JUnit - Time Test

JUnit provides a handy option of Timeout. If a test case takes more time than the specified number of milliseconds, then JUnit will automatically mark it as failed. The **timeout** parameter is used along with @Test annotation. Let us see the @Test(timeout) in action.

Create a Class

Create a java class to be tested, say, **MessageUtil.java** in C:\>JUNIT_WORKSPACE.

Add an infinite while loop inside the printMessage() method.

```
/*
* This class prints the given message on console.
*/

public class MessageUtil {

   private String message;

   //Constructor
   //@param message to be printed
   public MessageUtil(String message){
      this.message = message;
   }

   // prints the message
   public void printMessage(){
      System.out.println(message);
      while(true);
   }

   // add "Hi!" to the message
   public String salutationMessage(){
      message = "Hi!" + message;
      System.out.println(message);
      return message;
   }
}
```

Create Test Case Class

Create a java test class, say, **TestJunit.java**. Add a timeout of 1000 to testPrintMessage() test case.

Create a java class file named **TestJunit.java** in C:\>JUNIT_WORKSPACE.

```
import org.junit.Test;
import org.junit.Ignore;
import static org.junit.Assert.assertEquals;

public class TestJunit {

   String message = "Robert";
   MessageUtil messageUtil = new MessageUtil(message);

   @Test(timeout = 1000)
   public void testPrintMessage() {
      System.out.println("Inside testPrintMessage()");
      messageUtil.printMessage();
   }

   @Test
   public void testSalutationMessage() {
      System.out.println("Inside testSalutationMessage()");
      message = "Hi!" + "Robert";
      assertEquals(message,messageUtil.salutationMessage());
   }
}
```


Create Test Runner Class

Create a java class file named **TestRunner.java** in C:\>JUNIT_WORKSPACE to execute test case(s).

```
import org.junit.runner.JUnitCore;
import org.junit.runner.Result;
import org.junit.runner.notification.Failure;

public class TestRunner {
```

```
   public static void main(String[] args) {
      Result result = JUnitCore.runClasses(TestJunit.class);

      for (Failure failure : result.getFailures()) {
         System.out.println(failure.toString());
      }

      System.out.println(result.wasSuccessful());
   }
}
```

Compile the MessageUtil, Test case and Test Runner classes using javac.

```
C:\JUNIT_WORKSPACE>javac MessageUtil.java TestJunit.java TestRunner.java
```

Now run the Test Runner, which will run the test cases defined in the provided Test Case class.

```
C:\JUNIT_WORKSPACE>java TestRunner
```

Verify the output. testPrintMessage() test case will mark the unit testing failed.

```
Inside testPrintMessage()
Robert
Inside testSalutationMessage()
Hi!Robert
testPrintMessage(TestJunit): test timed out after 1000 milliseconds
false
```