IT7320 – Software Development and Testing

Assignment 3

Test Driven Development

*Compiled October 2018*

*Aleisha Perry - 2115039*

For

*Chalinor Baliuag*

*Due Date: Week 12, Trimester 2, 2018*

# 

Contents

[Part 3 2](#_Toc527548067)

[Test Driven Development 2](#_Toc527548068)

[Step 1: 2](#_Toc527548069)

[Step 2: 2](#_Toc527548070)

[Step 3: 2](#_Toc527548071)

[Step 4: 3](#_Toc527548072)

[Step 5: 3](#_Toc527548073)

[Step 6: 3](#_Toc527548074)

[Step 7: 3](#_Toc527548075)

[Step 8: 4](#_Toc527548076)

[Step 9: 4](#_Toc527548077)

[Step 10: 4](#_Toc527548078)

[Step 11: 4](#_Toc527548079)

[Step 12: 5](#_Toc527548080)

[Conclusion 5](#_Toc527548081)

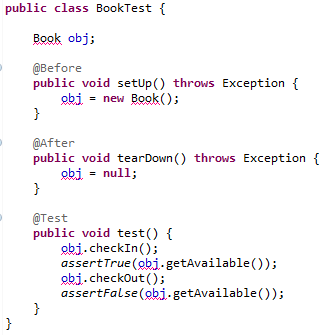
[Appendix A 6](#_Toc527548082)

[GitHub 6](#_Toc527548083)

# Part 3

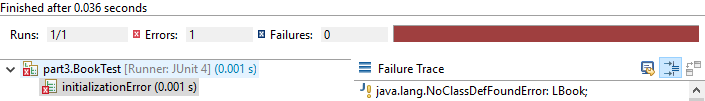
## Test Driven Development

### Step 1:



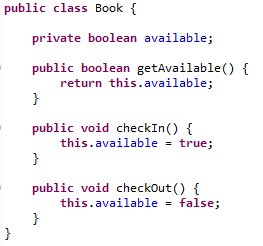
Step 1 >> the BookTest JUnit test is created. This code shows that checkOut(), checkIn(), and getAvaliable() methods will be required in the development in order to determine the status of a book. The class called Book must be created also.

### Step 2:



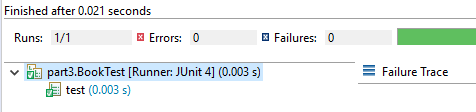
Step 2 >> the test was run and failed as there is no Book class yet.

### Step 3:



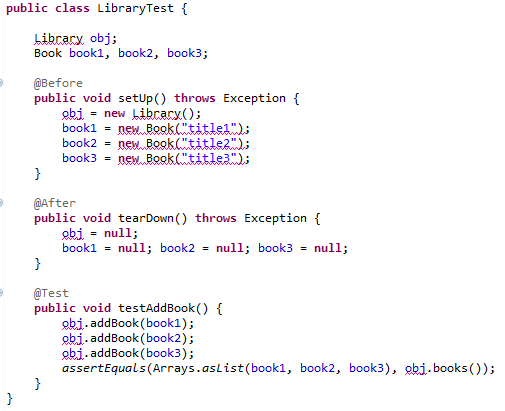
Step 3 >> the Book class has been created, with an accessor method getAvailable(), and the checkIn() and checkout() methods, changing the status of a book from true or false respectively, meaning that the book is available in the library.

### Step 4:



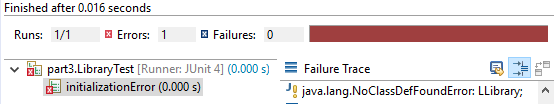
Step 4 >> the test now passes with the appropriate attributes and methods in place in the Book class.

### Step 5:



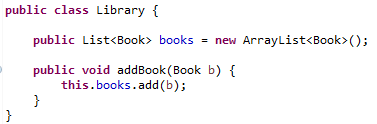
Step 5 >> the LibraryTest JUnit test is created. This code shows the addBook() method that will add a book to the list of books available in a library. The class Library must be created. In addition to the new Library class, a new attribute will have to be added to the Book class to store the title, and a constructor to store the title.

### Step 6:



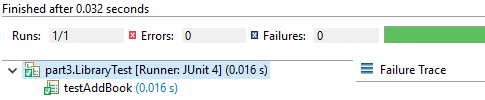
Step 6 >> the test was run and failed as there is no Library class yet.

### Step 7:



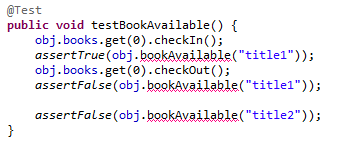
Step 7 >> the Library class has been created with the addBook() method, adding book objects to a list.

### Step 8:



Step 8 >> the test now passes with the appropriate attributes and methods in place in the Library class.

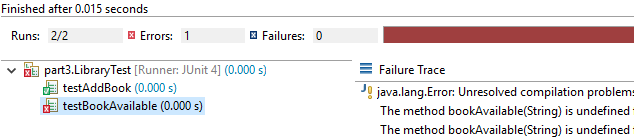
### Step 9:



Step 9 >> the new bookAvailable() method test was created the LibraryTest JUnit. This test will check a book in, changing its available value to true, then make sure it work. The same will be done for checking out, instead the value will be false.

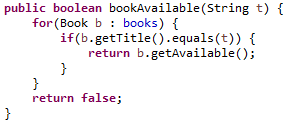
The last *assertFalse()* will make sure the default boolean value for the book availability is false.

### Step 10:



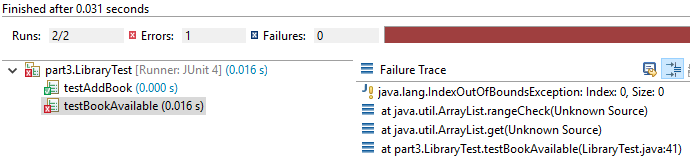
Step 10 >> the test was run and failed as there is no bookAvailable method in the Library class yet.

### Step 11:



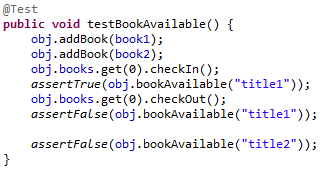
Step 11 >> the bookAvailable() method has been created in the Library class. It will search through the list of book objects and try to find the one associated with the title, and return the availability of that book. An accessor method was added to the Book class in order to retrieve the information stored in the Book title attribute.

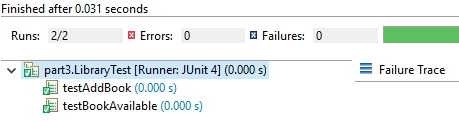
### Step 12:



Step 12 >> the test was run but failed, according to an IndexOutOfBounds exception.

Going back through the test I realised I didn’t add any Book objects to the Library list, so the list points to something that isn’t there.

 Two new lines of code added to the test (first two) and rerun.



>> The test now passes with the mistake fixed.

## Conclusion

Test driven development was a very new concept to me; never heard of TDD before this class, so had no experience how it worked. It was slow learning how to implement this process, but after a few trial and errors, where I’d create a test case then create the class and methods for the test case, and then realise I had done it all wrong and need *another* class with its own methods in order to make it work, I got a basic handle in what I was doing and actually started working on the assignment.

It seems to be a good method of programming if one was affluent in it. I wouldn’t mind learning more about TDD and practising it.

# Appendix A

## GitHub

GitHub was used for the duration of this assignment in order to submit assignment in stages (Parts 1 - 4).

**Profile - Aleisha**

Remote Repository main branch (owner) link: <https://github.com/kiwiFudge/IT7320_Assignment3_Part3.git>