

## ASSIGNMENT 12.3

### **Problem Statement:**

Create a list of tuples, where the 1st element of the tuple is an int and the second element is string.

Example - ((1, 'alpha'), (2, 'beta'), (3, 'gamma'), (4, 'zeta'), (5, 'omega'))

1. For the above list, print the numbers where the corresponding string length is 4
2. Find the average of all numbers, where the corresponding string contains alphabet 'm' or 'z'

### **Solution:**

1. For the above list, print the numbers where the corresponding string length is 4

Here is the code snippet I have written in Scala to get the required result:

```
val myTuple = List((1,"alpha"), (2,"beta"), (3,"gamma"), (4,"zeta"), (5,"omega"))  
  
val result = myTuple.filter(tuple => tuple._2.length() == 4)  
  
println("Numbers for which the corresponding string length is 4:")  
  
result.foreach(tuple => println(tuple._1))
```

### **Comments:**

Line 1: Initialize a list with few tuples (key, value pairs) of our choice. The resulting value is of type, List [(Int, String)]

Line 2: Apply filter() method to get only those tuples which are of length 4. The resulting value is of same type as previous one.

Line 3: Print a caption for resulting values.

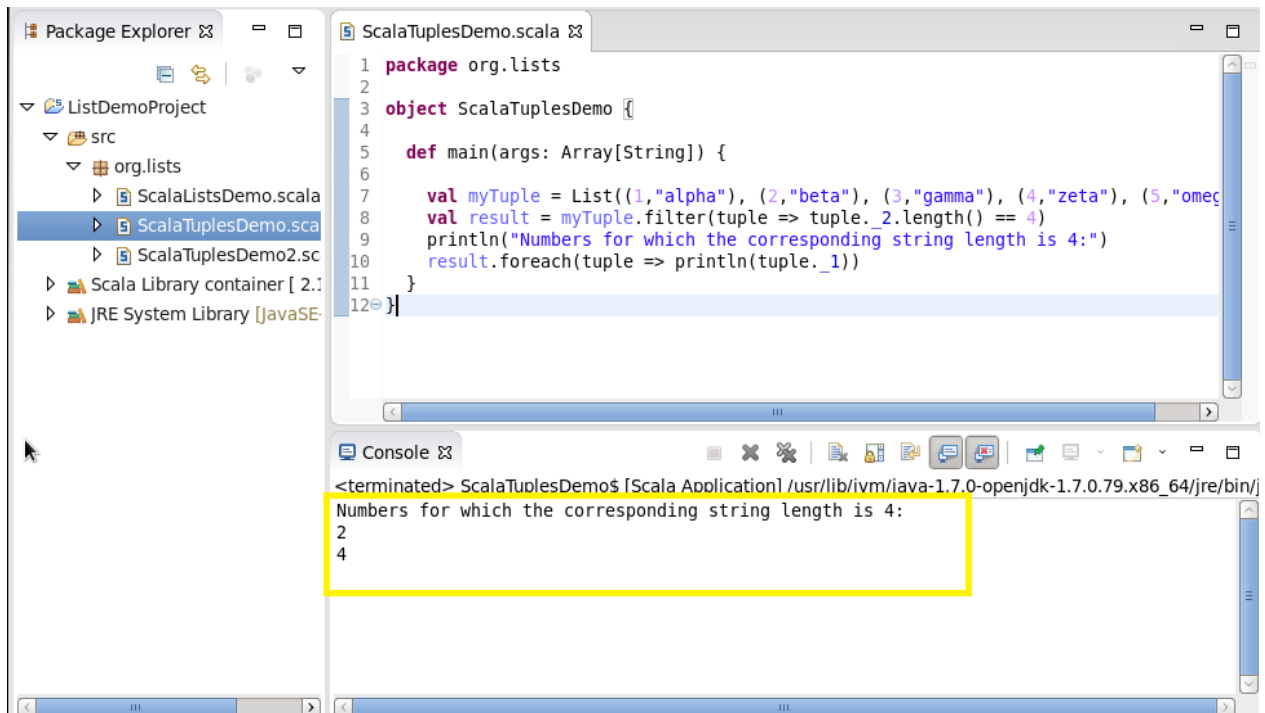
Line 4: Print result to the console.

### **Output:**

Numbers for which the corresponding string length is 4:

2

4



2. Find the average of all numbers, where the corresponding string contains alphabet 'm' or 'z'

Here is the code snippet I have written in Scala to get the required result:

```
val myTuple = List((1,"alpha"), (2,"beta"), (3,"gamma"), (4,"zeta"), (5,"omega"))

val result = myTuple.filter(tuple => tuple._2.contains('m') || tuple._2.contains('z'))

val keys = result.map(tuple => tuple._1)

println("Average of all numbers where the corresponding string contains 'm' or 'z': "

      + keys.sum / keys.length)
```

### Comments:

Line 1: Initialize a list with few tuples (key, value pairs) of our choice. The resulting value is of type, List [(Int, String)]

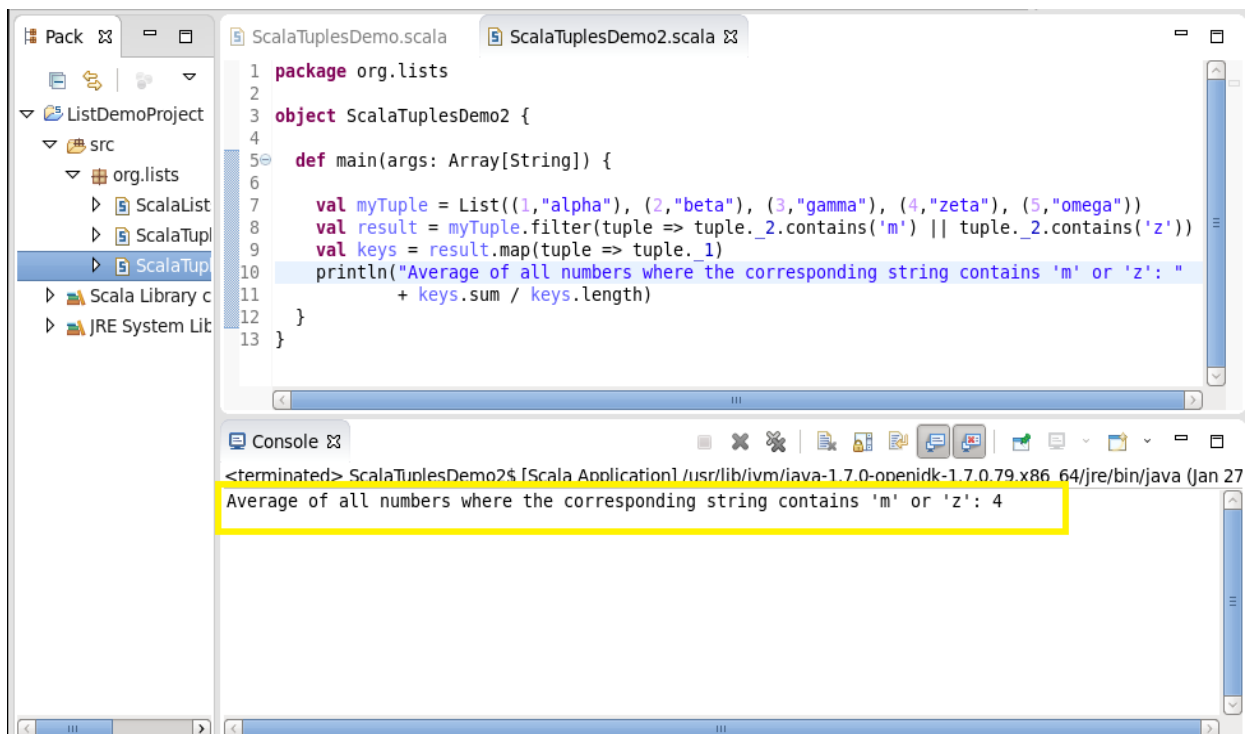
Line 2: Apply filter() method to get only those tuples in which the String value contains either of these two alphabets: 'm' or 'z'. The resulting value is of same type as previous one.

Line 3: Apply map function (for one-to-one mapping of list elements) that gives us only the key element of each tuple from the list of tuples. The resulting value is of type, List [Int]

Line 4: Calculate average of all the integer values from the list we obtained in previous step and print it.

### Output:

Average of all numbers where the corresponding string contains 'm' or 'z': 4



The screenshot shows an IDE with two tabs: `ScalaTuplesDemo.scala` and `ScalaTuplesDemo2.scala`. The `ScalaTuplesDemo2.scala` tab is active, displaying the following code:

```
1 package org.lists
2
3 object ScalaTuplesDemo2 {
4
5     def main(args: Array[String]) {
6
7         val myTuple = List((1,"alpha"), (2,"beta"), (3,"gamma"), (4,"zeta"), (5,"omega"))
8         val result = myTuple.filter(tuple => tuple._2.contains('m') || tuple._2.contains('z'))
9         val keys = result.map(tuple => tuple._1)
10        println("Average of all numbers where the corresponding string contains 'm' or 'z': "
11                + keys.sum / keys.length)
12    }
13 }
```

The `Console` tab at the bottom shows the output of the program:

```
<terminated> ScalaTuplesDemo2$ [Scala Application] /usr/lib/ivm/java-1.7.0-openjdk-1.7.0.79.x86_64/jre/bin/java (Jan 27
Average of all numbers where the corresponding string contains 'm' or 'z': 4
```

The output line is highlighted with a yellow box.