IT314 -Software Engineering

Lab-8 Unit Testing with JUnit

Name: Aniruddh Damor
ID: 202001255

➢ **Sample code for the ‘Boa’ class created for testing.**

```
// represents a boa constrictor
public class Boa {
        private String name;
        private int length; // the length of the boa, in feet
        private String favoriteFood;
        public Boa (String name, int length, String favoriteFood){
                this.name = name;
                this.length = length;
                this.favoriteFood = favoriteFood;
        }
        // returns true if this boa constrictor is healthy
        public boolean isHealthy(){
                return this.favoriteFood.equals("granola bars");
        }
        // returns true if the length of this boa constrictor is
        // less than the given cage length
        public boolean fitsInCage(int cageLength){
                return this.length < cageLength;
        }
}
```

➢ **Now we create a Junit Test case for testing the above ‘Boa’ class and named it ‘Boa_test’.**

```
package test_lab8;
import static org.junit.jupiter.api.Assertions.*;
import org.junit.Before;
import org.junit.jupiter.api.Test;
class Boa_test {
  @Test
  public void testIsHealthyWithFavoriteFoodGranolaBars() {
      Boa boa = new Boa("Benny", 5, "granola bars");
      assertTrue(boa.isHealthy());
  }

  @Test
  public void testIsHealthyWithFavoriteFoodNotGranolaBars() {
      Boa boa = new Boa("Benny", 5, "mice");
      assertFalse(boa.isHealthy());
  }

  @Test
  public void testFitsInCageWhenLengthLessThanCageLength() {
      Boa boa = new Boa("Benny", 5, "granola bars");
      assertTrue(boa.fitsInCage(10));
  }

  @Test
  public void testFitsInCageWhenLengthGreaterThanCageLength() {
      Boa boa = new Boa("Benny", 20, "granola bars");
      assertFalse(boa.fitsInCage(10));

 }
}
```

- Now we ran the above test cases successfully.
- Now we use the **@Before** tag to run another test case.
- The **@Before tag** runs before every test case(@Test).

➢ Q4: The code snippet below show how to execute it.

```
   private Boa jen;
   private Boa ken;
```

```
  @Before
  public void setUp() throws Exception {
            jen = new Boa("Jennifer", 2, "grapes");
            ken = new Boa("Kenneth", 3, "granola bars");

}
```

➢ Q5: Below is the test code for the question.

```
  @Test
  public void testIsHealthy() {
          Boa jen = new Boa("Jen", 5, "granola bars");
      Boa ken = new Boa("Ken", 6, "mice");

    assertTrue(jen.isHealthy());
    assertFalse(ken.isHealthy());
  }

  @Test
  public void testFitsInCage() {
          Boa jen = new Boa("Jen", 5, "granola bars");
      Boa ken = new Boa("Ken", 6, "mice");

    assertFalse(jen.fitsInCage(2));
    assertTrue(jen.fitsInCage(10));
    assertTrue(jen.fitsInCage(15));
    assertFalse(ken.fitsInCage(4));
    assertTrue(ken.fitsInCage(15));
    assertTrue(ken.fitsInCage(20));
  }
```

➢ Q7: Creating another method in the 'Boa' class.

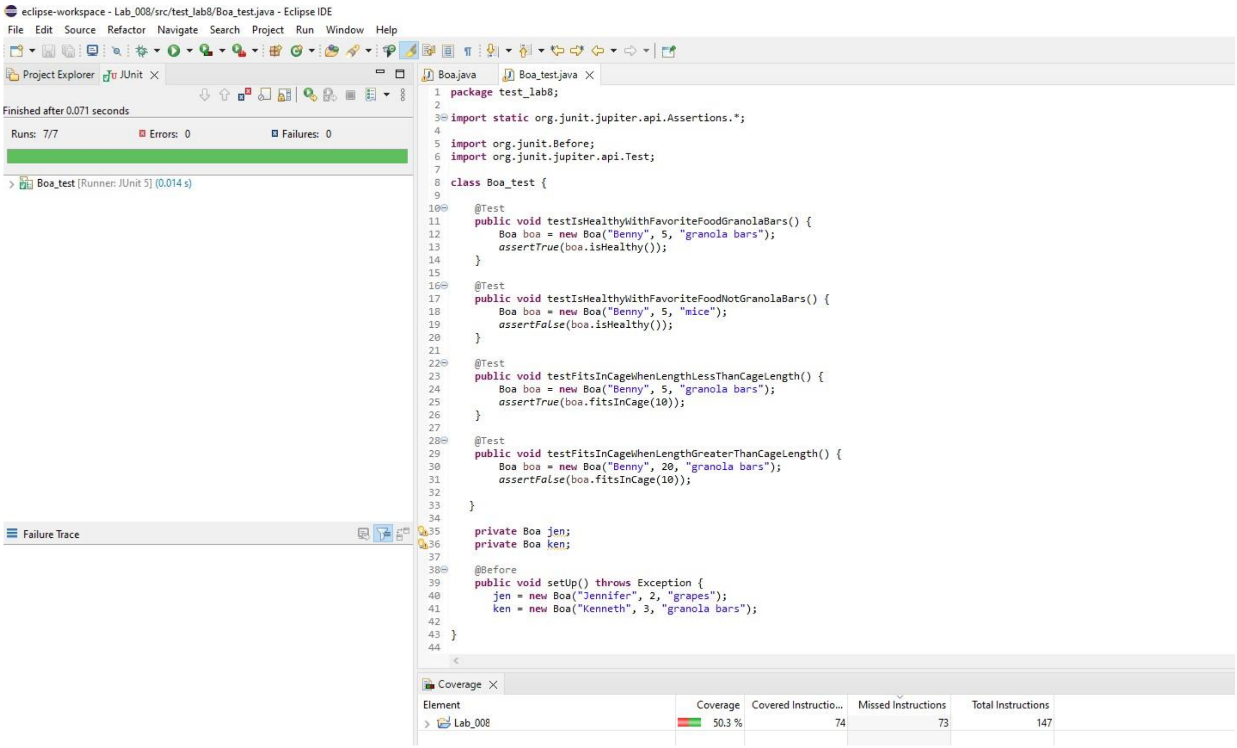
Method Code:

```
public int lengthInInches() {
  return this.length * 12;
}
```

Testing code:

```
@Test
  public void testLengthInInches() {
        Boa boa = new Boa("John", 5, "grapes");
        int expectedLengthInInches = 60;
        int actualLengthInInches = boa.lengthInInches();
        assertEquals(expectedLengthInInches, actualLengthInInches);
  }
```

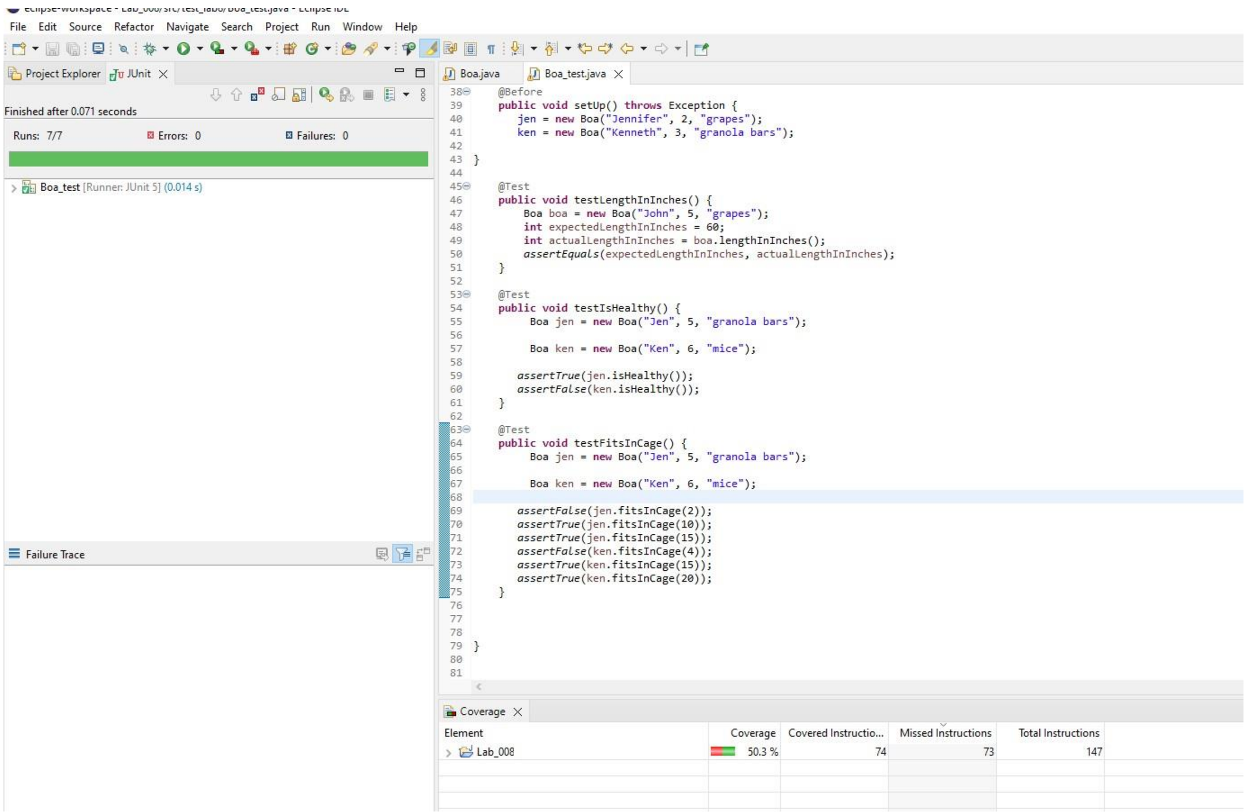
- □ Here is the snippet for the complete testing code.


eclipse-workspace - Lab_008/src/test_lab8/Boa_test.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
JUnit
Boa.java
Boa_test.java
Failure Trace
Coverage

```
Finished after 0.071 seconds

Runs: 7/7          Errors: 0          Failures: 0

> Boa_test [Runner: JUnit 5] (0.014 s)
```

```java
package test_lab8;

import static org.junit.jupiter.api.Assertions.*;

import org.junit.Before;
import org.junit.jupiter.api.Test;

class Boa_test {

    @Test
    public void testIsHealthyWithFavoriteFoodGranolaBars() {
        Boa boa = new Boa("Benny", 5, "granola bars");
        assertTrue(boa.isHealthy());
    }

    @Test
    public void testIsHealthyWithFavoriteFoodNotGranolaBars() {
        Boa boa = new Boa("Benny", 5, "mice");
        assertFalse(boa.isHealthy());
    }

    @Test
    public void testFitsInCageWhenLengthLessThanCageLength() {
        Boa boa = new Boa("Benny", 5, "granola bars");
        assertTrue(boa.fitsInCage(10));
    }

    @Test
    public void testFitsInCageWhenLengthGreaterThanCageLength() {
        Boa boa = new Boa("Benny", 20, "granola bars");
        assertFalse(boa.fitsInCage(10));

   }

    private Boa jen;
    private Boa ken;

    @Before
    public void setUp() throws Exception {
       jen = new Boa("Jennifer", 2, "grapes");
       ken = new Boa("Kenneth", 3, "granola bars");

}

```

Element	Coverage	Covered Instructio...	Missed Instructions	Total Instructions
> Lab_008	50.3 %	74	73	147


eclipse-workspace - Lab_008/src/test_lab8/Boa_test.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
JUnit
Boa.java
Boa_test.java
Failure Trace
Coverage

Finished after 0.071 seconds

Runs: 7/7 Errors: 0 Failures: 0

Boa_test [Runner: JUnit 5] (0.014 s)

```java
    @Before
    public void setUp() throws Exception {
       jen = new Boa("Jennifer", 2, "grapes");
       ken = new Boa("Kenneth", 3, "granola bars");

}

    @Test
    public void testLengthInInches() {
        Boa boa = new Boa("John", 5, "grapes");
        int expectedLengthInInches = 60;
        int actualLengthInInches = boa.lengthInInches();
        assertEquals(expectedLengthInInches, actualLengthInInches);
    }

    @Test
    public void testIsHealthy() {
         Boa jen = new Boa("Jen", 5, "granola bars");

         Boa ken = new Boa("Ken", 6, "mice");

       assertTrue(jen.isHealthy());
       assertFalse(ken.isHealthy());
    }

    @Test
    public void testFitsInCage() {
         Boa jen = new Boa("Jen", 5, "granola bars");

         Boa ken = new Boa("Ken", 6, "mice");

       assertFalse(jen.fitsInCage(2));
       assertTrue(jen.fitsInCage(10));
       assertTrue(jen.fitsInCage(15));
       assertFalse(ken.fitsInCage(4));
       assertTrue(ken.fitsInCage(15));
       assertTrue(ken.fitsInCage(20));
    }



}
```

Element	Coverage	Covered Instructio...	Missed Instructions	Total Instructions
Lab_008	50.3 %	74	73	147