

## Unit Testing Report: Shubham Mehta, Harnoor Reen, Tyler Hong

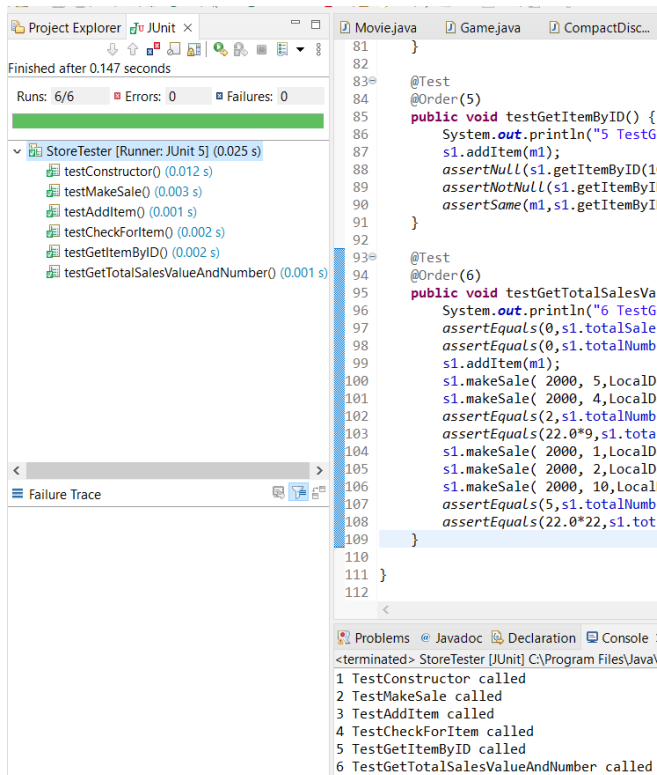
All testConstructors: All these tests check if the values have been initialized(not null), and that the values are initialized properly. They also check the copy constructors by checking the details and the ItemID. After using the copy constructor we check to see if the CurrentItemID is incremented or not.

All getPrice: For Item we check the boundary conditions by providing a date of before 1 year, exactly 1 year and 1 day after 1 year to see if the discounts are applied properly or not. For book we check that no matter what the date is the value is not affected. For Game we check that if the game is discontinued then the value is times 20 because it is vintage. For the other classes the test is the same as for Item.

All makeCopy: We get the object returned from the makeCopy() and check that the two objects are not same i.e. they do not refer to the same memory by using assertNotSame().

For the rest of the tests we use Boundary testing and decision testing to rigorously test all the units.

### Proof of StoreTester:



The screenshot displays the IDE interface during the execution of the StoreTester unit tests. The Project Explorer on the left shows the test results for StoreTester, including testConstructor(), testMakeSale(), testAddItem(), testCheckForItem(), testGetItemByID(), and testGetTotalSalesValueAndNumber(). The main editor shows the code for Movie.java, Game.java, and CompactDisc.java. The Console at the bottom shows the output of the tests, including the assertion messages and the test results.

```
81 }
82
83 @Test
84 @Order(5)
85 public void testGetItemByID() {
86     System.out.println("5 TestG
87     s1.addItem(m1);
88     assertNull(s1.getItemByID(1
89     assertNotNull(s1.getItemByI
90     assertEquals(m1,s1.getItemByI
91 }
92
93 @Test
94 @Order(6)
95 public void testGetTotalSalesVa
96     System.out.println("6 TestG
97     assertEquals(0,s1.totalSale
98     assertEquals(0,s1.totalNumb
99     s1.addItem(m1);
100     s1.makeSale( 2000, 5,LocalID
101     s1.makeSale( 2000, 4,LocalID
102     assertEquals(2,s1.totalNumb
103     assertEquals(22.0*9,s1.tota
104     s1.makeSale( 2000, 1,LocalID
105     s1.makeSale( 2000, 2,LocalID
106     s1.makeSale( 2000, 10,Local
107     assertEquals(5,s1.totalNumb
108     assertEquals(22.0*22,s1.tot
109 }
110 }
111 }
112
```

Failure Trace

<terminated> StoreTester [JUnit] C:\Program Files\Java\

- 1 TestConstructor called
- 2 TestMakeSale called
- 3 TestAddItem called
- 4 TestCheckForItem called
- 5 TestGetItemByID called
- 6 TestGetTotalSalesValueAndNumber called

## Proof of ItemTester:

Finished after 0.15 seconds

Runs: 6/6 Errors: 0 Failures: 0

ItemTester [Runner: JUnit 5] (0.028 s)

- testItemConstructors() (0.016 s)
- testGetPrice() (0.003 s)
- testSetDiscount() (0.002 s)
- testSell() (0.001 s)
- testAddStock() (0.000 s)
- testMakeCopy() (0.001 s)

```
64 @Test
65 @Order(2)
66 public void testGetP
67     System.out.print
68     assertAll("Get P
69     () -> as
70     () -> as
71     () -> as
72 }
73
74 @Test
75 @Order(3)
76 public void testSetD
77     System.out.print
78     assertFalse(i2.s
79     assertTrue(i2.se
80     assertTrue(i2.se
81 }
82
83 @Test
84 @Order(4)
85 public void testSell
86     System.out.print
87     assertEquals(0,i
88     assertEquals(0,i
89     assertEquals(0,i
90     assertEquals(0,i
91     assertNotEquals(
92     assertNotEquals(
93     assertNotEquals(
94 }
```

Failure Trace

Problems Javadoc Declaration

<terminated> ItemTester [JUnit] C:\Progr

- 1 testItemConstructors called
- 2 testGetPrice called
- 3 testSetDiscount called
- 4 testSell called
- 5 testAddStock called
- 6 testMakeCopy called

## Proof of BookTester:

Finished after 0.146 seconds

Runs: 4/4 Errors: 0 Failures: 0

BookTester [Runner: JUnit 5] (0.023 s)

- testConstructor() (0.017 s)
- testGetAuthors() (0.002 s)
- testGetPrice() (0.001 s)
- testMakeCopy() (0.000 s)

```
60 () -> a:
61 () -> a:
62 () -> a:
63 () -> a:
64 () -> a:
65 () -> a:
66 }
67
68 @Test
69 @Order(2)
70 public void testGet/
71     System.out.print
72     ArrayList<String
73     assertNotSame(ai
74 }
75
76 @Test
77 @Order(3)
78 public void testGetf
79     System.out.print
80     assertEquals(b1.
81 }
82
83
84 @Test
85 @Order(4)
86 public void testMak
87     System.out.print
88     assertNotSame(b.
89 }
90 }
```

Failure Trace

Problems Javadoc Declaration

<terminated> BookTester [JUnit] C:\Ph

- 1 TestConstructor called
- 2 testGetAuthors called
- 3 testGetPrice called
- 4 testMakeCopy called

## Proof of CompactDiscTester:

Project Explorer

JUnit x

Finished after 0.147 seconds

Runs: 3/3 Errors: 0 Failures: 0

CompactDiscTester [Runner: JUnit 5] (0.022 s)

- testConstructor() (0.017 s)
- testGetPrice() (0.002 s)
- testMakeCopy() (0.002 s)

Failure Trace

Movie.java

Game.java

CompactDiscTester.java

```
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66 @Test  
67 @Order(2)  
68 public void testGetPri  
69     System.out.println  
70     assertAll("Get Pri  
71     () -> asse  
72     () -> asse  
73     () -> asse  
74  
75  
76 @Test  
77 @Order(3)  
78 public void testMakeCc  
79     System.out.println  
80     assertNotSame(c1,c  
81  
82  
83  
84
```

Problems Javadoc Declaration

<terminated> CompactDiscTester [JUnit]

1 TestConstructor called  
2 testGetPrice called  
3 testMakeCopy called

## Proof of GameTester:

Project Explorer

JUnit x

Finished after 0.152 seconds

Runs: 3/3 Errors: 0 Failures: 0

GameTester [Runner: JUnit 5] (0.020 s)

- testConstructor() (0.013 s)
- testGetPrice() (0.002 s)
- testMakeCopy() (0.003 s)

Failure Trace

Movie.java

CompactDisc...

GameTester.java

```
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60 @Test  
61 @Order(2)  
62 public void testGetPrice  
63     System.out.println(  
64     assertAll("Get Price  
65     () -> assert  
66     () -> assert  
67  
68  
69 @Test  
70 @Order(3)  
71 public void testMakeCopy  
72     System.out.println(  
73     assertNotSame(g1,g1.  
74  
75  
76  
77
```

Problems Javadoc Declaration

<terminated> GameTester [JUnit] C:\Program

1 TestConstructor called  
2 testBookConstructors called  
3 testMakeCopy called

Note the specific order here

## Proof of MovieTester:

The screenshot displays an IDE window with the following components:

- Project Explorer:** Shows the project structure with 'MovieTester [Runner: JUnit 5] (0.024 s)' expanded, listing three tests: 'testConstructor() (0.019 s)', 'testGetPrice() (0.002 s)', and 'testMakeCopy() (0.001 s)'.
- JUnit Console:** Shows the execution progress: 'Finished after 0.167 seconds', 'Runs: 3/3', 'Errors: 0', and 'Failures: 0'.
- Code Editor:** Displays the 'Movie.java' file with the following code:

```
53         () -> a:
54         () -> a:
55         () -> a:
56         () -> a:
57         () -> a:
58         () -> a:
59         () -> a:
60         () -> a:
61         () -> a:
62         () -> a:
63         () -> a:
64     }
65
66     @Test
67     @Order(2)
68     public void testGetP
69         System.out.print
70         assertAll("Get f
71         () -> a:
72         () -> a:
73         () -> a:
74     }
75
76     @Test
77     @Order(3)
78     public void testMak
79         System.out.print
80         assertNotSame(m
81     }
82
83 }
84
```
- Failure Trace:** Shows the execution flow: '1 TestConstructor called', '2 testGetPrice called', and '3 testMakeCopy called'.