**LAB5: JAVABEAN & DEPENDANCE INJECTION**

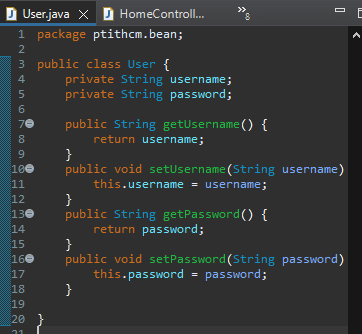
**PHẦN I**

Bài 1 (2 điểm)

Hãy tạo bean **User** gồm 2 thuộc tính **username** và **password**. Khai báo bean vào file cấu hình **spring-config-bean.xml** sau đó tạo HomeController và sử dụng **@Autowired** tiêm bean vào để sử dụng.

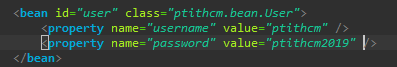
**Bước 1:** Định nghĩa bean **User**.

Bean User có 2 thuộc tính là **username** và **password**.



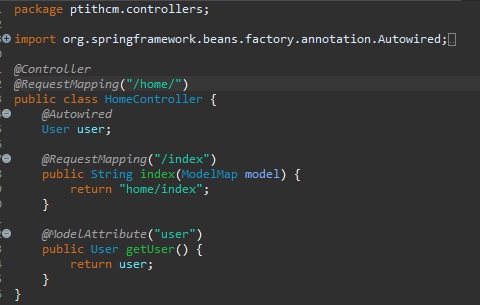
Bước 2: Khai báo bean

Khai báo bean sau vào file **spring-config-mvc.xml** chứa khai báo sau:



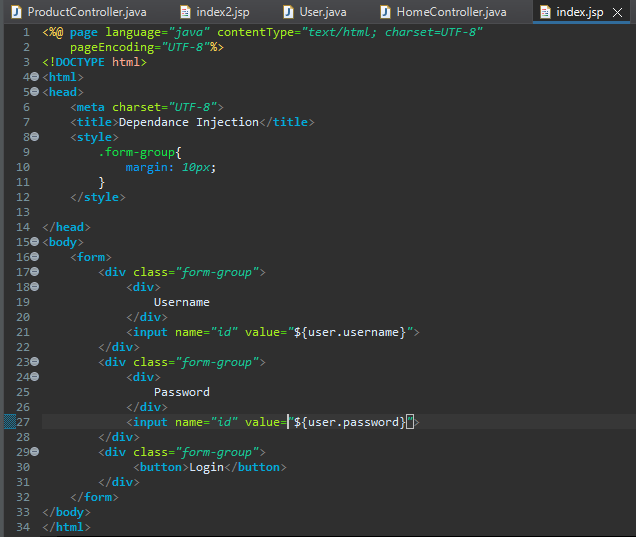
Bước 3: Tiêm bean vào Controller

Sử dụng **@Autowired** để tiêm bean vào **HomeController** sau đó đặt nó vào **ModelAttribute** bằng cách sử dụng **@ModelAttribute(“user”)**.

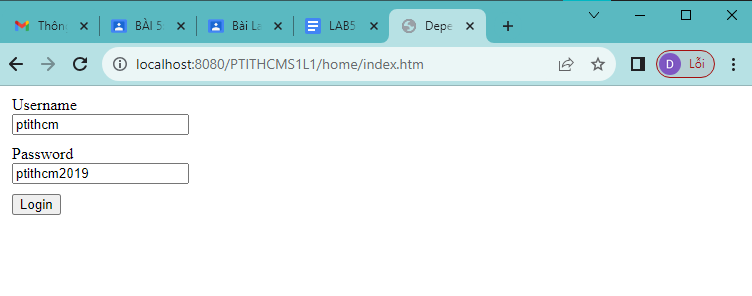


Bước 4: Hiển thị thông tin của bean lên view

Tạo view **index.jsp** và hiển thị thông tin của bean user như sau:



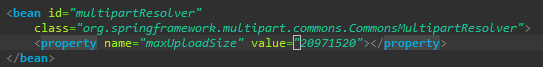
Bước 5: Chạy home/index.htm sẽ thấy thông tin của bean user hiển thị lên form



Bài 2

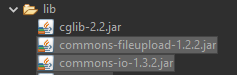
Bước 1: Khai báo bean **CommonsMultipartResolver** cho phép xử lý upload file.

Chú ý: mặc định Spring chỉ cho tổng kích thước các file được upload là 2MB. Để thay đổi kích thước cho phép cần điều chỉnh giá trị thuộc tính **maxUploadSize**.

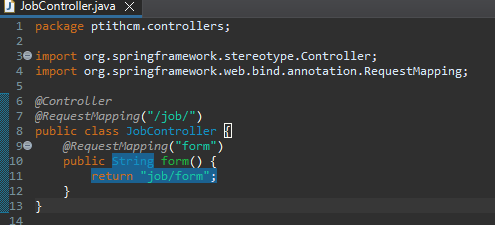


Bước 2: Thư viện

Khi sử dụng bean CommonsMultipartResolver bạn cần phải cung cấp 2 thư việc sau vào thư mục lib.

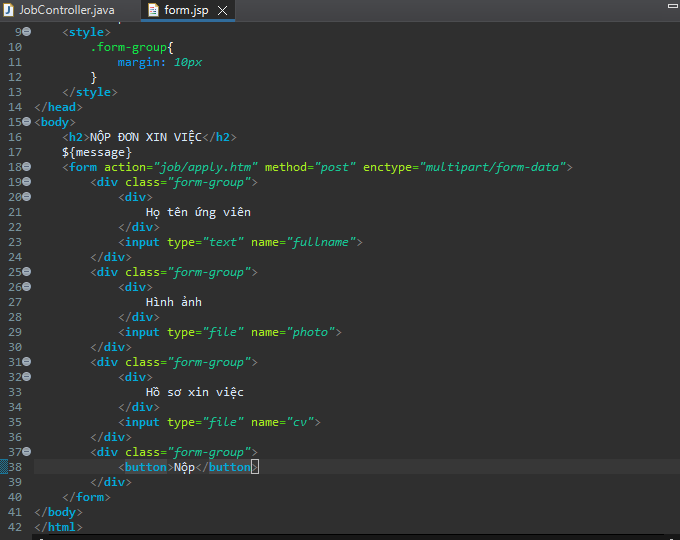


Bước 3: Xây dựng phương thức action hiển thị form xin việc



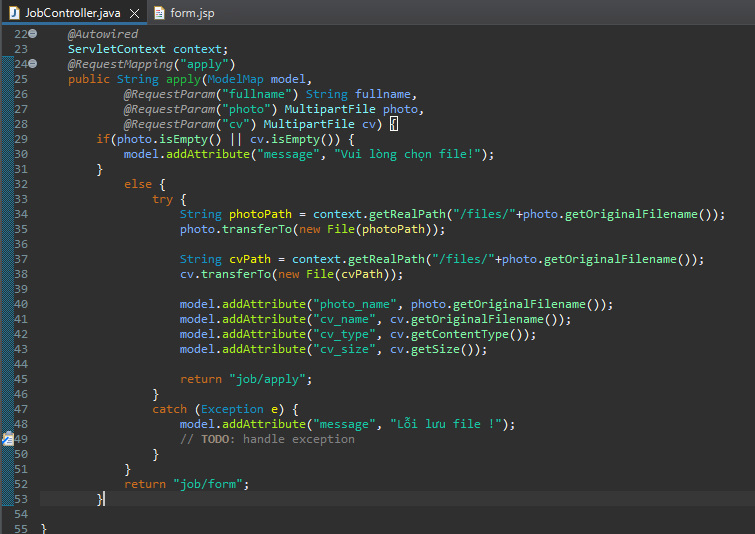
Bước 4: Thiết kế form xin việc form.jsp

Form này chứa 3 control (2 file và 1 text) có tên là fullname(text), photo(file) và cv(file). Do có chứa file nên form cần có method là post và enctype phải là multipart/form-data.



Bước 5: Xây dựng phương thức action xử lý file upload

Bổ sung mã sau vào JobController (Yêu cầu kiểm tra thêm loại file cho upload lên server).



Bước 6: Xây dựng view apply.jsp hiển thị thông tin file đã upload