

JAVA – Assignment 11

Cách nộp bài: xem folder "Hướng dẫn nộp bài qua Git"

Chú ý:

Tạo 1 Project đặt tên là "TestingSystem_Assignment_11"

Tạo 3 package entity, backend, front-end

Trong phần backend sẽ tạo các class Exercise1, Exercise2, ... mỗi method là 1 Question

Trong front-end tạo các class Program1, Program2, Program3, ... để demo kết quả của các Exercise1, Exercise2, Exercise3, ...

Chú ý:

Tất cả các bài bên dưới đều comment theo template đã cho

Exercise 1: Basic

Question 1: (Sử dụng Database Testing System đã xây dựng ở SQL)
Tạo structure 3-tiers.

Gợi ý:

Tạo project maven & config Jdbc library

Tạo 3 package: frontend, backend, database, utils

Trong package utils copy hết các file IOManager, FileManager, ScannerUtils, JDBCUtils, ...

Question 2: tiếp tục question 1 (entity)

Trong package entity tạo các class: Account, Group, ...

Chú ý: config theo Object hướng đối tượng như bài 1 Java (không config groupID như trong SQL)

Question 3: tiếp tục question 1 (backend)

Trong package backend tạo 3 subpackage: presentationlayer, businesslayer, datalayer.

- a) Trong subpackage datalayer tạo class AccountRepository, trong class tạo method getListAccounts() để lấy tất cả danh sách account từ database

Gợi ý: sử dụng JDBC để lấy tất cả danh sách account từ database

- b) Trong subpackage businesslayer tạo ra class AccountService, trong class tạo method **getListAccounts()** và return ra **List<Account>**

Gợi ý: trong method getListAccounts() sẽ gọi tới method getListAccounts() của class AccountRepository

- c) Trong subpackage presentationlayer tạo ra class AccountController, trong class tạo method **getListAccounts()** và return ra **List<Account>**

Gợi ý: trong method getListAccounts() sẽ gọi tới method getListAccounts() của class AccountService

Question 4:

Trong package frontend tạo class Program.java demo method trên

Exercise 2: Basic (interface)

Question 1: tiếp tục Exercise 1

Để tránh phụ thuộc giữa các package ở backend ta sẽ tạo interface cho các service và repository (đặt tên như trong hướng dẫn slide).

Question 2:

Tương tự tạo các chức năng createAccount, getListAccounts, getAccountByID(int id), isAccountExists(String username), isAccountExists(int id), updateAccountByID(int id), deleteAccount(int id)

Exercise 3: Validation

Question 1: Validation

Tạo validation & xử lý exception ở class controller, repository cho các chức năng trên.

Question 2: Handling business logic

Handling business logic ở class service cho các chức năng trên.

Exercise 4: Demo

Question 1: Viết menu ở frontend để demo các chức năng

Exercise 5 (Optional):

Question 1:

Làm tương tự với entity Group

Exercise 6 (Optional):

Chuẩn bị project template để thi

- Tạo sẵn template database và kết nối JDBC

- Tạo sẵn File ScannerUtil có các method nhập vào Email, sdt, ...
- Làm sẵn các chức năng CRUD và demo
- Tạo template comment sẵn

Bài thi sẽ xoay quanh CRUD và login, validate

Chú ý:

- Viết đúng coding convention
- Tuân thủ các best practice
- Không chép bài người khác (sẽ có hình thức phạt nếu bị phát hiện)

