

# Report 01

## 1. Reflection

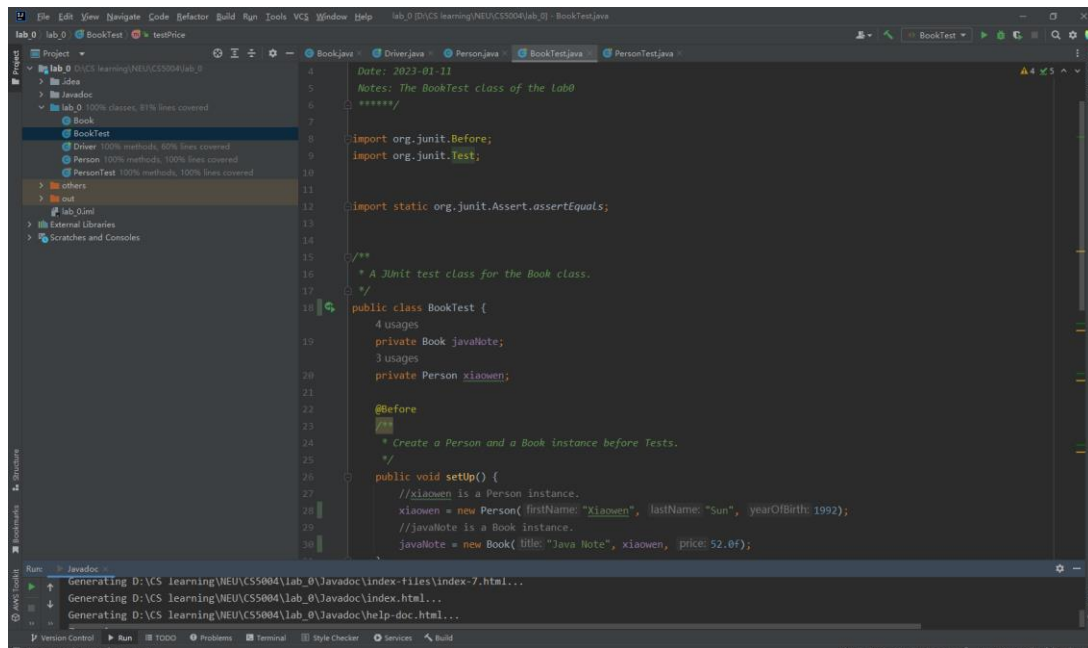
In this assessment, I learned to establish the Java coding environment, IDE installation, as well as running Java programs by IDE, command line and online compiler.

I learned to create class, instance and write a driver in a Java project. In terms of test, I learned to write simple unit tests for the program and utilizing a few modules in junit.

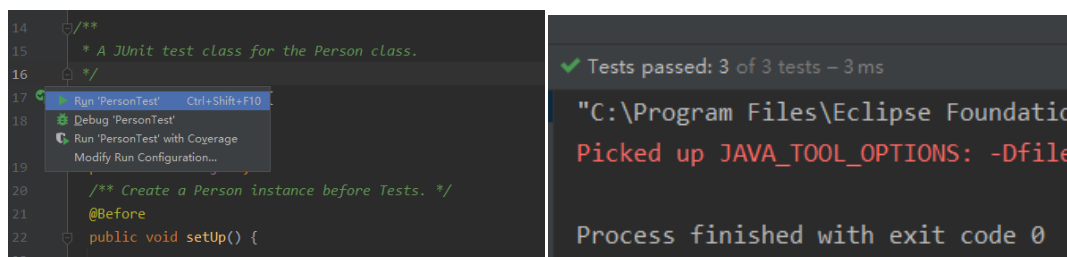
## 2. Required Task Elements

### Part 5: Running an Existing Project and Running JUnit Tests

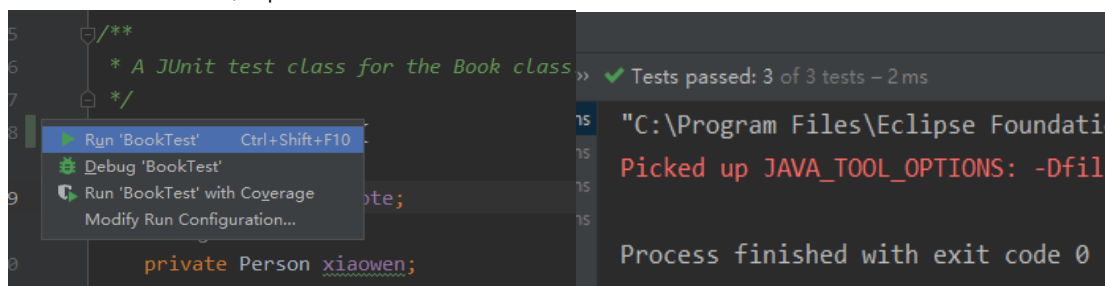
#### IntelliJ project screen shot



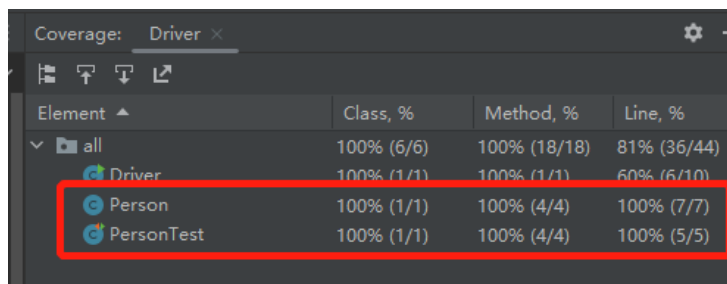
Test for person class, 3 passed



Test for book class, 3 passed



The coverage of booktest and persontest is 100%



Coverage: Driver x			
Element	Class, %	Method, %	Line, %
all	100% (6/6)	100% (18/18)	81% (36/44)
Driver	100% (1/1)	100% (1/1)	60% (6/10)
Person	100% (1/1)	100% (4/4)	100% (7/7)
PersonTest	100% (1/1)	100% (4/4)	100% (5/5)

## Part 6: Documenting Your Code

Top comment block

```
/**
 * Name: Xiaowen Sun
 * Assignment: Lab_0
 * Date: 2023-01-11
 * Notes: The BookTest class of the Lab0
 */
```

Class comment block

```
/**
 * A JUnit test class for the Book class.
 */
public class BookTest {
    4 usages
    private Book javaNote;
    3 usages
    private Person xiaowen;
```

Function comment block

```
@Test
/** Test getPrice() method and price attribute of Book class.
 * getPrice() method returns the price attribute, and it should be a float,
 * price for the Book instance javaNote should be 52.0f. */
public void testPrice() {
    //there should be three arguments for the assertEquals method for float.
    assertEquals( expected: 52.0f, javaNote.getPrice(), delta: 0.0001);
}
```

Generated Java Doc

### All Classes

Class Summary	
Class	Description
Book	This class represents a book.
BookTest	A JUnit test class for the Book class.
Driver	This class is a JUnit test driver to run all the tests of the project lab_o.
Person	This class represents a person The person has a first name, last name and an year of birth.
PersonTest	A JUnit test class for the Person class.

### Hierarchy For All Packages

#### Class Hierarchy

- java.lang.Object<sup>12</sup>
  - Book
  - BookTest
  - Driver
  - Person
  - PersonTest

### Class Book

```

java.lang.Object
Book

public class Book
extends Object

This class represents a book. A book has a title, an author and a price.

```

#### Constructor Summary

Constructors	Description
<code>Book(String title, Person author, float price)</code>	Construct a Book object that has the provided title, author and price.

#### Method Summary

All Methods	Instance Methods	Concrete Methods	
Modifier and Type	Method		Description
Person	<code>getAuthor()</code>		Return the author of this object.
float	<code>getPrice()</code>		Return the price of this book.
String	<code>getTitle()</code>		Return the title of this book.

##### Methods inherited from class java.lang.Object

```

clone, equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

```

#### Constructor Details

##### Book

```

public Book(String title,
            Person author,
            float price)

Construct a Book object that has the provided title, author and price.

Parameters:
title - the title to be given to this book
author - the author to be given to this book
price - the price to be assigned to this book

```

#### Method Details

##### getTitle

```

public String getTitle()

Return the title of this book.

Returns:
the title of this book

```

3. Extensions

I tried different methods to give more interesting properties to my drawings, extensions are as follows.

- Provide screenshots of both Eclipse and IntelliJ
- High quality submission
- Created a driver for this application.
- Submitted code as a GitHub link instead

Grading Statement

Both two python files are functioning well without significant error, scaling is added in both python files, the report is also provided.

I tried to create a self-designed logo and a complicated rose pattern, and let my submission more creative, also I learned different advanced methods in turtle, which were not taught in class, and applied them in my assignment.

For the above reasons, I think this work deserves a full mark.

	Possible	Given
Main Objectives		
Simple functionality with shapeOne.py ✓	6	0
Simple functionality with shapeTwo.py ✓	6	0
Scaling added to shapeTwo.py ✓	6	0
Misc		
Report ✓	4	0
Code Quality (correct indentation, comment blocks, variable naming, etc) ✓	4	0
Not included in total possible:		
Extensions (Not calculated without report) ✓	4	0
Creative or went above and beyond ✓	4	0
Code does not compile	-30	0
Late penalty	-6	0
Not implemented as requested	-30	0
TOTAL POINTS POSSIBLE out of 30	26	0