**BÀI THỰC HÀNH SỐ 04 – LẬP TRÌNH VÀ KIỂM THỬ ĐƠN VỊ**

**Tuần 11, ngày 07/12/2021**

Họ và tên: Trần Văn Tuyền

MSSV: 20184012

Mã lớp thực hành: 710808

Mục lục

[Link đã nộp bài trên Github 1](#_Toc89800134)

[1. Thực hành thiết kế unit tests theo hướng TDD 1](#_Toc89800135)

[1.1. Đặc tả yêu cầu 1](#_Toc89800136)

[1.2. Tạo UnitTest bằng Eclipse 2](#_Toc89800137)

[1.3. Thực hành xây dựng các phương thức theo TDD 2](#_Toc89800138)

[2. Lập trình 4](#_Toc89800139)

[2.1. Bắt đầu với code sample 4](#_Toc89800140)

[2.2. Làm quen với Javadoc 4](#_Toc89800141)

[2.3. Thực hành tạo Javadoc với Eclipse 4](#_Toc89800142)

[2.4. Làm quen với refactoring 6](#_Toc89800143)

[3. Bài tập về nhà 7](#_Toc89800144)

# Link đã nộp bài trên Github

https://github.com/tuyentran2222/710808\_AIMS\_20184012

# 1. Thực hành thiết kế unit tests theo hướng TDD

## 1.1. Đặc tả yêu cầu

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên tham số** | **Yêu cầu** |
| 1 | name | chỉ bao gồm chữ cái, không chứa ký tự đặc biệt, không được phép null |
| 2 | phone | chỉ bao gồm chữ số, độ dài 10 ký tự và bắt đầu là số 0 |
| 3 | address | không được phép null, không chứa ký tự đặc biệt |

## 1.2. Tạo UnitTest bằng Eclipse

Ảnh có chứa văn bản

Mô tả được tạo tự động

## 1.3. Thực hành xây dựng các phương thức theo TDD

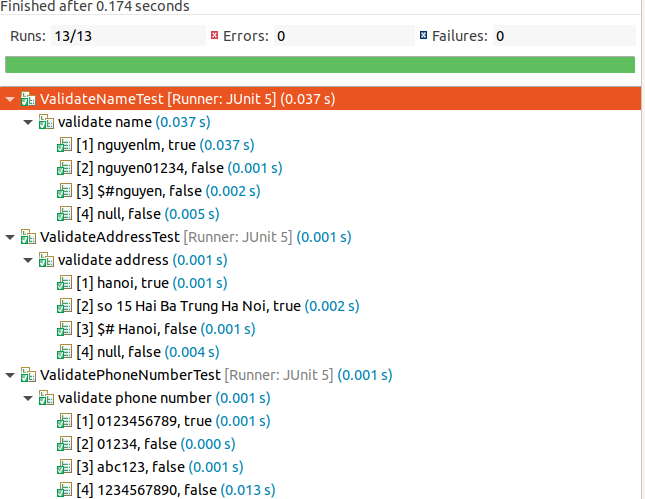
Chỉnh sửa method validatePhoneNumber

Ảnh có chứa văn bản

Mô tả được tạo tự động

Kết quả thu được sau khi chạy testẢnh có chứa văn bản

Mô tả được tạo tự động



# 2. Lập trình

## 2.1. Bắt đầu với code sample

## 2.2. Làm quen với Javadoc

* JAVADOC thường được mô tả bởi các annotation (bắt đầu bởi @), một vài loại annotatio phổ biến trong JAVADOC
  + ***@author***: chỉ tên tác giả của đoạn code hoặc có thể là người đóng góp nhiều nhất. Thường được áp dụng cho các level: class hoặc package
  + ***@param***: Mô tả tham số truyền vào một phương thức hoặc constructor
  + ***@return***: mô tả giá trị trả về của một class hoặc phương thức
  + ***@since***: phiên bản mà thuộc tính được thêm vào
  + ***@throws***: loại exception mà phương thức có thể tung ra
  + ***@deprecated***: chỉ cho người khác biết là phương thức hoặc class này không còn được sử dụng nữa
  + {***@link***}: tạo liên kết tới những phương thức hoặc phần nội dung khác

## 2.3. Thực hành tạo Javadoc với Eclipse

Attributes, classẢnh có chứa văn bản

Mô tả được tạo tự động

Get methodẢnh có chứa văn bản

Mô tả được tạo tự động

**allowMethods** method, **getConnection** methodẢnh có chứa văn bản

Mô tả được tạo tự động

Thêm comment cho phương thức

Ảnh có chứa văn bản

Mô tả được tạo tự động

## 2.4. Làm quen với refactoring

Code chúng ta sau khi refactoring sẽ như sau

Ảnh có chứa văn bản

Mô tả được tạo tự động

Ảnh có chứa văn bản

Mô tả được tạo tự động

**readResponse method, getConnection method**

Ảnh có chứa văn bản

Mô tả được tạo tự động

# 3. Bài tập về nhà

Trên link github