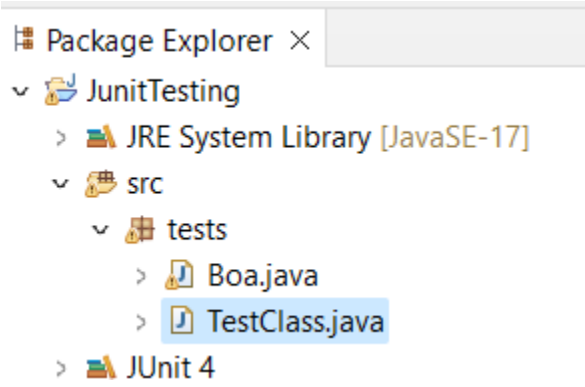
Dhirubhai Ambani
Institute of Information and Communication Technology

Name - **Kavish Shah**
Student ID - **202001114**
Course - **Software Engineering (IT314)**

Lab 8

1. Creating a new eclipse project and a package within the project



2. Creating a class for Boa

Boa.java × TestClass.java

```java
package tests;

// 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;
    }
}
```

3. Creating a JUnit test case with methods isHealthy and fitsInCage

Boa.java | TestClass.java

```java
package tests;

import static org.junit.Assert.*;
import org.junit.Before;
import org.junit.Test;

public class TestClass {
    private Boa jen, ken;

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

    @Test
    public void isHealthy() {
        fail("Not yet implemented");
    }

    @Test
    public void fitsInCage() {

    }
}
```

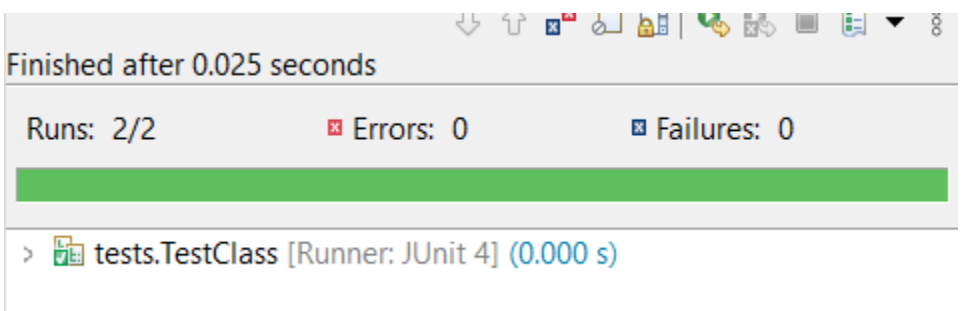
4. Creating setup method and annotating with @before and creating jen and ken objects of boa class

```java
public class TestClass {
    private Boa jen, ken;

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

5. Writing tests for fitsInCage and isHealthy method

```
    @Test
    public void testIsHealthy() {
        assertFalse(jen.isHealthy());
        assertTrue(ken.isHealthy());
    }

    @Test
    public void testFitsInCage() {
        assertTrue(jen.fitsInCage(3));
        assertFalse(jen.fitsInCage(2));
        assertFalse(jen.fitsInCage(1));
        assertFalse(jen.fitsInCage(0));
        assertFalse(jen.fitsInCage(-1));
        assertTrue(ken.fitsInCage(10));
        assertFalse(ken.fitsInCage(3));
        assertFalse(ken.fitsInCage(0));
        assertFalse(ken.fitsInCage(-1));
    }
}
```

6. Running both tests



```
    // produces the length of the Boa in inches
    public int lengthInInches(){
        return 12*this.length;

    }
}
```

7. Writing the tests for length in inches

```
    @Test
    public void testLengthInInches() {
        assertEquals(24, jen.lengthInInches());
        assertEquals(36, ken.lengthInInches());
    }
}
```

8. Running the tests