipleOfThree(input), multipleOfFive(input)) match {
      case (true, true)   => "Multiple of Both Three and Five";
      case (true, false)  => "Multiple of Three";
      case (false, true)  => "Multiple of Five";
      case (false, false) => "Not a Multiple of Three or Five";
    }

    println(f"$result\n");
  }
}

```

```
PS C:\Users\User\Desktop\UCSC\2nd Year Sem-01\SCS 2204 - Functional Programming\Practical 08> scala 22001212_q2.scala
Please provide an integer input.

java.lang.ArrayIndexOutOfBoundsException: Index 0 out of bounds for length 0
        at Categorizer$.main(22001212_q2.scala:8)
        at Categorizer.main(22001212_q2.scala)
        at java.base/jdk.internal.reflect.DirectMethodHandleAccessor.invoke(DirectMethodHandleAccessor.java:103)
        at java.base/java.lang.reflect.Method.invoke(Method.java:580)
        at dotty.tools.scripting.ScriptingDriver.compileAndRun(ScriptingDriver.scala:33)
        at dotty.tools.scripting.Main$.process(Main.scala:45)
        at dotty.tools.MainGenericRunner$.run$1(MainGenericRunner.scala:250)
        at dotty.tools.MainGenericRunner$.process(MainGenericRunner.scala:270)
        at dotty.tools.MainGenericRunner$.main(MainGenericRunner.scala:281)
        at dotty.tools.MainGenericRunner.main(MainGenericRunner.scala)
        at java.base/jdk.internal.reflect.DirectMethodHandleAccessor.invoke(DirectMethodHandleAccessor.java:103)
        at java.base/java.lang.reflect.Method.invoke(Method.java:580)
        at coursier.bootstrap.launcher.a.a(Unknown Source)
        at coursier.bootstrap.launcher.Launcher.main(Unknown Source)
PS C:\Users\User\Desktop\UCSC\2nd Year Sem-01\SCS 2204 - Functional Programming\Practical 08> scala 22001212_q2.scala 96
Multiple of Three

PS C:\Users\User\Desktop\UCSC\2nd Year Sem-01\SCS 2204 - Functional Programming\Practical 08> scala 22001212_q2.scala 75
Multiple of Both Three and Five

PS C:\Users\User\Desktop\UCSC\2nd Year Sem-01\SCS 2204 - Functional Programming\Practical 08>
```