

## Bài tập tuần 10

### Xây dựng phần mềm

#### Mục tiêu

- Thực hiện các bài tập (câu hỏi) về các nội dung trong Xây dựng phần mềm
- Cài đặt các chức năng của phần mềm cho nhóm chức năng đã chọn trong bài toán:
  - o Viết mã xử lý trong các lớp Controller
  - o Luyện tập các quy ước viết mã và phong cách lập trình
  - o Xây dựng sơ đồ lớp chi tiết cho phần mềm

#### Đánh giá

- Hoàn thành các bài tập về các nội dung trong Xây dựng phần mềm
- Hoàn thành cài đặt các chức năng của phần mềm
- Xây dựng thành công sơ đồ lớp chi tiết

#### Phần I:

---

##### Bài 1.1

- a) Cài đặt phần mềm là triển khai thiết kế chi tiết thành chương trình?
  1. Đúng
  2. Sai
- b) Mã nguồn dễ bảo trì là?
  1. Dễ hiểu
  2. Dễ sửa lỗi
  3. Dễ nâng cấp
  4. Tất cả các phương án trên
- c) Các quy định về chuẩn viết code được đề xuất bởi?
  1. Khách hàng
  2. Ngôn ngữ lập trình
  3. Công ty
  4. Một ai đó thông minh
- d) Đặt tên biến và tên hàm nên?
  1. Có nghĩa, gợi nhớ
  2. Càng dài càng tốt
  3. Viết bằng chữ hoa cho rõ

4. Tất cả các phương án trên đều đúng
- e) Việc chú thích các mô-đun là để làm rõ?
1. Mục đích, chức năng của mô-đun
  2. Tham số, giá trị trả về
  3. Người viết, thời gian sửa đổi mô-đun
  4. Tất cả các phương án trên đều đúng

## Bài 1.2

- a) Tìm hiểu và nêu các phương pháp gỡ lỗi phổ biến?
- b) Mục đích của "code review" là gì ?

## Bài 1.3

Thảo luận nhóm và xây dựng bộ quy ước viết mã + phong cách lập trình chung cho các thành viên trong nhóm?

### Ví dụ:

- Đầu mỗi tệp mã nguồn, khai báo một header mô tả các thông tin:
  - Tên người lập trình
  - Ngày
  - Tên của dự án đã lưu
  - Tên của giáo viên
  - Tên lớp
  - Tên của bất kỳ ai đã giúp bạn
  - Mô tả ngắn gọn về những gì chương trình thực hiện

#### Sample Header

```
/**
 * MyClass <br>
 *
 * This class is merely for illustrative purposes. <br>
 *
 * Revision History:<br>
 * 1.1 - Added javadoc headers <br>
 * 1.0 - Original release<br>
 * Class: 111678, Teacher's Name: ...
 * @author Ng.Van A
 * @version 1.1, 31/07/2020
 */
public class MyClass {
    . . .
}
```

- Ví dụ: Java Coding Conventions

## Java Coding Conventions on One Page

William C. Wake (William.Wake@acm.org), 2-17-2000

Specify a package (not default)

```
package com.some.display;
```

General conventions: see <http://java.sun.com/docs/codeconv/index.html>

May use "."

```
import java.util.*;
```

Each word capitalized

```
public class NounPhrase implements Interfaceable {
```

Each word capitalized: noun phrase + optional "able" or "ing"

All caps with "\_" separator

```
    public static final int MIN_SIZE = 22;
```

Constant

```
    static int height;
```

No special prefix for "static" variables

```
    String lastName;
```

Noun phrase. Fields not "public".

Javadoc conventions: see <http://java.sun.com/products/jdk/javadoc/writingdoccomments/index.html>

```
    /** @deprecated Use getLastName(), not getName() */
```

One-liners are OK

```
    public String getName() {return lastName;}
```

JavaBeans naming conventions: see <http://java.sun.com/beans/docs/beans.101.pdf>

Braces here and here

```
    public int getScore() {
        if (height < MIN_SIZE) {
            return MIN_SIZE;
        }
        return height;
    }
```

4-space indent throughout

```
protected void paste(String filename, Vector v) throws IOException {
    Writer out;
    try {
        out = new FileWriter(filename);
        super.save(v);
        out.println(v.size());
    } catch (FileNotFoundException fnf) {
        Log.log(fnf);
        throw fnf;
    } finally {
        if (out != null) try {out.close();} catch (Exception ignored) {}
    }
}
```

Exceptions either re-thrown or handled

"Finally" clause to release resources

Catch any exceptions in "finally" clause so original exception is reported

- Công cụ hỗ trợ: **Checkstyle** là công cụ phân tích mã tĩnh được sử dụng trong phát triển phần mềm để kiểm tra xem mã nguồn Java có tuân thủ các quy tắc mã hóa hay không.



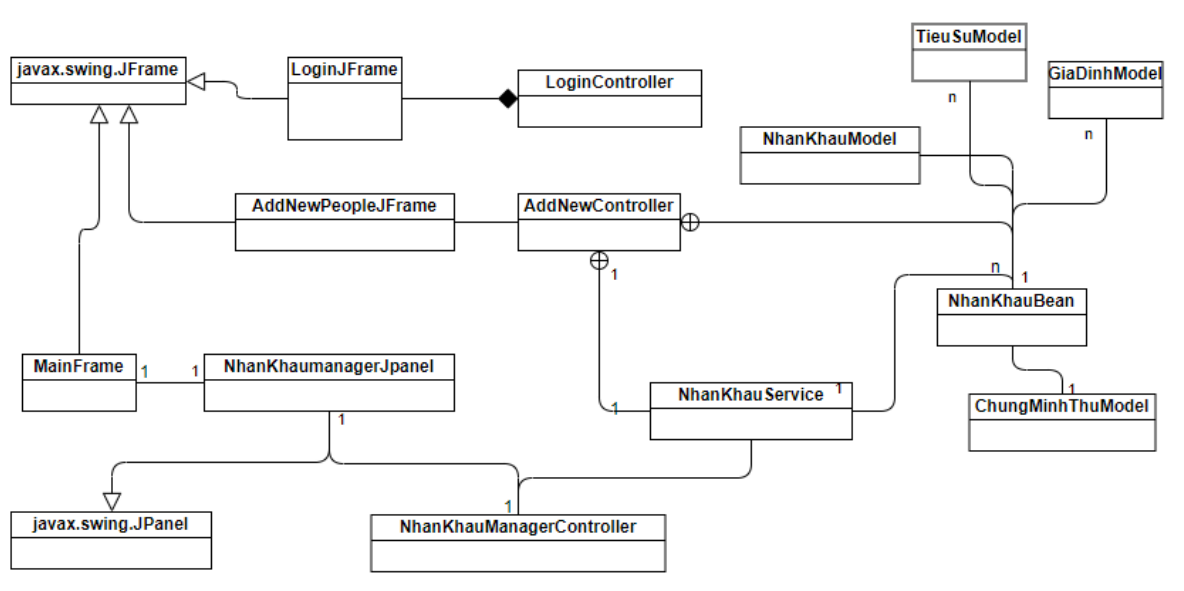
## Phần II: Cài đặt các chức năng của bài toán (case study)

## 1. Hoàn thành lập trình các chức năng của bài toán (case study)

- Viết mã thực thi các chức năng, chú ý tuân thủ các quy ước viết mã nguồn và phong cách lập trình mà các thành viên trong nhóm đã thống nhất.

## 2. Xây dựng sơ đồ lớp chi tiết cho giải pháp của bài toán

- Sau khi đã hoàn thiện viết mã cho các lớp → xây dựng **sơ đồ lớp**
- Viết các mô tả quan hệ giữa các lớp.
- Ví dụ:



**Phần nội dung này các nhóm làm vào trong file .docx (báo cáo)**

## Nội dung bài tập tự làm

**HẾT**