

## **BÁO CÁO LAB 3 – KIỂM THỬ NÂNG CAO (JUnit 5)**

Môn học: Kiểm thử nâng cao

Sinh viên: Bùi Tiến Sỹ

MSSV: PS32117

### **I. MỤC TIÊU LAB 3**

- Hiểu và áp dụng vòng đời kiểm thử trong JUnit 5.
- Thực hiện kiểm thử ngoại lệ trong chương trình Java.
- Áp dụng Parameterized Test để kiểm thử với nhiều bộ dữ liệu.

### **II. BÀI 1 – VÒNG ĐỜI KIỂM THỬ JUNIT**

Trong bài 1, em đã sử dụng các annotation của JUnit 5 để quản lý vòng đời kiểm thử bao gồm @BeforeAll, @BeforeEach, @AfterEach và @AfterAll.

Các annotation được sử dụng:

- @BeforeAll: Chạy một lần trước tất cả các test.
- @BeforeEach: Chạy trước mỗi test case.
- @AfterEach: Chạy sau mỗi test case.
- @AfterAll: Chạy một lần sau tất cả các test.

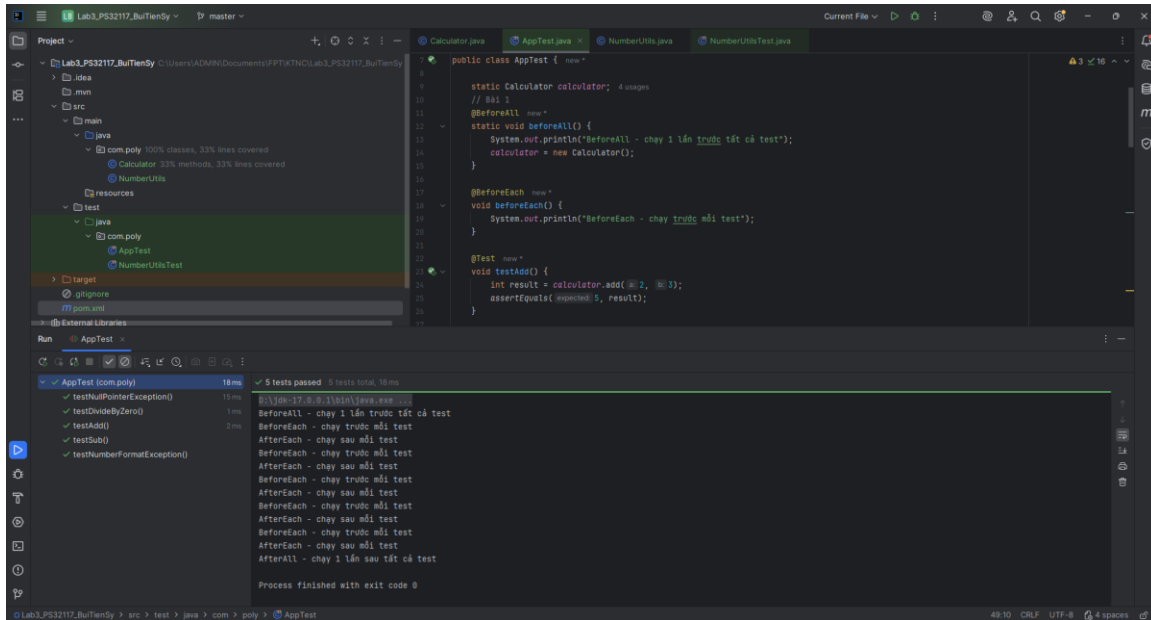
### **III. BÀI 2 – KIỂM THỬ NGOẠI LỆ**

Trong bài 2, em sử dụng phương thức `assertThrows()` của JUnit 5 để kiểm thử các ngoại lệ phổ biến có thể xảy ra trong chương trình.

Các ngoại lệ đã kiểm thử:

- `ArithmeticException`: Xảy ra khi chia cho 0.
- `NumberFormatException`: Xảy ra khi chuyển đổi chuỗi không hợp lệ sang số.
- `NullPointerException`: Xảy ra khi truy cập đối tượng null.

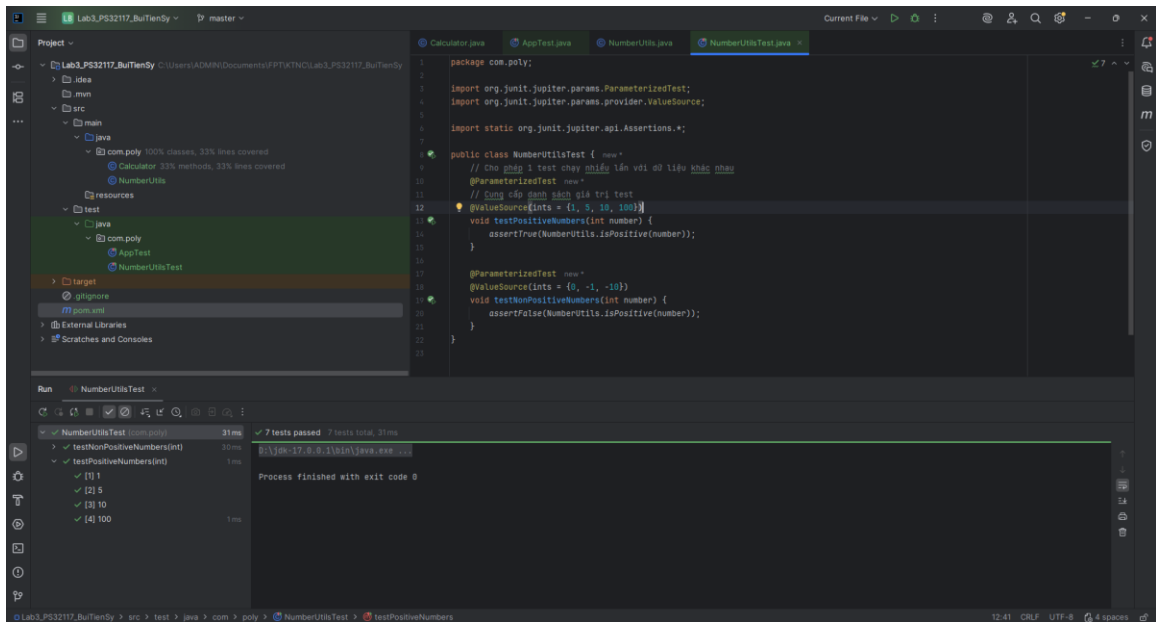
## Bài 1 và 2 gộp chung



### IV. BÀI 3 – PARAMETERIZED TEST

Trong bài 3, em đã sử dụng Parameterized Test của JUnit 5 để kiểm thử chức năng kiểm tra số nguyên dương với nhiều bộ dữ liệu khác nhau.

Việc sử dụng @ParameterizedTest kết hợp với @ValueSource giúp giảm việc viết code lặp và tăng hiệu quả kiểm thử.



## V. KẾT LUẬN

Thông qua Lab 3, em đã nắm được cách sử dụng JUnit 5 để thực hiện kiểm thử nâng cao, bao gồm vòng đời kiểm thử, kiểm thử ngoại lệ và kiểm thử có tham số. Lab giúp em hiểu rõ hơn về tầm quan trọng của kiểm thử trong phát triển phần mềm.