**Problem Statement**

* Find square root of number using Babylonian method.

1 Start with an arbitrary positive start value x (the closer to the root, the better).

2 Initialize y = 1.

3 Do following until desired approximation is achieved.

a) Get the next approximation for root using average of x and y

b) Set y = n/x

**Solution:**

//package declaration

**package** assignment13\_3

//Importing Scanner Class to take input from user

**import** java.util.Scanner;

//Scala Object declaration

**object** Babylonian

{

//Main Method

**def** main(args: Array[*String*]): Unit = {

println("--------Babylonian Method To Find Square Root--------")

println("Enter the Number:")

**var** n = scala.io.StdIn.readFloat()//user input of number

**var** x = n

**var** y =1.toFloat

**var** e = 0.000001 /\* e decides the accuracy level\*/

//Logic to find square root using Babylonian method

**while**(x-y > e)

{

x = (x+y)/2

y= n/x

}

//Displaying Square Root

println("Square Root of " + n + " is :" + x)

}//end of main method

}//closing scala object

**Output:**

