Scala

ACADGILD

Assignment 14.1

Ashish Kumar

Scala Session Basics 1

Assignment 14.1

Task 1

Given a list of strings - List[String] (“alpha”, “gamma”, “omega”, “zeta”, “beta”)

- Find count of all strings with length 4.

```
val task1 = List[String] ("alpha", "gamma", "omega", "zeta", "beta")

println (task1.count(result => result.length == 4))
```

```
scala> val task1 = List[String] ("alpha", "gamma", "omega", "zeta","beta")
task1: List[String] = List(alpha, gamma, omega, zeta, beta)

scala> println(task1.count(result => result.length == 4))
2

scala>
```

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

Here we have used **map** method to map list of string to list of integers and **length** method to find out length of corresponding element and **toInt** method to convert it to integer.
So, it gives below result:

```
val map_int = task1.map (x => x.length().toInt)
```

- Find count of all strings which contain alphabet ‘m’.

There are two elements **“gamma”** and **“omega”** in alphabets list. So, count is 2.

```
val m_count = task1.count(x => x.contains(‘m’))
```

- Find the count of all strings which start with the alphabet ‘a’.

There is only one element **“alpha”** which start with alphabet ‘a’ in alphabets list. So, count is **1**.

```
val a_count = task1.count(x => x.charAt(0)==’a’)
```

Scala Session Basics 1

Assignment 14.1

Task 2

Create a list of tuples, where the 1st element of the tuple is an int and the second element is a string.
Example - ((1, 'alpha'), (2, 'beta'), (3, 'gamma'), (4, 'zeta'), (5, 'omega'))
- For the above list, print the numbers where the corresponding string length is 4.

Creating List of elements named tuples as

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

then, using filter to find the number of where the corresponding string length is 4 as below code –

tuples.filter(_._2.length == 4).foreach(x => println(x._1))

```
scala> val tuples = List((1, "alpha"), (2, "beta"), (3, "gamma"), (4, "zeta"), (5, "omega"))
tuples: List[(Int, String)] = List((1,alpha), (2,beta), (3,gamma), (4,zeta), (5,omega))

scala> tuples.filter(_._2.length == 4).foreach(x => println(x._2))
beta
zeta

scala> tuples.filter(_._2.length == 4).foreach(x => println(x._1))
2
4

scala>
```

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

Average method is not a readily available, as it is not a built-in scala function.
Hence we have created an **average** method as shown below by using **foldleft** function which is using 0 as first value and starts from the left side and iterates to the right till the last element in the list:

def average(a: List[Int]) = { val sum: Float = a.foldLeft(0){ case (a,b) => a + b }; sum / a.length }

```
scala> def average(a: List[Int]) = { val sum: Float = a.foldLeft(0){ case (a,b) => a + b }; sum / a.length }
average: (a: List[Int])Float
```

Then we have created **list_of_numbers** as List. Then we have List of numbers whose corresponding string contains either character 'm' or 'z' as shown below.

val list_of_numbers = tuples.collect{case(integer,string) if string.contains('m') || string.contains('z') => integer}

Then we have used this **average** method which we have created to apply it on **list_of_numbers**.

As we are applying average method on List (3,4,5), it is giving average as 4.0

val average_of_numbers = average(list_of_numbers)

```
scala> val average_of_numbers = average(list_of_numbers)
average_of_numbers: Float = 4.0
```