Session 13 Assignment 3

To find the square root of a number using Babylonian method.

```
[bigdata@localhost Session13Assignment3]$ scala Session13Assignment3.scala 15
Square Root of 15 is 3.8729837
[bigdata@localhost Session13Assignment3]$ scala Session13Assignment3.scala 9
Square Root of 9 is 3.0
[bigdata@localhost Session13Assignment3]$ scala Session13Assignment3.scala 8
Square Root of 8 is 2.828427
[bigdata@localhost Session13Assignment3]$ ls
```

Session13Assignment3.scala

```
//Session 13 Assignment 3 - Find the square root of a number using Babylonian Method
object Session13Assignment3 {
        def squareRoot(a:Float): Float = {
                //Assume that the number is the square root itself
                var x:Float=a

                //Set y to 1;
                var y:Float=1

                //Set the variable to decide the number of digits for precision
                var e:Float=.000001f

                //Loop till the difference is greater than e
                while (x-y>e){
                 //Calculate average
                 x=(x+y)/2
                 y=a/x
                }
                //return the final value
                x
        }

        def main(args: Array[String]) {
      //Accept the input from command line and pass it to Square Root function
        println("Square Root of " + args(0) + " is " + squareRoot(args(0).toInt));
        }
}
```