

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет радіоелектроніки, комп'ютерних систем та інфокомунікацій

Кафедра комп'ютерних систем, мереж і кібербезпеки

Лабораторна робота

з Кросплатформенні технології

(назва дисципліни)

на тему: «Інструментальні засоби розроблення для платформонезалежного
програмування»

Виконав: студент 4 курсу групи № 5456
напряму підготовки (спеціальності)
123 – комп'ютерна інженерія

(шифр і назва напряму підготовки (спеціальності))

Поліщук А.О.

(прізвище й ініціали студента)

Прийняв: асистент каф.503

Годованюк П.А.

(посада, науковий ступінь, прізвище й ініціали)

Національна шкала: _____

Кількість балів: _____

Оцінка: ECTS _____

1. Постановка задачи

1.1 Ознакомиться с интегрированной средой разработки IntelliJ IDEA

1.2 Ознакомиться с интегрированной средой разработки Eclipse.

2. Создание простого приложения с помощью IntelliJ IDEA

Сначала был создан проект по шаблону «Console application», дальше добавил в метод main вывод простого текста на терминал и построил проект.

Скриншот дерева файлов проекту и коду приведен на рисунке 1.

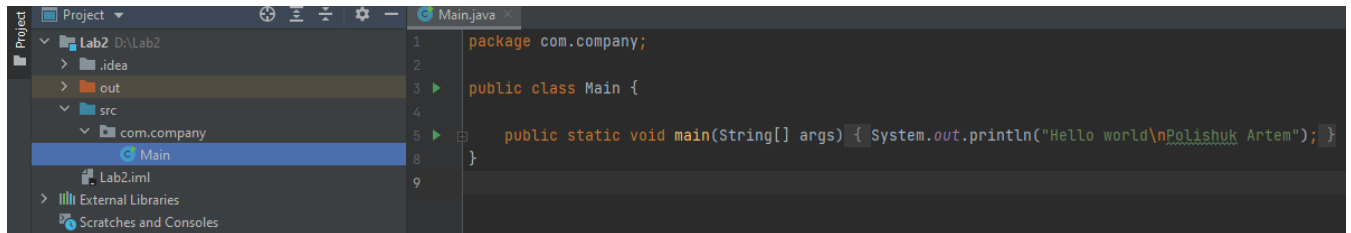


Рисунок 1 – Дерево файлов и код проекта

Скриншот работы проекта по выводу результата в терминал изображено на рисунке 2 .

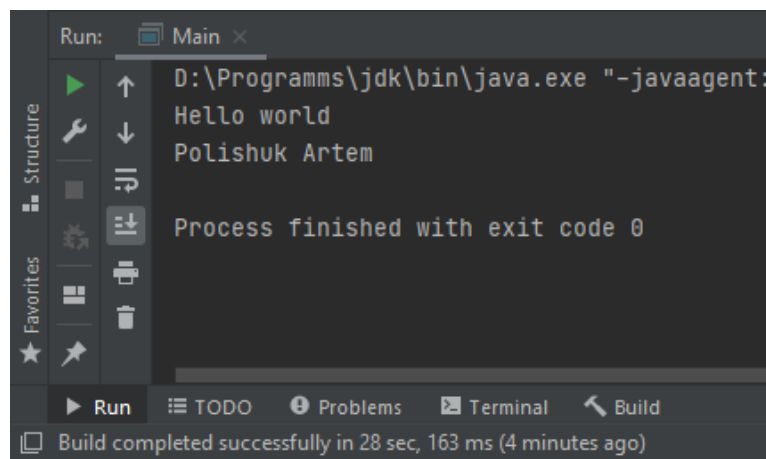
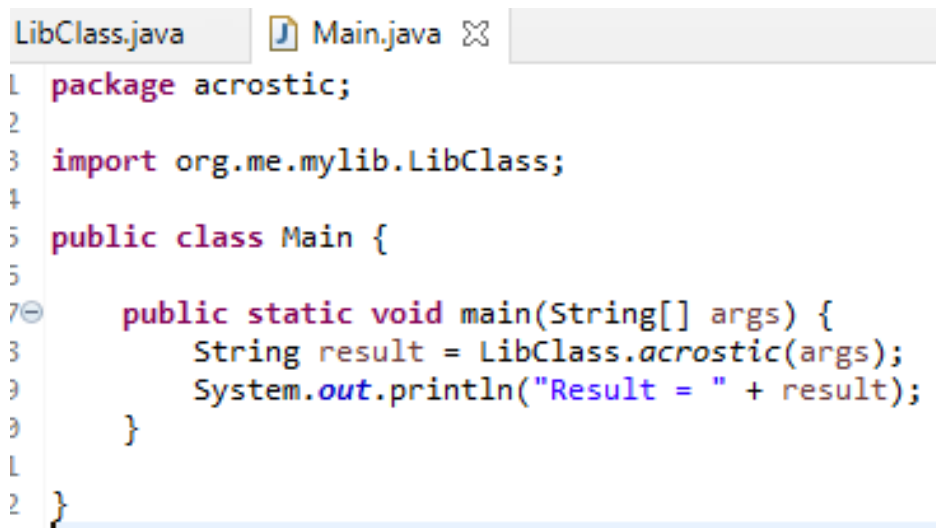


Рисунок 2 – Вывод результата в терминал

3. Основы разработки приложения на Java в IDE Eclipse

Изначально создали 2 проекта, добавили библиотеку в проект основной программы, а далее написали в главной программе простой код.

Код приведен на рисунке 3



```

LibClass.java  Main.java
1 package acrostic;
2
3 import org.me.mylib.LibClass;
4
5 public class Main {
6
7     public static void main(String[] args) {
8         String result = LibClass.acrostic(args);
9         System.out.println("Result = " + result);
10    }
11
12 }

```

Рисунок 3 – Код метода main

Теперь добавим входным параметром в программу текст, чтобы в результате выполнения программы была получено строчка – Hello(изображено на рисунке 4).

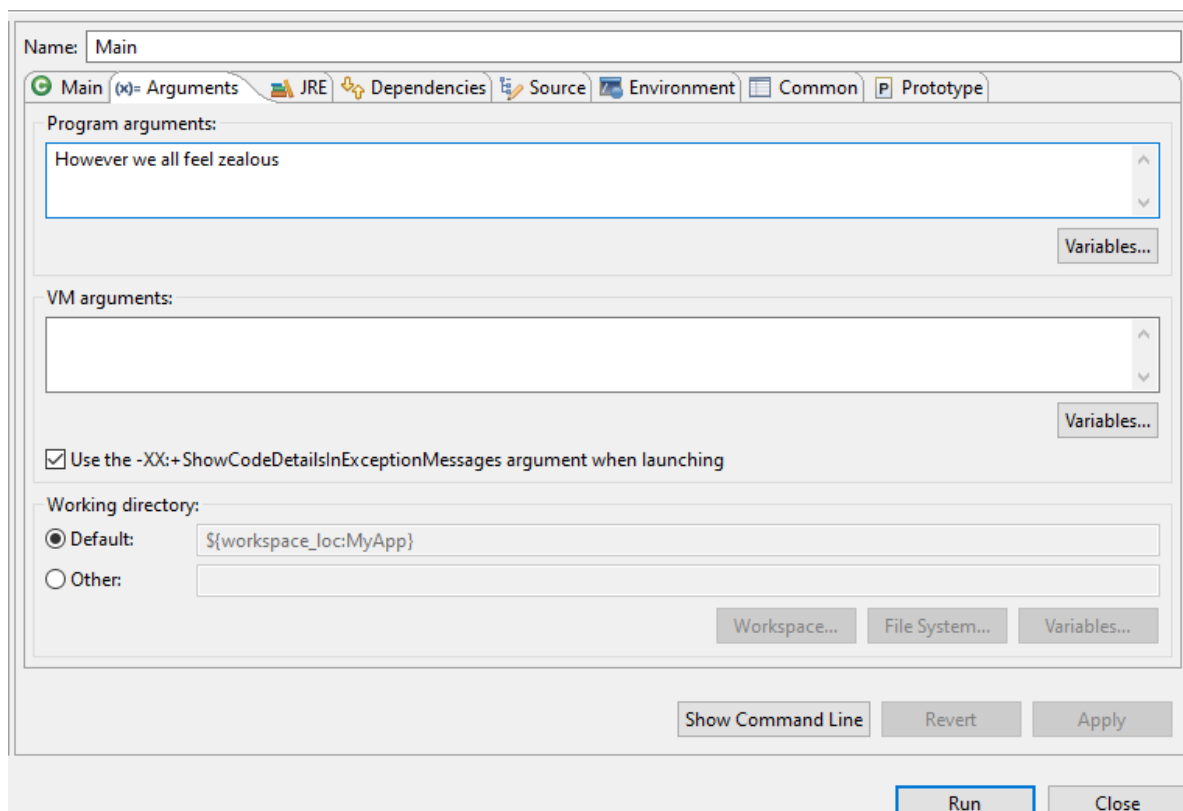


Рисунок 4 – Установка входных аргументов для приложения

Далее добавим комментарии к коду и сгенерируем документацию, пример генерации приведен на рисунке 5.

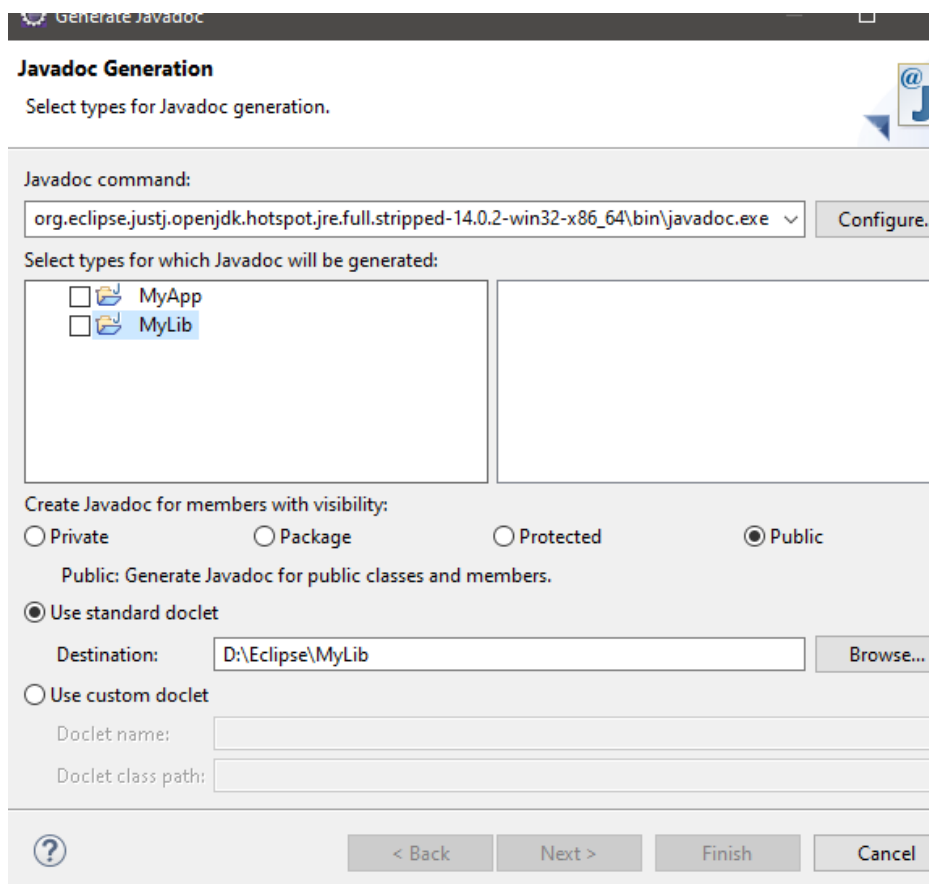


Рисунок 5 – Окно генерации javadoc

Теперь создадим тест для метода библиотеки, и выполним его (Результат на рисунке 6).

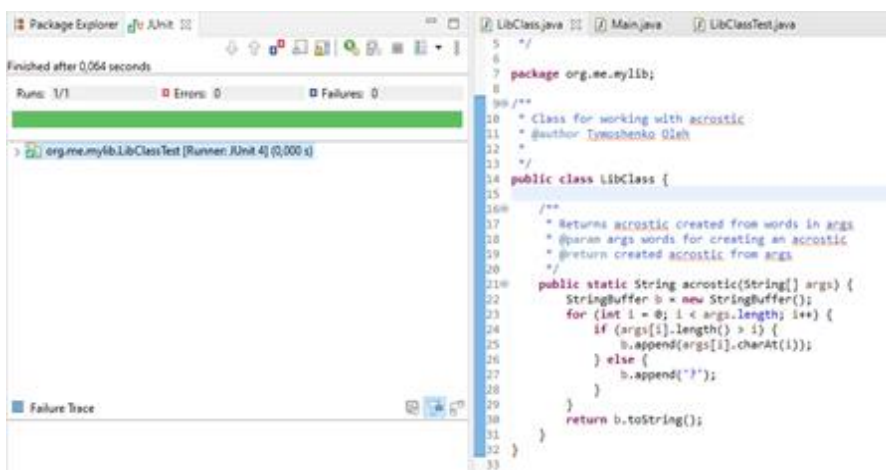


Рисунок 6 – Скриншот теста и результат его выполнения

Теперь попробуем добавить breakpoint в библиотеку, и на этой точке остановки посмотреть значения всех переменных, результат этих действий приведен на рисунке 7.

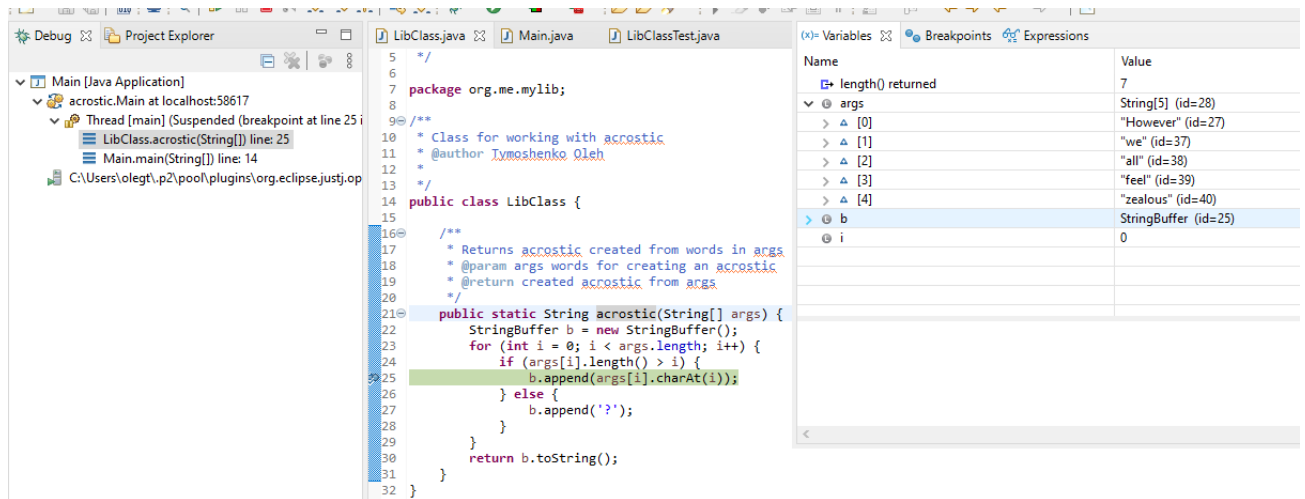


Рисунок 7 – Отладка приложения с изображением значений всех переменных

Структура проекта после всех проведенных действий на рисунке 8.

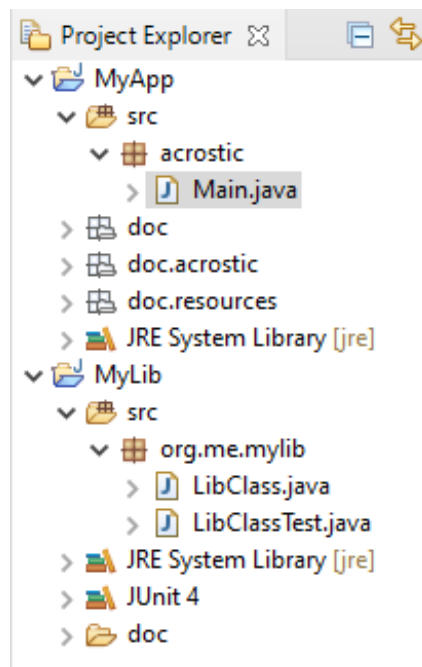


Рисунок 8 – Структура проекта после всех действий

4. Основы розроблення Java застосунків в Eclipse IDE

Package Explorer с добавленным архивом Junit и развернутым файлом TestCase.java представленный на рисунке 9.

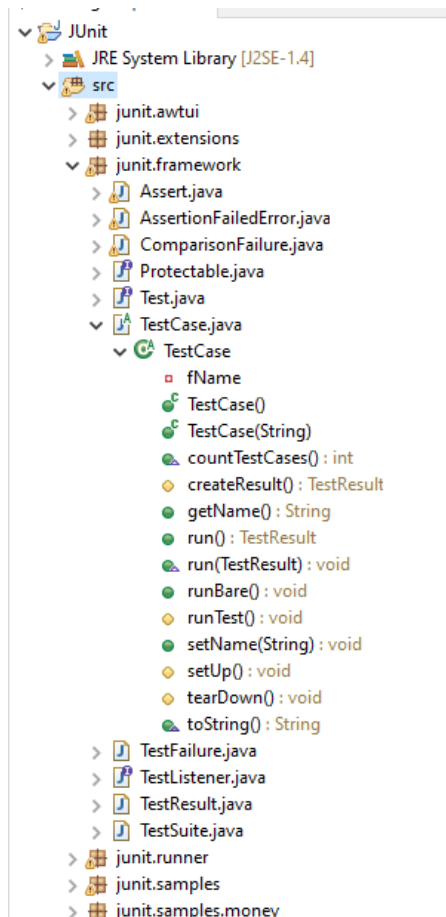


Рисунок 9 – Package Explorer с добавленным архивом Junit и развернутым файлом TestCase.java

Согласно заданию, далее был добавлен новый метод VectorTest.java. Скриншот метода в коде и новой структуры проекта на рисунке 10

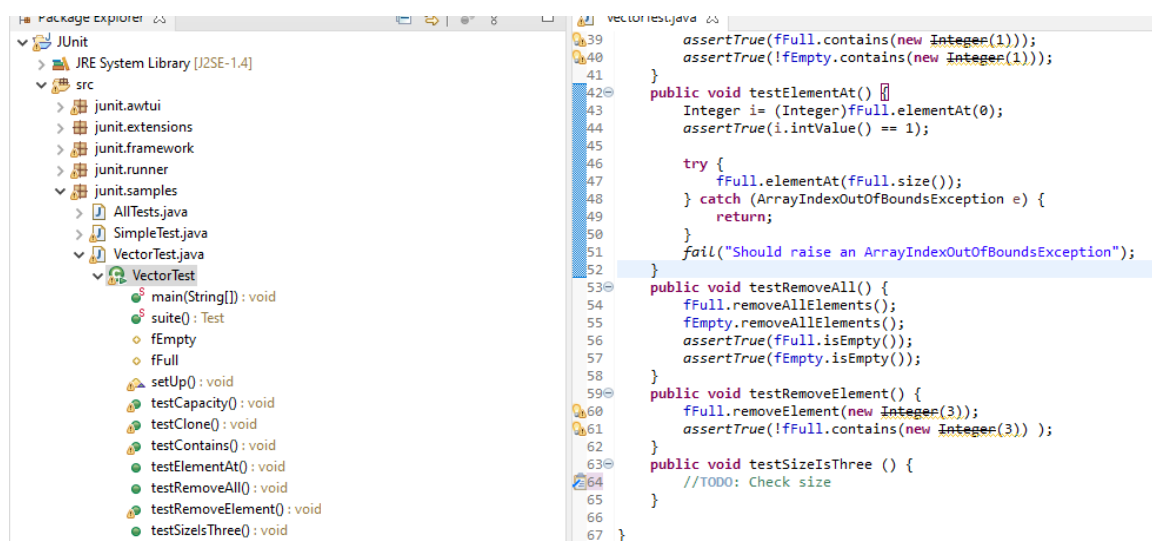


Рисунок 9 – Скриншот метода в коде и новой структуры проекта

Теперь внесем в код ошибку, удалив символ ';' в файле TestCase.java и сохраним файл. В результате IDE подсветит ошибки в нескольких местах. Скриншот ошибок на рисунке 10.

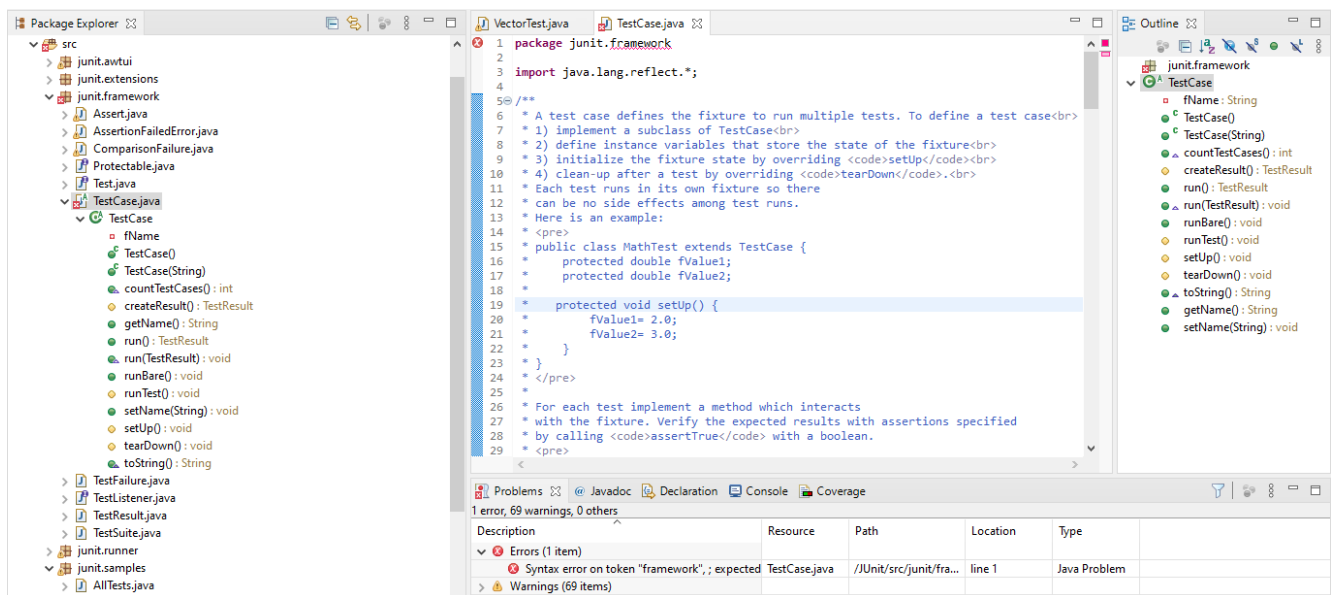


Рисунок 10 – Скриншот ошибок в нескольких местах

Далее был создан новый метод testValues, и в нем был написан код с помощью шаблонов, результат на рисунке 11.

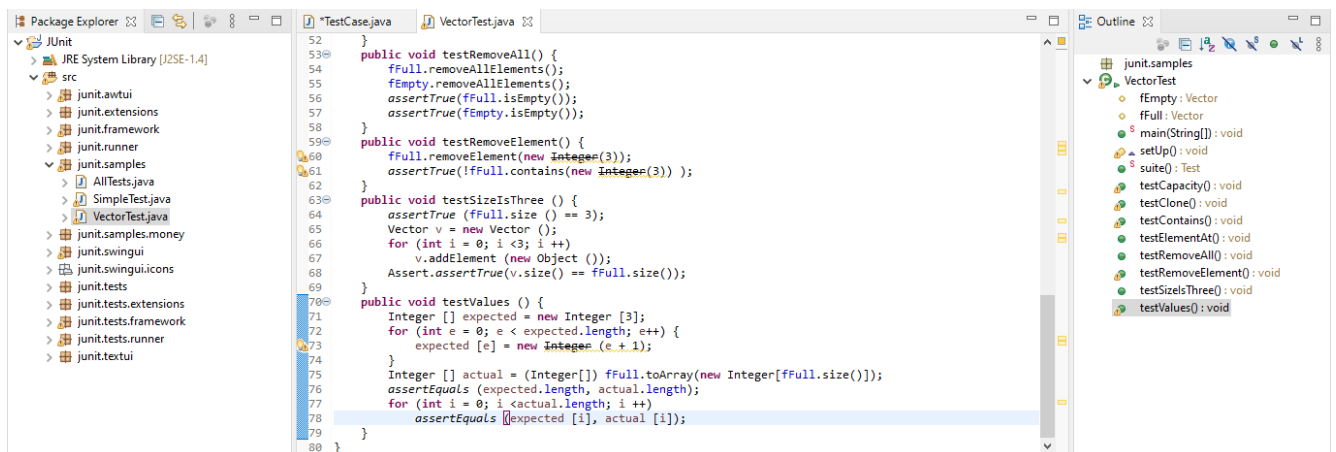


Рисунок 11 – Скриншот созданного с помощью шаблонов метода и структуры проекта

Далее создали новый пакет, класс, добавили методы, свойства, геттер и сеттер.

Скриншот кода и структуры проекта в результате приведен на рисунке 12.

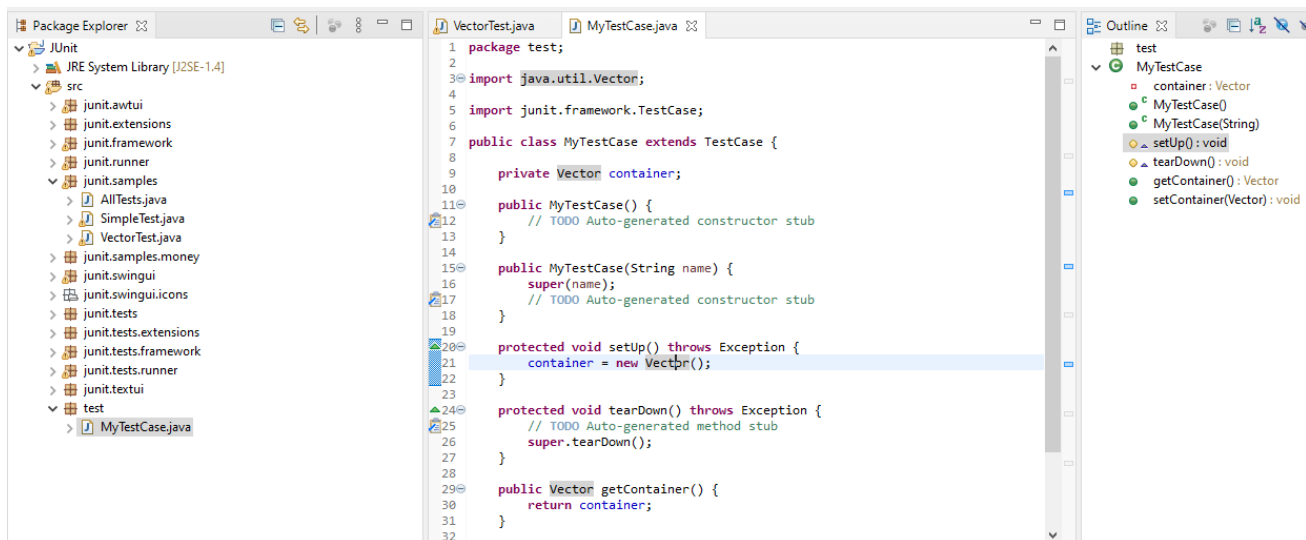


Рисунок 12 – Скриншот кода и структуры проекта

5. Посилання на репозитарій із проектом у системі керування версіями.

<https://github.com/PolishukArtem/Cross/tree/main/LAB2>

Висновки:

В результате выполнения лабораторной работы я ознакомился с 2-мя основными IDE для разработки на Java, это IntelliJ IDEA и Eclipse. Я создал простое приложение, состоящий из основного приложения, библиотеки и юнит-теста метода библиотеки. Также я поработал с архивом Junit в Eclipse IDE, научился использовать основные возможности этой среды разработки для более быстрого и удобного написания кода. После недолгого ознакомления с этими 2-мя среде разработки, для себя я решил, что IntelliJ IDEA больше понравилась мне через большую простоту и вид дизайна ide. В плане функционала, как минимум для начального уровня, обе среды разработки дают хорошие возможности для удобного написания кода.