BigData Assignment 5.3

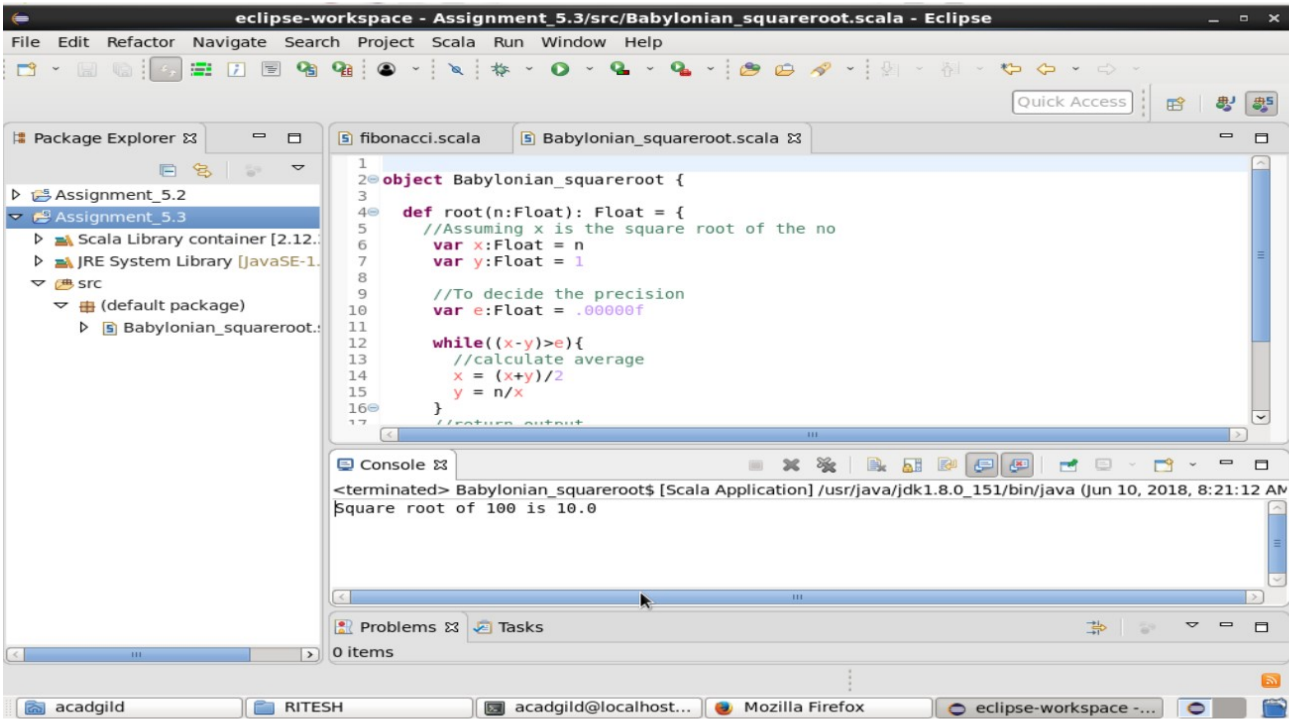
Find square root of number using Babylonian method.

Solution -

```
object Babylonian_squareroot {

  def root(n:Float): Float = {
    //Assuming x is the square root of the no
     var x:Float = n
     var y:Float = 1

     //To decide the precision
     var e:Float = .00000f

     while((x-y)>e){
       //calculate average
       x = (x+y)/2
       y = n/x
     }
     //return output
     x
  }
  def main(args: Array[String]){
    println("Square root of "+args(0)+" is
"+root(args(0).toInt))
  }
}
```

Input - 100

Output - As in the below screenshot , in the console window it is visible that sqaure root of 100 is 10.0 which was calculated using Babylonian method.


eclipse-workspace - Assignment_5.3/src/Babylonian_squareroot.scala - Eclipse
File Edit Refactor Navigate Search Project Scala Run Window Help
Quick Access
Package Explorer
Assignment_5.2
Assignment_5.3
Scala Library container [2.12.
JRE System Library [JavaSE-1.
src
(default package)
Babylonian_squareroot.
fibonacci.scala
Babylonian_squareroot.scala
Console
Problems
Tasks
0 items
acadgild
RITESH
acadgild@localhost...
Mozilla Firefox
eclipse-workspace -...

```

object Babylonian_squareroot {

  def root(n:Float): Float = {
    //Assuming x is the square root of the no
     var x:Float = n
     var y:Float = 1

     //To decide the precision
     var e:Float = .00000f

     while((x-y)>e){
       //calculate average
       x = (x+y)/2
       y = n/x
     }
     //return output
```

```
<terminated> Babylonian_squareroot$ [Scala Application] /usr/java/jdk1.8.0_151/bin/java (Jun 10, 2018, 8:21:12 AM
Square root of 100 is 10.0
```