Unit Testing

* 1. Main
     1. Zach
  2. ADSBInterface
     1. Josh

import junit.framework.TestCase;

import java.util.Random;

public class ADSBInterfaceTest extends TestCase {

Aircraft[] testNearList = new Aircraft[40];

ADSBInterface testClass;

Random rng;

public static void main(String[] args)

{

ADSBInterfaceTest it = new ADSBInterfaceTest();

}

public ADSBInterfaceTest()

{

rng = new Random();

double[] l = new double[4];

double[] h = new double[4];

for (int i = 0; i < 40; i++)

{

for(int j = 0; j < 4; j++)

{

l[j] = 0;

h[j] = 0;

}

String id;

if(i < 10)

id = "TST100" + i;

else

id = "TST10" + i;

testNearList[i] = new Aircraft(id, l, h);

}

testClass = new ADSBInterface(null);

try

{

testAddAircraft();

}

catch(Exception e)

{

System.out.println("Error occurred: " + e.getMessage());

}

}

public void testAddAircraft() throws Exception

{

int result;

//test the addition of 1 aircraft

result = testClass.addAircraft(testNearList[0]);

System.out.println("Aircraft " + testNearList[0].getId() + " added. Result: " +

determineResult(result));

//test the addition of multiple aircraft <= 20

for(int i = 1; i < 20; i++)

{

result = testClass.addAircraft(testNearList[i]);

System.out.println("Aircraft " + testNearList[i].getId() + " added. Result: " +

determineResult(result));

}

//test updating records by resending all of the first 20

for(int i = 0; i < 20; i++)

{

result = testClass.addAircraft(testNearList[i]);

System.out.println("Aircraft " + testNearList[i].getId() + " added. Result: " +

determineResult(result));

}

//test the addition of > 20 aircraft. This one should always error out.

for(int i = 20; i < 40; i++)

{

result = testClass.addAircraft(testNearList[i]);

System.out.println("Aircraft " + testNearList[i].getId() + " added. Result: " +

determineResult(result));

}

}

public String determineResult(int result)

{

String report;

switch(result)

{

case 1:

report = "passed as new add.";

break;

case 2:

report = "passed as updated aircraft.";

break;

case 3:

report = "failed - too many elements for array.";

break;

case 0:

default:

report = "failed.";

break;

}

return report;

}

}

* 1. Aircraft
     1. Lauren
  2. Collision
     1. Shelley
  3. CollisionCalculator
     1. Adam
  4. WarningLevelCalculator
     1. Casey
  5. TextCommunicator
     1. Christi

**import java.util.Date;**

**import org.junit.After;**

**import org.junit.AfterClass;**

**import org.junit.Before;**

**import org.junit.BeforeClass;**

**import org.junit.Test;**

**import static org.junit.Assert.\*;**

**/\*\***

**\***

**\* @author Christi Kazakov**

**\*/**

**public class TextCommunicationIT {**

**public TextCommunicationIT() {**

**TextCommunication instance = new TextCommunication();**

**}**

**@BeforeClass**

**public static void setUpClass() {**

**}**

**@AfterClass**

**public static void tearDownClass() {**

**}**

**@Before**

**public void setUp() {**

**}**

**@After**

**public void tearDown() {**

**}**

**/\*\***

**\* Test of listener method, of class TextCommunication.**

**\*/**

**@Test**

**public void testListener() {**

**System.out.println("listener");**

**TextCommunication instance = new TextCommunication();**

**instance.listener();**

**// TODO review the generated test code and remove the default call to fail.**

**//fail("The test case is a prototype.");**

**}**

**/\*\***

**\* Test of receiver method, of class TextCommunication.**

**\*/**

**@Test**

**public void testReceiver() {**

**System.out.println("receiver");**

**Object data = new Object();**

**TextCommunication instance = new TextCommunication();**

**String expResult = data.toString();**

**String result = instance.receiver(data);**

**assertEquals(expResult, result);**

**// TODO review the generated test code and remove the default call to fail.**

**//fail("The test case is a prototype.");**

**}**

**/\*\***

**\* Test of send method, of class TextCommunication.**

**\*/**

**@Test**

**public void testSend() {**

**System.out.println("send");**

**String message = " ";**

**String message2 = "Warning Level 1";**

**String message3 = "MayDay MayDay";**

**TextCommunication instance = new TextCommunication();**

**instance.send(message);**

**instance.send(message2);**

**instance.send(message3);**

**// TODO review the generated test code and remove the default call to fail.**

**//fail("The test case did not go through");**

**}**

**/\*\***

**\* Test of log method, of class TextCommunication.**

**\*/**

**@Test**

**public void testLog() {**

**Date date = new Date();**

**System.out.println("log");**

**String log = " ";**

**String logWarning = "Warning level 1 ";**

**TextCommunication instance = new TextCommunication();**

**instance.log(date.toString() + " : " + logWarning);**

**// TODO review the generated test code and remove the default call to fail.**

**//fail("The test case is a prototype.");**

**}**

**}**