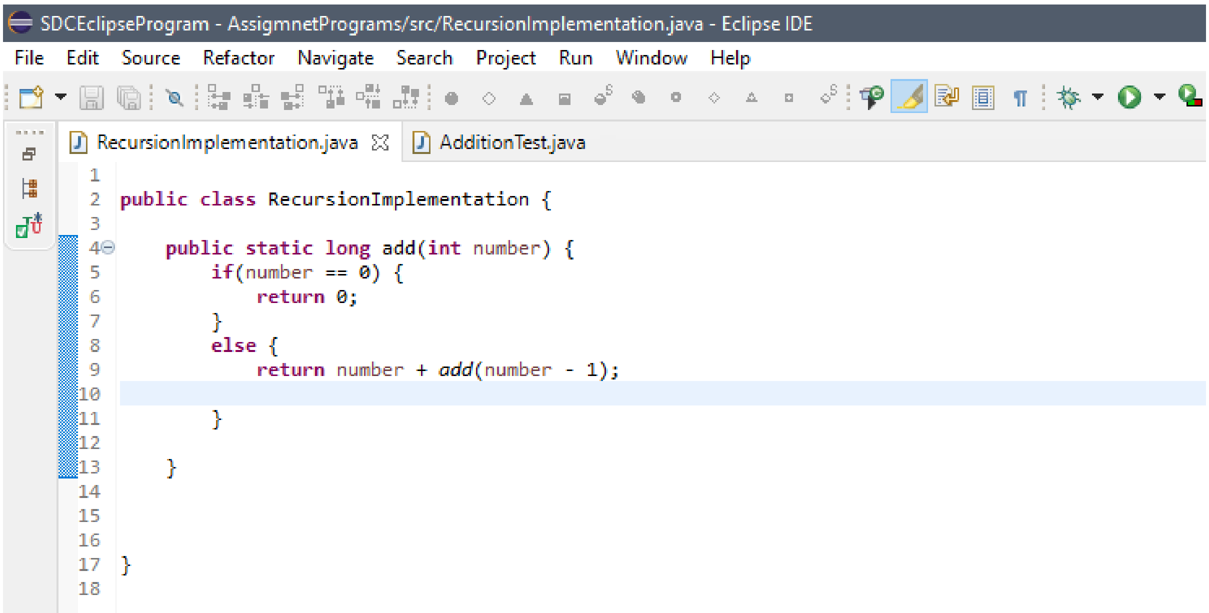
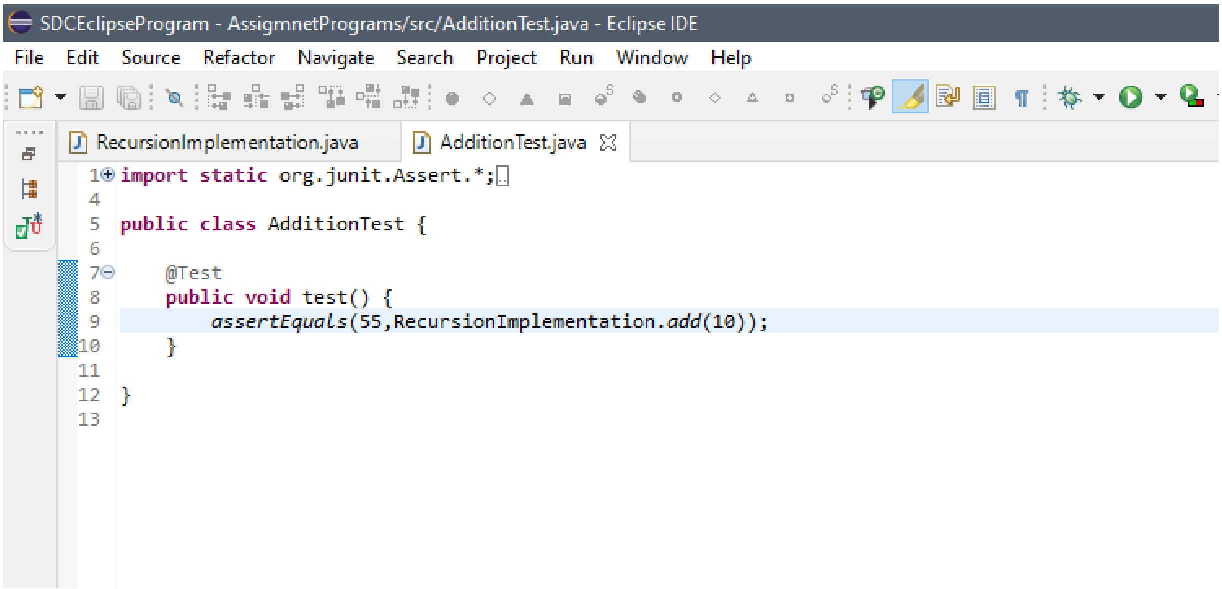
Software Construction and Development (SWE-312)
Assignment 2

SOLUTION 1B :-

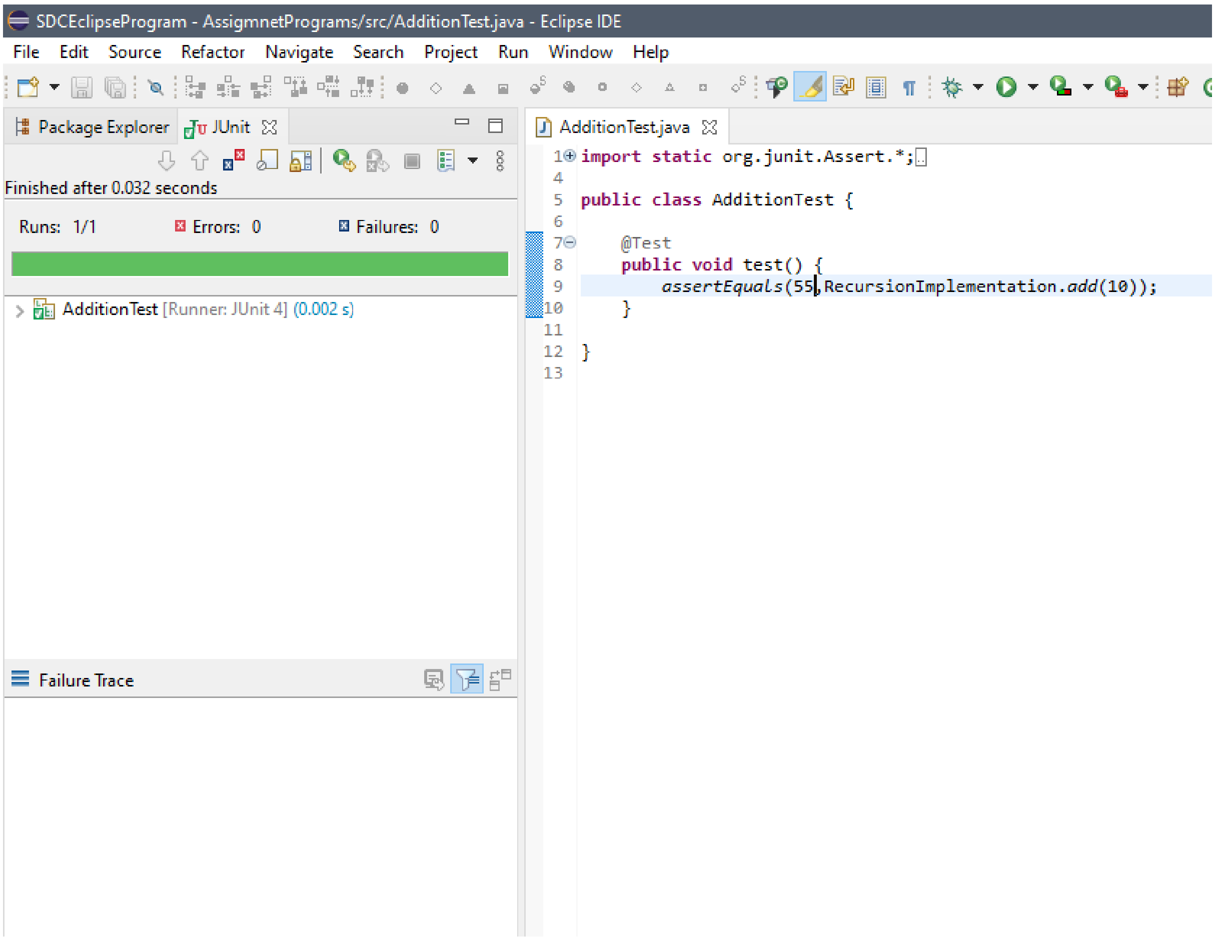

SDCEclipseProgram - AssigmnetPrograms/src/RecursionImplementation.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
RecursionImplementation.java
AdditionTest.java

```java

public class RecursionImplementation {

    public static long add(int number) {
        if(number == 0) {
            return 0;
        }
        else {
            return number + add(number - 1);

        }

    }



}

```


SDCEclipseProgram - AssigmnetPrograms/src/AdditionTest.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
RecursionImplementation.java
AdditionTest.java

```java
import static org.junit.Assert.*;

public class AdditionTest {

    @Test
    public void test() {
        assertEquals(55,RecursionImplementation.add(10));
    }

}

```

Software Construction and Development (SWE-312)
Assignment 2

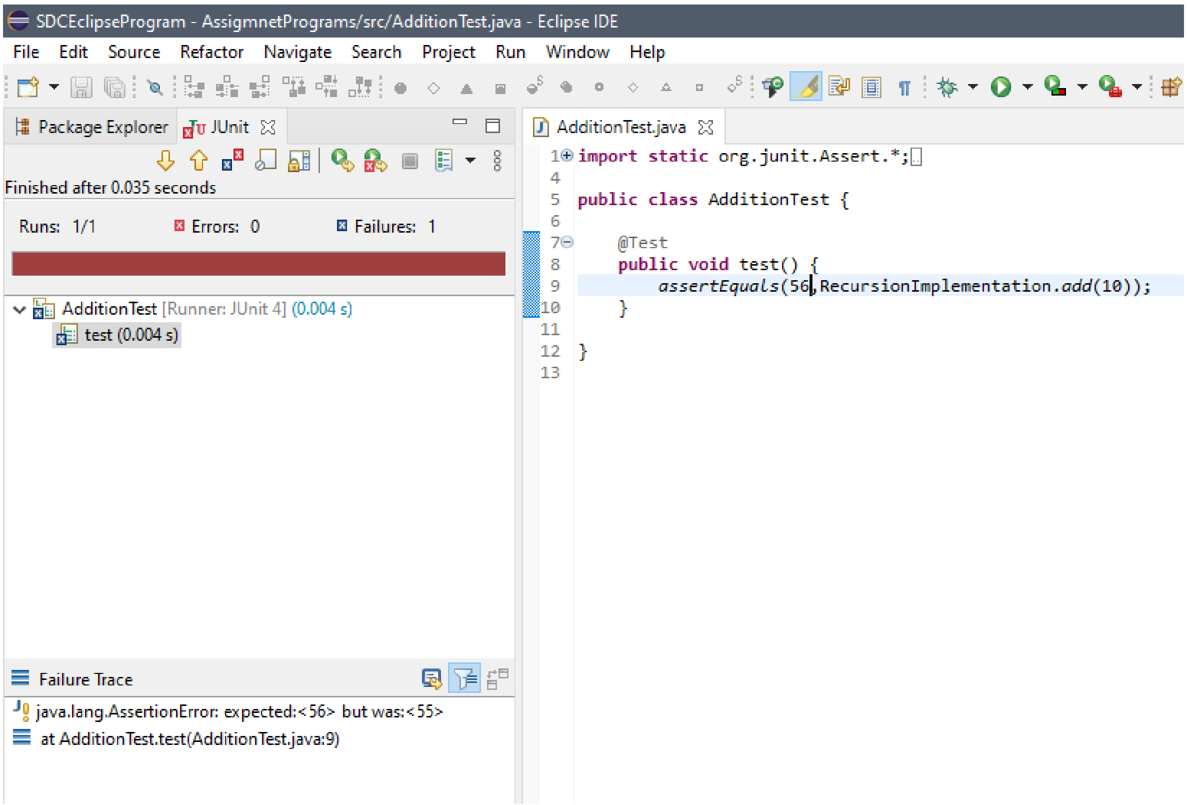

SDCEclipseProgram - AssigmnetPrograms/src/AdditionTest.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
JUnit
AdditionTest.java
Failure Trace

Finished after 0.032 seconds

Runs: 1/1 Errors: 0 Failures: 0

AdditionTest [Runner: JUnit 4] (0.002 s)

```java
import static org.junit.Assert.*;

public class AdditionTest {

    @Test
    public void test() {
        assertEquals(55,RecursionImplementation.add(10));
    }

}
```


SDCEclipseProgram - AssigmnetPrograms/src/AdditionTest.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
JUnit
AdditionTest.java
Failure Trace

Finished after 0.035 seconds

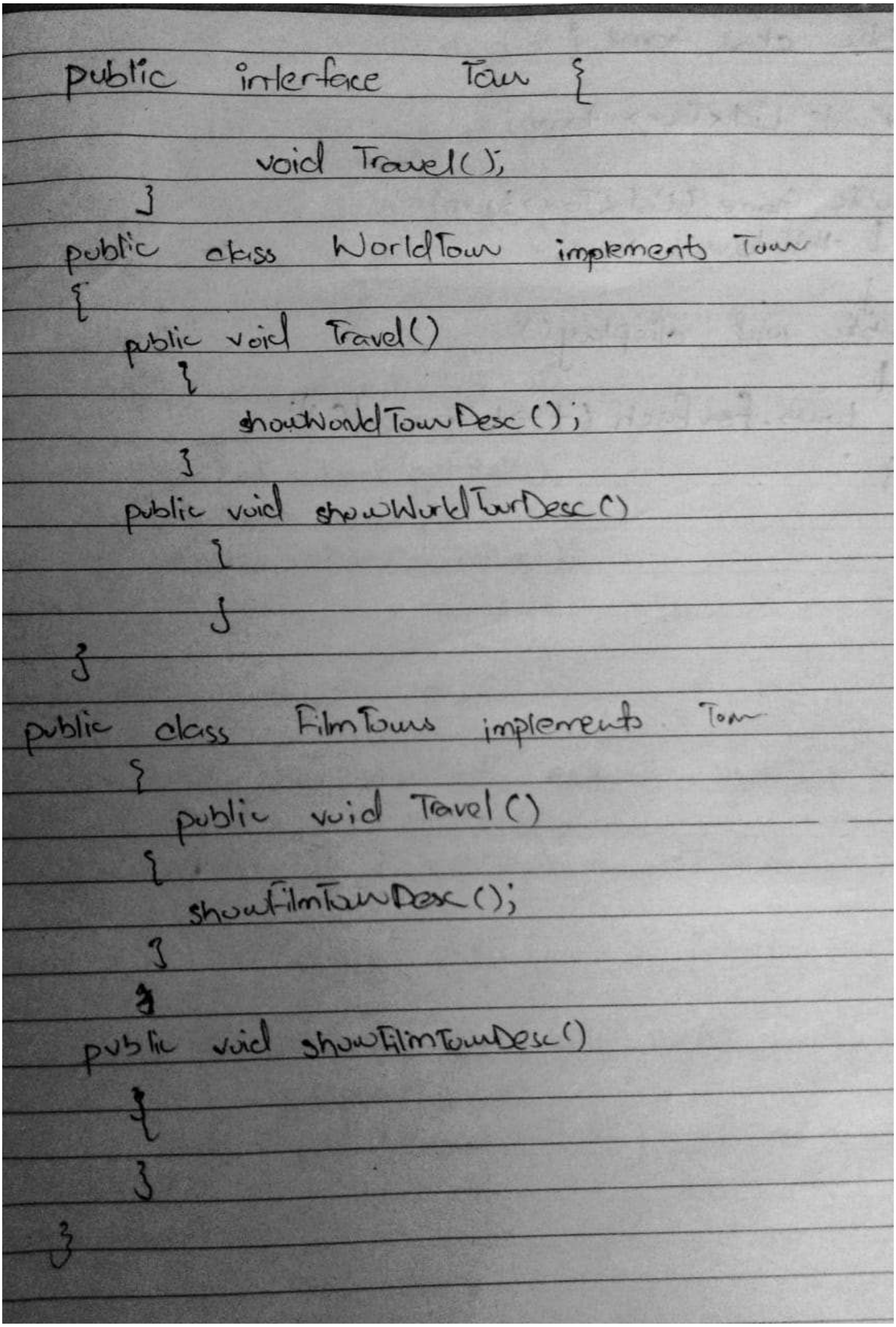
Runs: 1/1 Errors: 0 Failures: 1

AdditionTest [Runner: JUnit 4] (0.004 s)
test (0.004 s)

```java
import static org.junit.Assert.*;

public class AdditionTest {

    @Test
    public void test() {
        assertEquals(56,RecursionImplementation.add(10));
    }

}
```

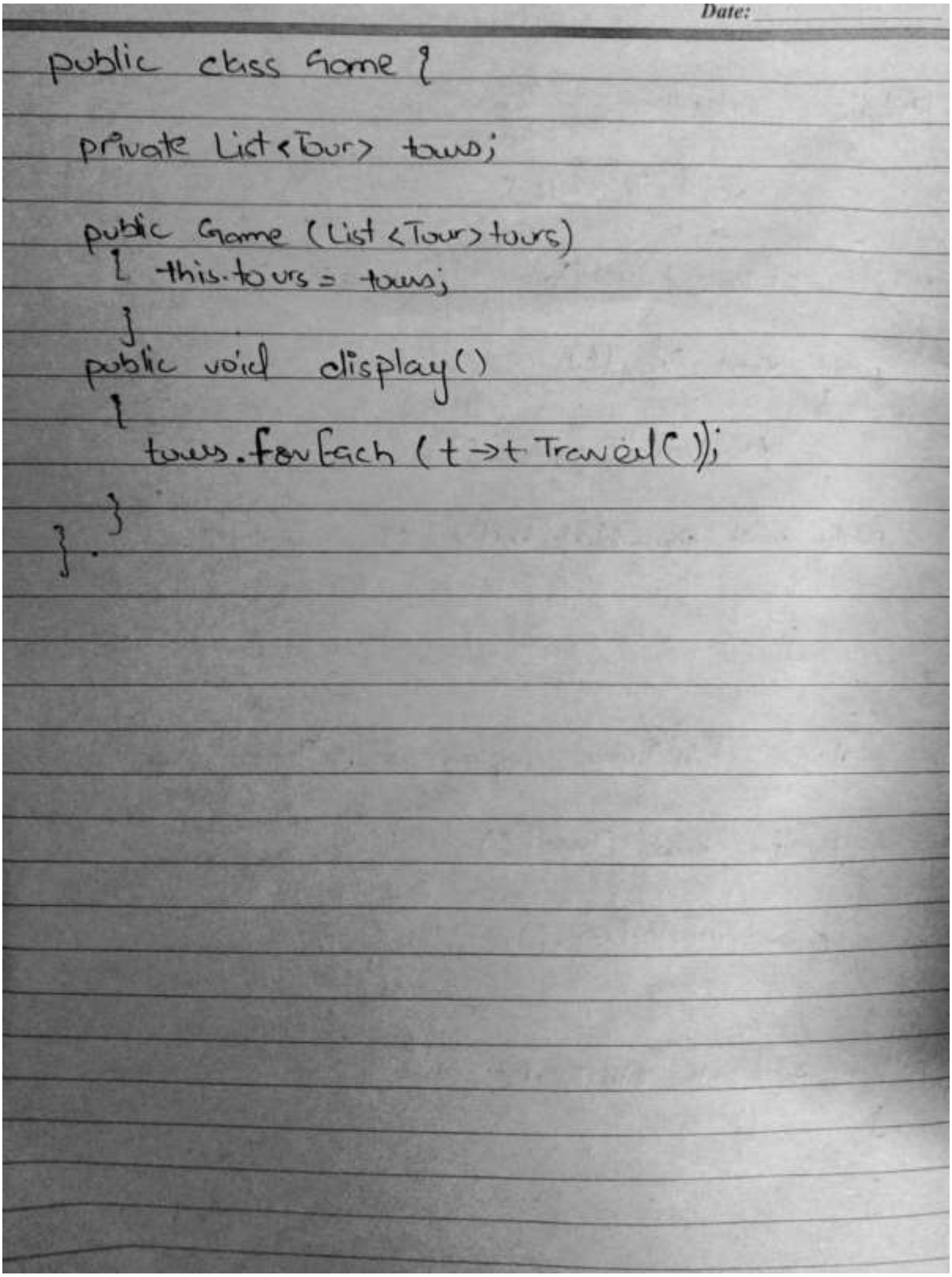
```
java.lang.AssertionError: expected:<56> but was:<55>
at AdditionTest.test(AdditionTest.java:9)
```

SOLUTION 2:-



```
public interface Tour {
            void Travel();
    }
public class WorldTour implements Tour
{
    public void Travel()
       {
            showWorldTourDesc();
       }
    public void showWorldTourDesc()
         {

         }
}
public class FilmTours implements Tour
     {
        public void Travel()
     {
        showFilmTourDesc();
     }
     }
  public void showFilmTourDesc()
     {

     }
}
```

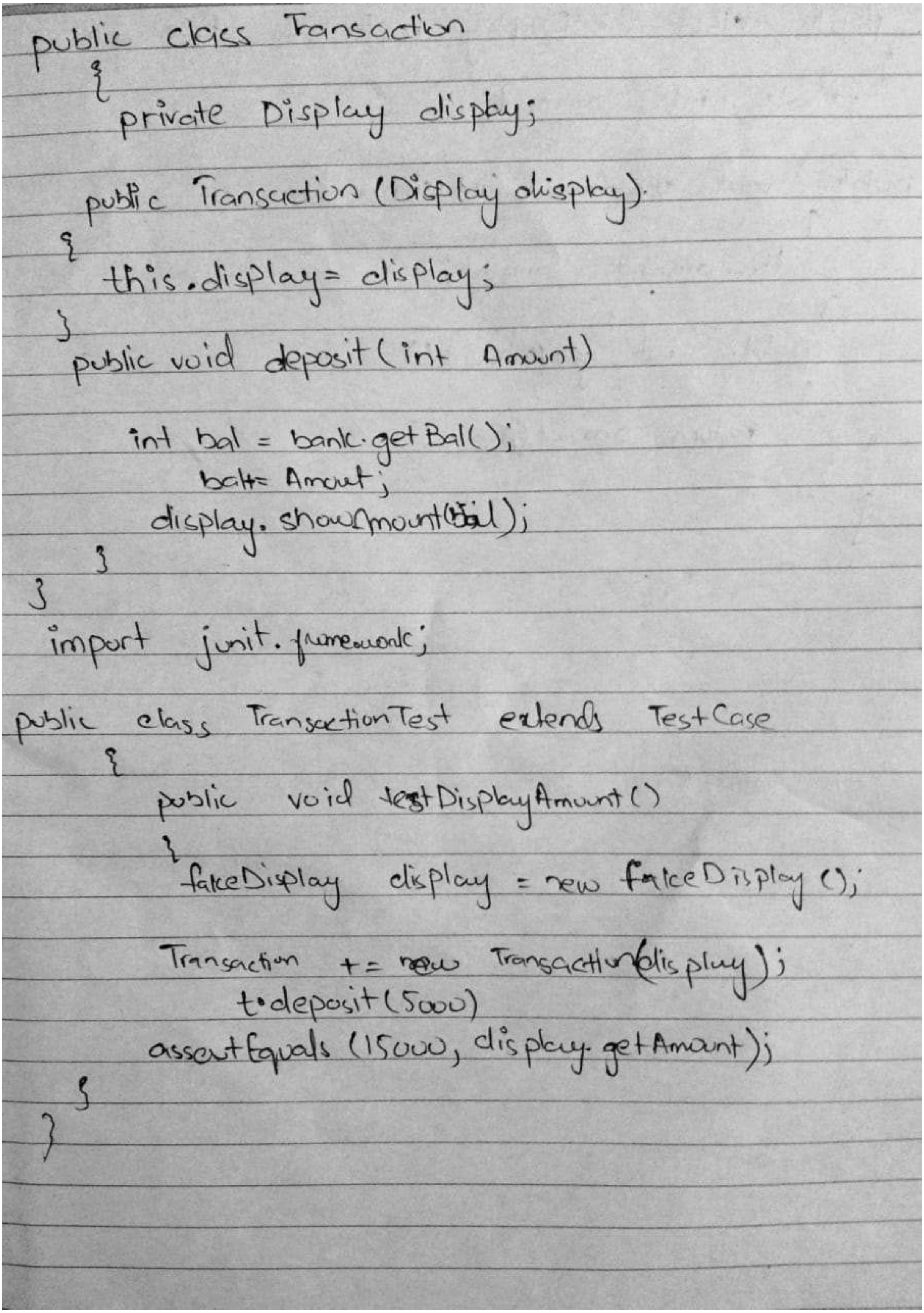
Software Construction and Development (SWE-312)
Assignment 2



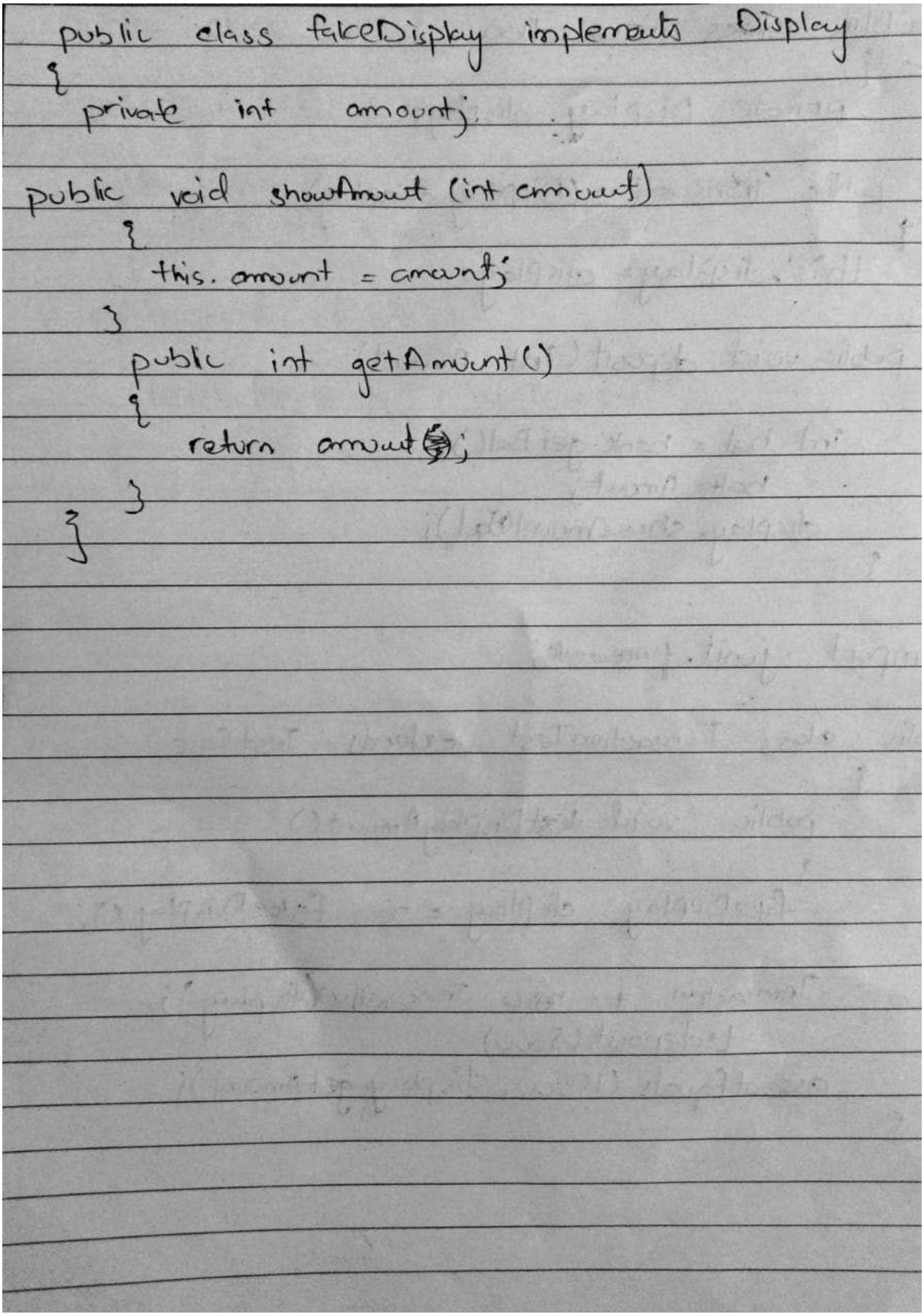
Date:

```
public class Game {

  private List<Tour> tours;

  public Game (List <Tour> tours)
   { this.tours = tours;
   }
  public void display()
   {
     tours.forEach (t ->t.Travel());
   }
}.
```

SOLUTION 3:-

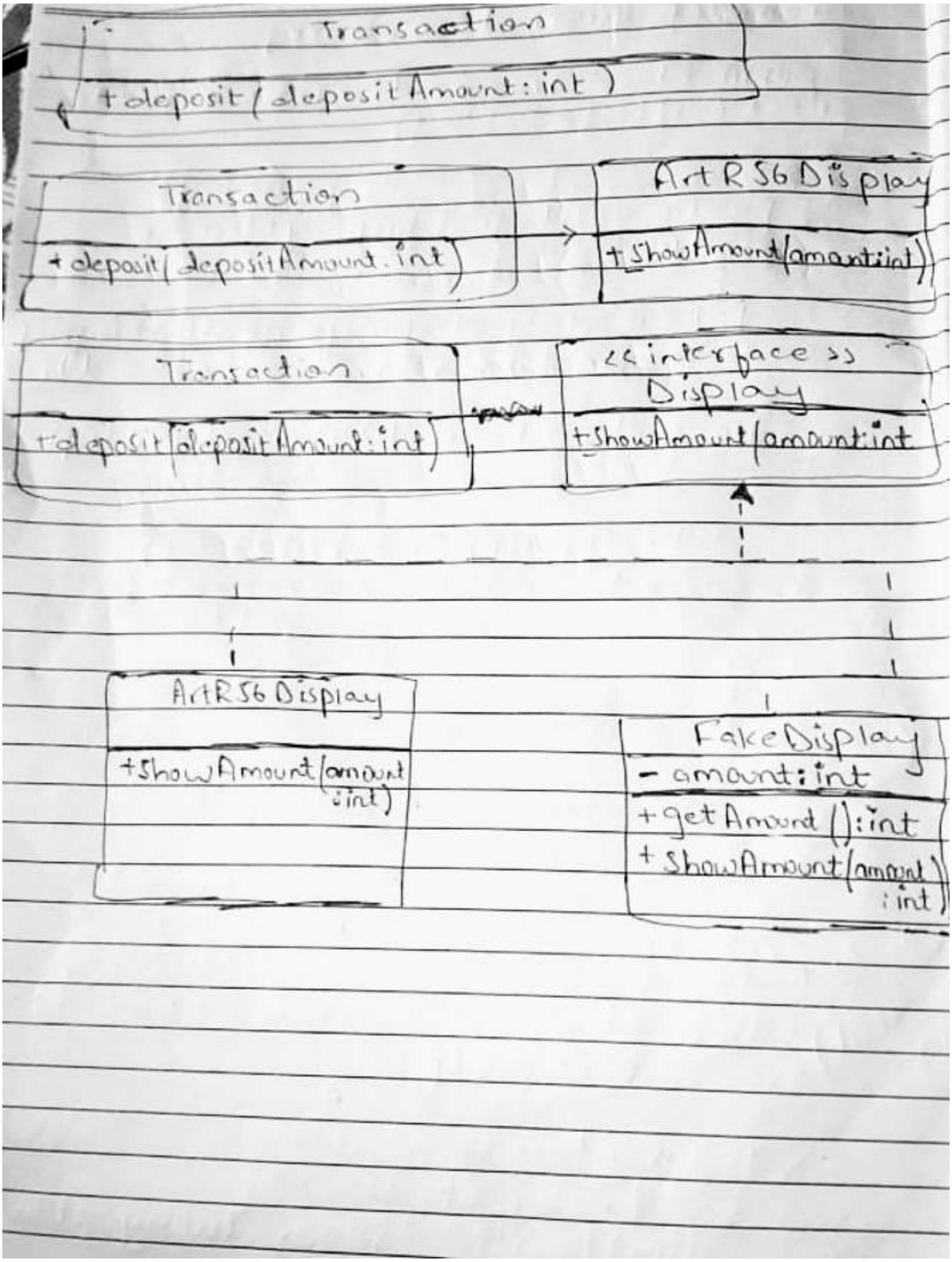


```
public class Transaction
   {
    private Display dispby;

  public Transaction (Display display).
  {
   this.display= display;
  }
  public void deposit(int Amount)

      int bal = bank.getBal();
          bal+= Amout;
      display. showAmount(Bal);
    }
}
 import junit.framework;

public class TransactionTest extends TestCase
     {
        public void testDisplayAmount()
        {
         fakeDisplay display = new fakeDisplay();

        Transaction t= new Transaction(display);
            t.deposit(5000)
        assertEquals(15000, display.getAmount);
    }
}
```



```
public class fakeDisplay implements Display
{
 private int amount;
public void showAmount (int amount)
    {
      this. amount = amount;
    }
    public int getAmount ()
    {
       return amount;
    }
}
```

Software Construction and Development (SWE-312)
Assignment 2


Transaction
+deposit(depositAmount: int)
Transaction
+deposit(depositAmount: int)
ArtR56Display
+ShowAmount(amount:int)
Transaction
+deposit(depositAmount:int)
<<interface>>
Display
+showAmount(amount:int
ArtR56Display
+ShowAmount(amount
:int)
FakeDisplay
- amount: int
+getAmount():int
+ShowAmount(amount
:int)