IT314
Lab-8
Unit Testing with JUnit

Name:Akshar Chaudhari
I'd:202001071
Date:19-04-2023

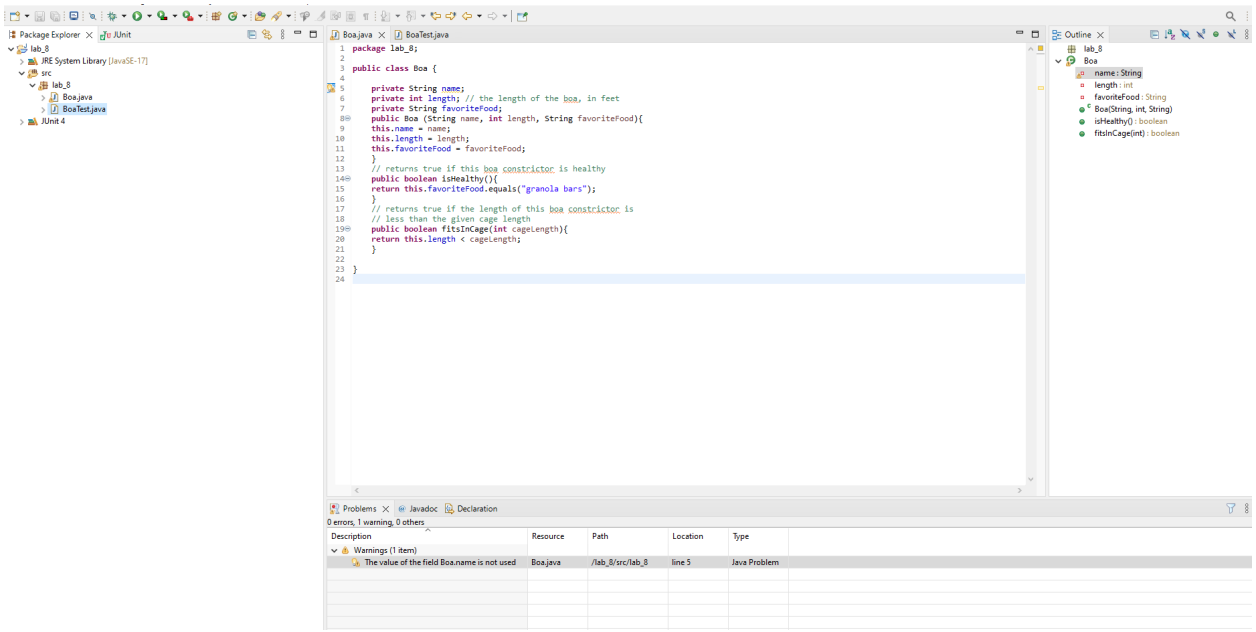
Goal:

The goal of this lab is to learn how to use JUnit to write unit tests for Java programs.

The primary goal of unit testing is to take the smallest piece of testable software in an application, isolate it from the remainder of the code, and determine whether or not it behavesthe way you expect it to behave. Each unit is tested separately before integrating it into therest of the program. In other words, classes should be tested in isolation from other classes(test the methods of a class before you use the class elsewhere). Unit testing has proven itsvalue in that a large percentage of defects are identified during unit testing.

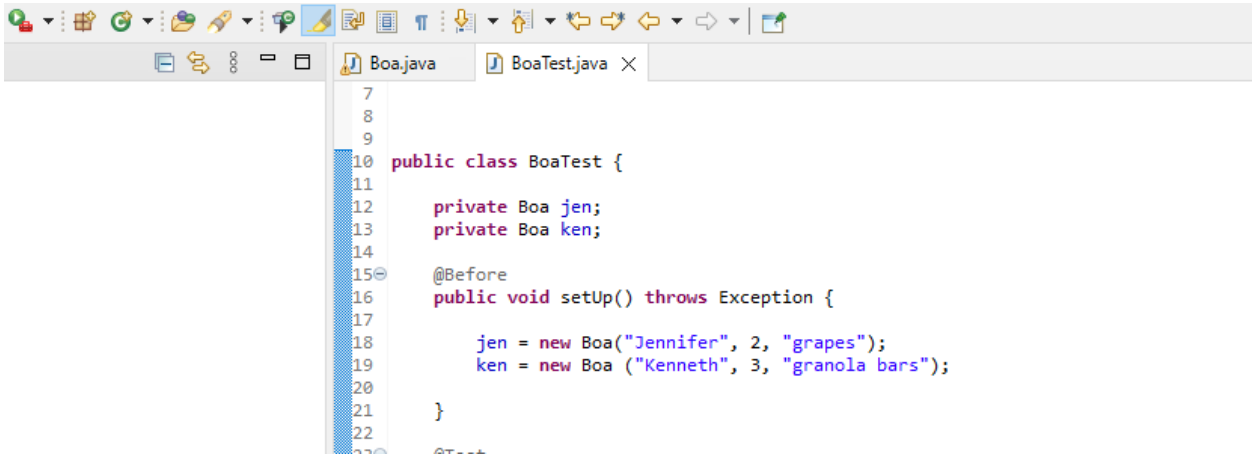
Lab Exercises:

1 .Created a new Eclipse project, and within the project create a package.
2. Created a class for a Boa. Here's the code:



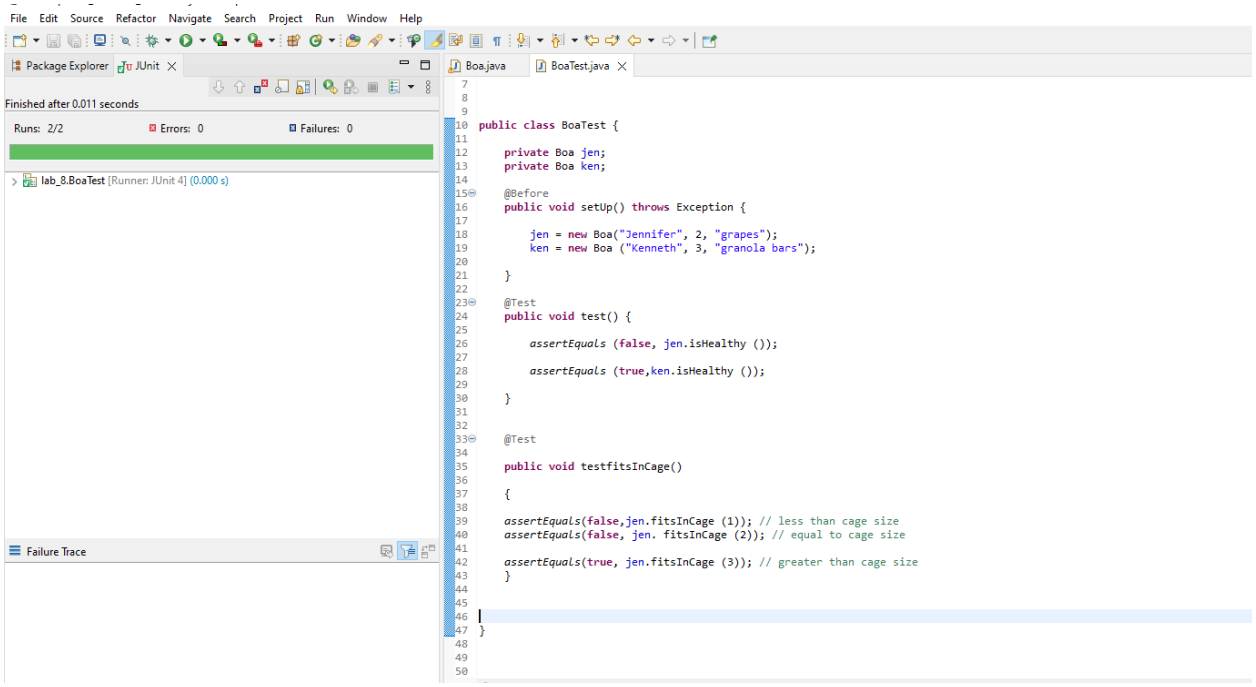
3 .After following the instructions in the JUnit tutorial in the section “Creating a JUnit Test Case in Eclipse”.i created a test case for the class Boa.select test method stubs, select both isHealthy() and fitsInCage(int).

4 .Now it’s time to write some unit tests. I Notice that the BoaTest class that JUnit created for my contains stubs for several methods. The first stub (for the method setUp()) is annotated with @Before. The @Before annotation denotes that the method setUp() will be run prior to the execution of each test method. setUp() is typically used to initialize data needed by each test. Modify the setUp() method so that it creates a couple of Boa objects, as follows:



5 . JUnit also provided stubs for two test methods, each annotated with @Test. Work on the testIsHealthy() method first. The purpose of this method is to check that the isHealthy() method in the Boa class behaves the way it's supposed to. In the JUnit tutorial, read the section on "Writing Tests". Modify the testIsHealthy() method so that it checks the results of activating the isHealthy() method on the two Boa objects you created in setup(). Likewise, modify the testFitsInCage() method to test the results of that method.

6 .Runing th etest case: Showing the green bar.

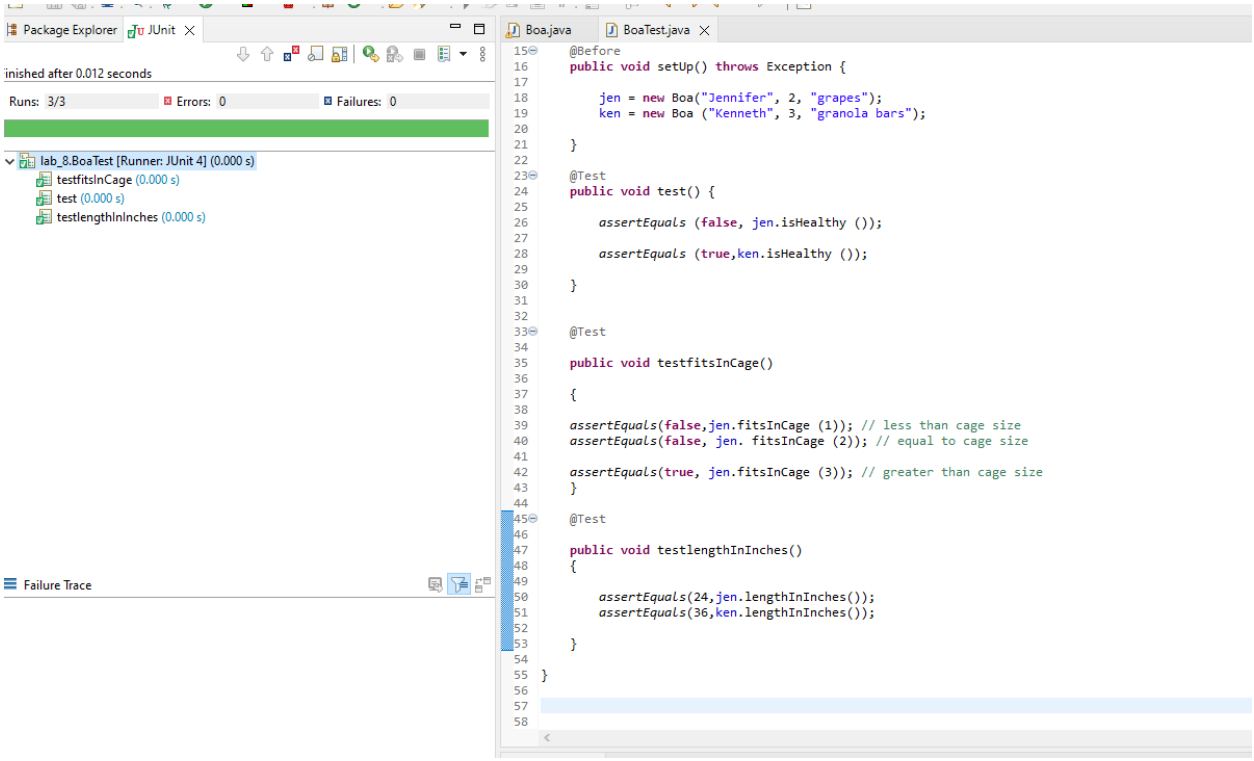


7 .

Add a new method to the Boa class, with this purpose and signature:

```
// produces the length of the Boa in inches
public int lengthInInches(){
// you need to write the body of this method
}
```

Add a new test case to the BoaTest class that tests the lengthInInches() method.



8 . Here are some other things i learn about unit testing and JUnit:

- Each method annotated with @Test will be run, but the order of the tests is not guaranteed.
- Any method annotated with @Before will be run before each test executes.
- Any method annotated with @After will be run after each test executes.