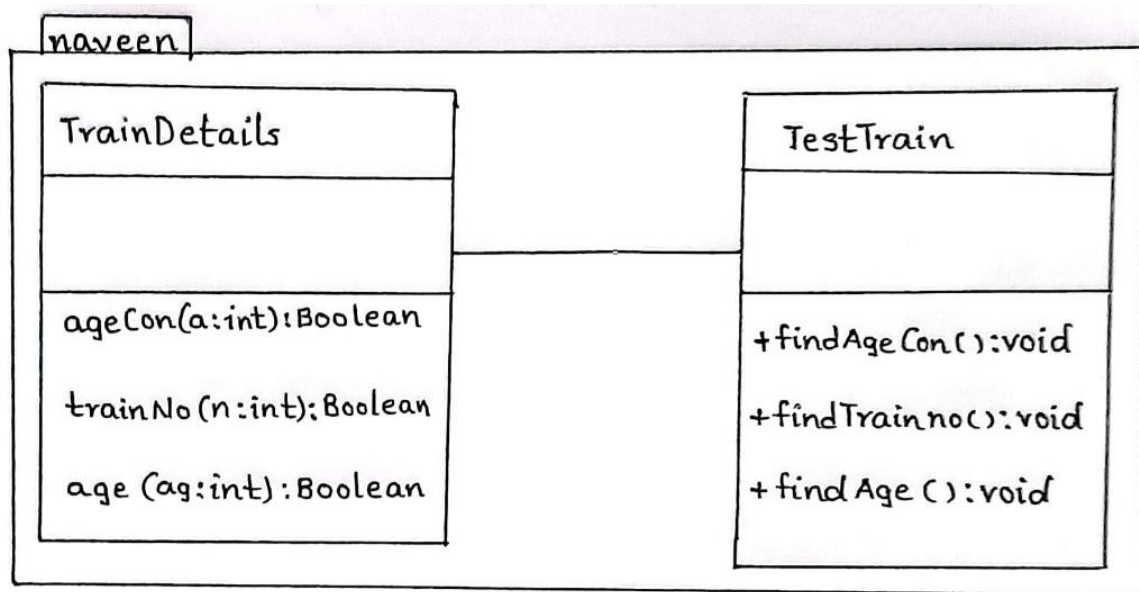


SKILLING EXERCISE-8

Name: Badisa Naveen

Reg.no: 2000031509

Class diagram:



About Junit:

Junit is a “Unit Testing” framework for Java Applications which is already included by default in android studio. It is an automation framework for Unit as well as UI Testing. It contains annotations such as **@Test**, **@Before**, **@After**, etc.

There are two ways to perform unit testing: 1) manual testing 2) automated testing.

- 1) **Manual Testing:** If you execute the test cases manually without any tool support, it is known as manual testing. It is time consuming and less reliable
- 2) **Automated Testing:** If you execute the test cases by tool support, it is known as automated testing. It is fast and more reliable.

Assert:

Assert is a method useful in determining Pass or Fail status of a test case, The assert methods are provided by the class org.junit.Assert which extends java.lang.Object class.

There are various types of assertions like Boolean, Null, Identical etc.

Code:

```
package naveen;
public class Train {
    Boolean ageConcession(int age)
    {
        assert(age>60 && age<100): "Age entered is invalid";
        if(age>60)
            return true;
        else
            return false;
    }
    Boolean trainNo(int no)
    {
        assert(no>0): "Invalid Train Number";
        if(no>0)
            return true;
        else
            return false;
    }
    Boolean age(int a)
    {
        assert(a>0): "Invalid age";
        if(a>0)
            return true;
        else
            return false;
    }
}
```

```
package naveen;
import org.junit.Test;
public class TestTrain {
    Train t = new Train();
    @Test
    public void findAgeConcession()
    {
        t.ageConcession(80);
    }
    @Test
```

```

public void findTrainNumber()
{
    t.trainNo(70);
}
@Test
public void findAge()
{
    t.age(70);
}

}

```

ScreenShots:

Given parameter Age is 40

The screenshot displays the Eclipse IDE interface. The top toolbar shows the 'Run' button (a green play icon) circled in red. Below the toolbar, the 'JUnit' tab is active, showing the test results. The test 'findAgeConcession' is highlighted with a red checkmark, indicating it passed. The 'Failure Trace' section at the bottom shows the error message 'java.lang.AssertionError: Age entered is invalid' circled in red. The code editor on the right shows the source code for 'TestTrain.java', which includes the following code:

```

1 package naveen;
2 import org.junit.Test;
3 public class TestTrain {
4     Train t = new Train();
5     @Test
6     public void findAgeConcession(
7     {
8         t.ageConcession(40);
9     }
10    @Test
11    public void findTrainNumber()
12    {
13        t.trainNo(70);

```

Given Train number -80:

The screenshot shows the Eclipse IDE with the Package Explorer on the left, the JUnit test runner in the center, and the source code on the right. The test runner shows a failure for the `findTrainNumber` test, with a red checkmark indicating a failure. The failure trace shows the error message "java.lang.AssertionError: Invalid Train Number" at line 13 of `TestTrain.java`. The source code shows the `findTrainNumber` method calling `t.trainNo(-8);`.

```
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer JUnit
Finished after 0.023 seconds
Runs: 3/3 Errors: 0 Failures: 1
naveen.TestTrain [Runner: JUnit 4] (0.000 s)
  findTrainNumber (0.000 s)
  findAge (0.000 s)
  findAgeConcession (0.000 s)
Failure Trace
java.lang.AssertionError: Invalid Train Number
  at naveen.Train.trainNo(Train.java:13)
  at naveen.TestTrain.findTrainNumber(TestTrain.java:13)
BinarySearch... StackMaxMin... QueueMaxMin... Test1.java Codechef
9 }
10 @Test
11 public void findTrainNumt
12 {
13     t.trainNo(-8);
14 }
15 @Test
16 public void findAge()
17 {
18     t.age(70);
19 }
20
21
Problems Javadoc Declaration Console Coverage
<terminated> TestTrain [JUnit] C:\Program Files\Java\jdk-16.0.1\bin\javaw.exe (11-Oct-2021, 12:17)
```

Given All parameters correctly:

The screenshot shows the Eclipse IDE with the Package Explorer on the left, the JUnit test runner in the center, and the source code on the right. The test runner shows successful results for all three tests: `findTrainNumber`, `findAge`, and `findAgeConcession`, each with a green checkmark. The source code shows the `TestTrain` class with three test methods: `findAgeConcession`, `findTrainNumber`, and `findAge`.

```
eclipse-workspace - HelloWorld/src/naveen/TestTrain.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer JUnit
Finished after 0.017 seconds
Runs: 3/3 Errors: 0 Failures: 0
naveen.TestTrain [Runner: JUnit 4] (0.001 s)
  findTrainNumber (0.001 s)
  findAge (0.000 s)
  findAgeConcession (0.000 s)
Failure Trace
3 public class TestTrain {
4     Train t = new Train();
5 @Test
6 public void findAgeConcession()
7 {
8     t.ageConcession(80);
9 }
10 @Test
11 public void findTrainNumber()
12 {
13     t.trainNo(70);
14 }
15 @Test
```