**CODING STANDARD**

# 1. Đặt tên

## 1.1 Các quy tắc chung

* Sử dụng tiếng Anh khi đặt tên và không đặt tên dài quá 20 ký tự
* Bình thường, các tên không được đặt gần giống nhau về mặt ký tự để tránh nhầm lẫn
* Tên object, table, class, entity, table, attribute phải là danh từ hoặc cụm danh từ
* Tên method, quan hệ, procedure, relation phải là động từ hoặc cụm động từ

## 1.2 Quy tắc đặt tên trong SQL

* Tên bảng, trigger đặt theo quy tắc Pascal Case
* Tên thuộc tính, function, proceude đặt theo quy tắc Camel Case
* Tên thuộc tính của bảng được đặt theo cấu trúc: *tên bảng + tên thuộc tính*

**Ghi chú:** Trong trường hợp tên bảng quá dài thì sử dụng các tên thay thế trong bảng dưới đây

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên đầy đủ** | **Tên viết tắt** |  | **Tên đầy đủ** | **Tên viết tắt** |
| Employee | emp |  | Import Receipt | iprt |
| Customer | ctm |  | Export Receipt | exrt |
| Warehouse | wrh |  |  |  |
| Invoice | inc |  |  |  |
| Invoice Detail | ind |  |  |  |
| Branch Warehouse | bwh |  |  |  |
| Material | mat |  |  |  |

## 1.3 Đặt tên trong Java

* Tên package đặt theo quy tắc Low Case
* Tên class đặt theo quy tắc Pascal Case
* Tên biến, phương thức, thuộc tính đặt theo quy tắc Camel Case
* Tên hằng đặt theo quy tắc Upper Case, sử dụng dấu \_ giữa các từ

## 1.4 Đặt tên trong XML Config

* name của form-bean, action, forward đặt theo quy tắc Camel Case
* path đặt theo quy tắc Low Case sử dụng dấu – giữa các từ

## 1.5 Đặt tên file

* Tên file được đặt theo quy tắc Camel Case
* Thông thường, các file thực hiện cùng chức năng thì có cùng tên chỉ khác nhau về phần hậu tố

Bảng hậu tố trong mô hình MVC

|  |  |  |  |
| --- | --- | --- | --- |
| **Layer** | **Hậu tố** | **Phần mở rộng** | **Ví dụ** |
| View | <none> | jsp | Login.jsp |
| Action | Action | java | LoginAction.java |
| Action Form | Form | java | LoginForm.java |
| Bussiness | BO | java | LoginBO.java |
| Data Access | DAO | java | LoginDAO.java |
| Bean | <none> | java | User.java |

# 2. Java Coding Convention

## 2.1 Cấu trúc chung

[package section]

[import section]

[general comment section]

[code section]

## 2.2 Convention

* Giữa các section, lệnh điều khiển phải sử dụng một dòng trống (empty line) để phân cách
* Giữa các khối try, catch, finally và câu lệnh đầu tiên trong khối phải sử dụng một empty line
* Các import thư viện hệ thống phải đặt trên các import tự tạo
* Các import có cùng chức năng được phân thành một nhóm và phải có một empty line giữa chúng

import util.Constant;

import util.Util;

import model.bean.Departments;

import model.bean.Designation;

import model.bean.EmpStatus;

import model.bean.EmployeeDetails;

import model.bean.Establishment;

import model.bean.Functions;

import model.form.EmployeeDetailsForm;

* Kiểu trả về của hàm, tối thiểu phải là bool
* Các hàm chỉ có tác dụng in thông báo cho phép kiểu trả về là void
* Chú thích trên một dòng phải có một space sau ký tự // và phải viết hoa chữ cái đầu tiên

// Get element from database

* Phải viết JavaDoc cho các hàm ở các lớp DAO, BO

# 3. JSP Coding Convention

## 3.1 Cấu trúc chung

[tag-lib section]

[general comment section]

[code section]

## 3.2 Convention

* Phải có một empty line giữa các section
* Tất cả các khai báo css được đặt ở tag head
* Tất cả các khai báo javascript, jquery, ajax được đặt ở tag footer

# 4. Cấu trúc project

src

controller.action

model.bean

model.bo

model.dao

model.form

config.property

web

META-INF

WEB-INF

js

css

template.css

font

NotoSan.ttf

OpenSan.ttf

template.jsp

index.jsp

# 5. Java Quick Code

|  |  |
| --- | --- |
| **Quick code** | **Describe** |
| \_action | Tạo class action |
| \_BaseDAO | Tạo lớp BaseDAO |
| \_comment | Tạo generic comment |
| \_DaoClass | Tạo lớp DAO thừa kế từ lớp BaseDAO |
| \_date | Lấy ngày hiện tại của hệ thống |
| \_executeSelectQuery | Tự động thêm khối try/catch khi xử lý trên ResultSet |
| \_FormClass | Tạo lớp ActionForm |
| \_package | Lấy tên package của lớp hiện tại |