

BigData Assignment 5.3

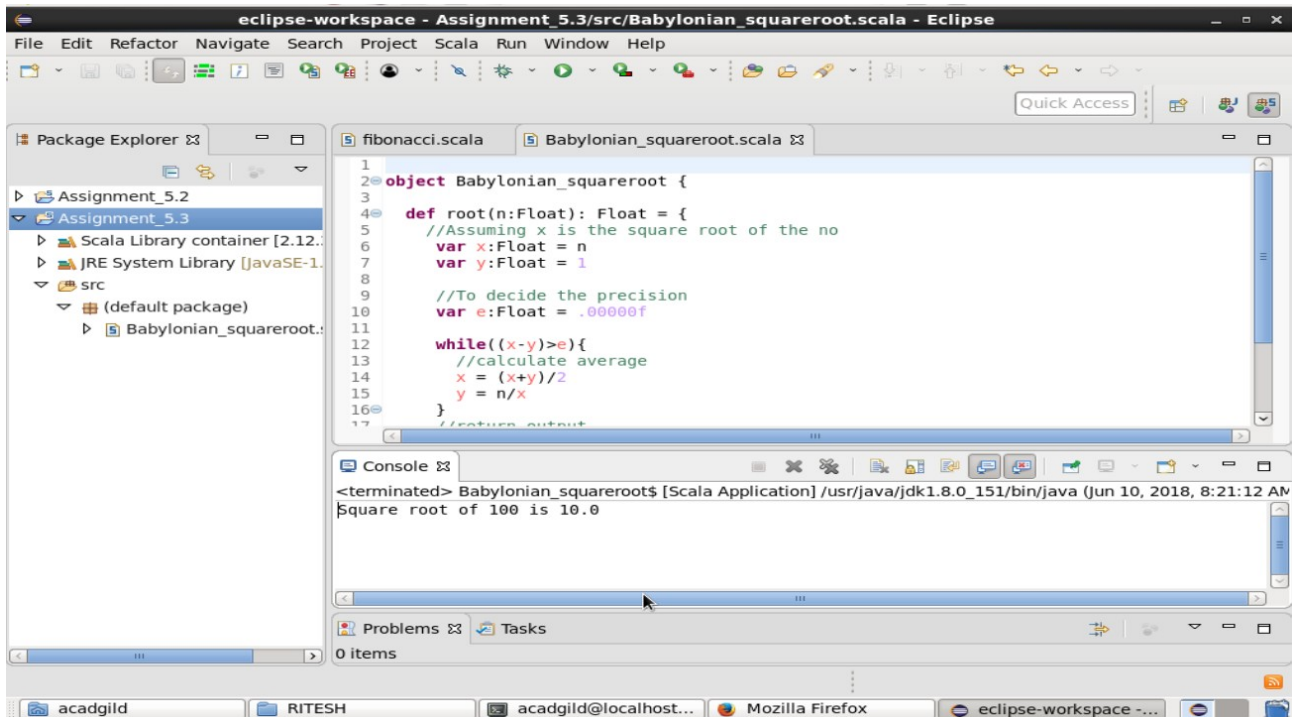
Find square root of number using Babylonian method.

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 square root of 100 is 10.0 which was calculated using Babylonian method.



The screenshot displays the Eclipse IDE interface. The Package Explorer on the left shows the project structure: Assignment_5.2, Assignment_5.3, Scala Library container [2.12.], JRE System Library [JavaSE-1.], src, (default package), and Babylonian_sqrt.scala. The main editor window shows the code for Babylonian_sqrt.scala, which defines a function root(n:Float): Float. The function uses a while loop to calculate the square root of n using the Babylonian method. The console window at the bottom shows the output: <terminated> Babylonian_sqrt [Scala Application] /usr/java/jdk1.8.0_151/bin/java (Jun 10, 2018, 8:21:12 AM) Square root of 100 is 10.0. The Problems and Tasks windows are also visible at the bottom.

```
1 object Babylonian_sqrt {
2
3
4 def root(n:Float): Float = {
5   //Assuming x is the square root of the no
6   var x:Float = n
7   var y:Float = 1
8
9   //To decide the precision
10  var e:Float = .000001f
11
12  while((x-y)>e){
13    //calculate average
14    x = (x+y)/2
15    y = n/x
16  }
17  //return output
```

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