

## Task 2:

//Create a list of Strings

```
scala> val list1 = List[String]("alpha", "beta", "gamma", "omega", "zeta", "beta")
```

**Output:** list1: List[String] = List(alpha, beta, gamma, omega, zeta, beta)

//Find count of all strings with length 4

//Check the length of each element, filter where length is equal to 4 and then count the number of strings

```
scala> println(list1.count(x=>x.length == 4))
```

**Output:** 3

//Convert the list of string to a list of integers, where each string is mapped to its corresponding length

//Get the length of each string and map it to a new list

```
scala> val list2 = list1.map(x=>x.length)
```

**Output:** list2: List[Int] = List(5, 4, 5, 5, 4, 4)

//Find count of all strings which contain alphabet 'm'

//Check the count of alphabet 'm' in each string, if count is greater than zero, add string to the list and count the no. of items in the list

```
scala> println(list1.count(x=>x.count(_ == 'm')>0))
```

**Output:** 2

//Find the count of all strings which start with the alphabet 'a'

//For each String, check if it starts with "a". If yes, add it to the count

```
scala> println(list1.count(x=>x.startsWith("a")))
```

**Output:** 1

## Task 3:

**//Scala application to calculate gcd for two numbers**

```
import org.apache.spark.SparkConf
```

```
import org.apache.spark.SparkContext
```

```
object GCD {
```

```
  //Define the main method
```

```
def main(args : Array[String]){  
  //Set the Spark Conf and Spark Context  
  val conf = new SparkConf().setAppName("GCD").setMaster("local[4]")  
  val sc = new SparkContext(conf)  
  sc.setLogLevel("ERROR")  
  
  //Cal the function for calculating the gcd and save the result  
  val result = gcd(30,40)  
  
  //Print the gcd for two numbers  
  println("The gcd is : " + result)  
  
  //Stop the Spark Context  
  sc.stop()  
}  
  
  //Define the method for calculating gcd  
def gcd(a: Int,b: Int): Int = {  
  if(b == 0) a else gcd(b, a%b)  
}  
}
```