**Bài1,2**

Application.properties: file cấu hình tài nguyên mặc định của ứng dụng

-Controller chứa các mapping method điều khiển các request của người sử dụng

+để tạo ra các request kích hoạt được mapping method thì request chứa các thông tin định vị gồm(URL,method,params)

-View là thành phần render giao diện phía back-end.

-Model và scopes(request,session,application) chức các dữ liệu (các attribute) để chia sẻ vơi view

-@RequestMapping được sử dụng để ánh xạ(URL,method và param) với một phương thức

-RequestParam được dùng để nhận tham số người dùng

-@RequestPart được sử dụng để nhận file upload từ client

-Spring mvc cho phép người dùng sử dụng HttpServletRequest để nhận tham số người dùng như trong lập trình servlet

-Ưu nhược điểm của HttpServletRequest và @RequestParam,javabean

1.HttpServletRequest

-ưu:giống servlet(gần gũi)

-nhươck điểm:+phải tự chuyển kiểu

+nhiều tham số

[2.@RequestParam](mailto:2.@RequestParam):

1.Ưu:

1.tự chuyển đổi kiểu dữ liệu

+giá trị mặc định

-Nhược : nhiều tham số

3.Javabean

Ưu:code đơn giản,rõ ràng với nhiều tham số

Nhược điểm:phải viết Javabean

**Bài 3:**

-Databinding là sự kết nối dữ liệu của bean trong Model vào các điều khiển form và ngược lại

Application.properties

@Controller

View

Model và scopes

-@RequestMapping

RequestParam

Databinding

Spring bean là các bean được spring quản lý:các bean bao gồm:

+bean hệ thống(built-in)

-httpServletRequest,response,httpsession,servletContext

+bean do người dùng sử dụng(user-defined) nạp vào

-Di(tiêm các đối tượng cần thiết) là các thức liên kết với các Spring Bean vào các biến cần thiết của chươn trình

-IOC container(đảo ngược điều khiển) là động cơ làm nhiệm vụ nạp bean từ bên ngoài vào hệ thống nhằm cho phép điều khiển theo mã tuỳ biến của bean bên ngoài

-IOC Container nạp bean bằng 2 cách:

+cách 1:tạo lớp có đính kèm các anotation :

* @Controller
* @Conponent
* @Service
* @Repository

+ cách 2: tạo lớp bất kỳ và viết phương thức tạo bean kèm @Bean khai báo trong file cấu hình @Configuration

-khi khởi động ứng dụng,Spring sẽ tìm kiếm các anotation đã được định nghĩa trên các lớp và các phương thức cùng với các scope để tạo và nạp các đối tượng vào thời điểm phù hợp

@Autowired

-dựa vào kiểu dữ liệu để tìm kiếm và liên kết với spring bean có kiểu phù hợp

-nếu có nhiều hơn một spring bean có kiểu phù hợp với biến thì hệ thống sẽ báo lỗi

\*\*Giải quyết xung đột:

Cách 1:Cấu hình @Bean với @Primary để báo cho hệ thống đó bean chính

Cách 2: đặt id cho bean và sử dụng @Qualifier(id) bên cạnh @Autowired

@Primary được sử dụng kèm với @Bean để báo cho IOC biết đây là bean chính.Nếu các bean cùng kiểu thì spring sẽ ưu tiên liên kết với bean này

@Qualifier được sử dụng kèm với @Autowired để liên kết đến Spring Bean theo kiểu và id

3 cách tiêm:

\*\*Tiêm vào field

@Autowired

@Qualifier(“bean1”)

Acount acount;

\*\*Tiêm vào setter

@Autowired

Public void set Acount(@Qualifier(“bean1”) Acount acount){}

\*\*Tiêm vào contructor

@Autowired

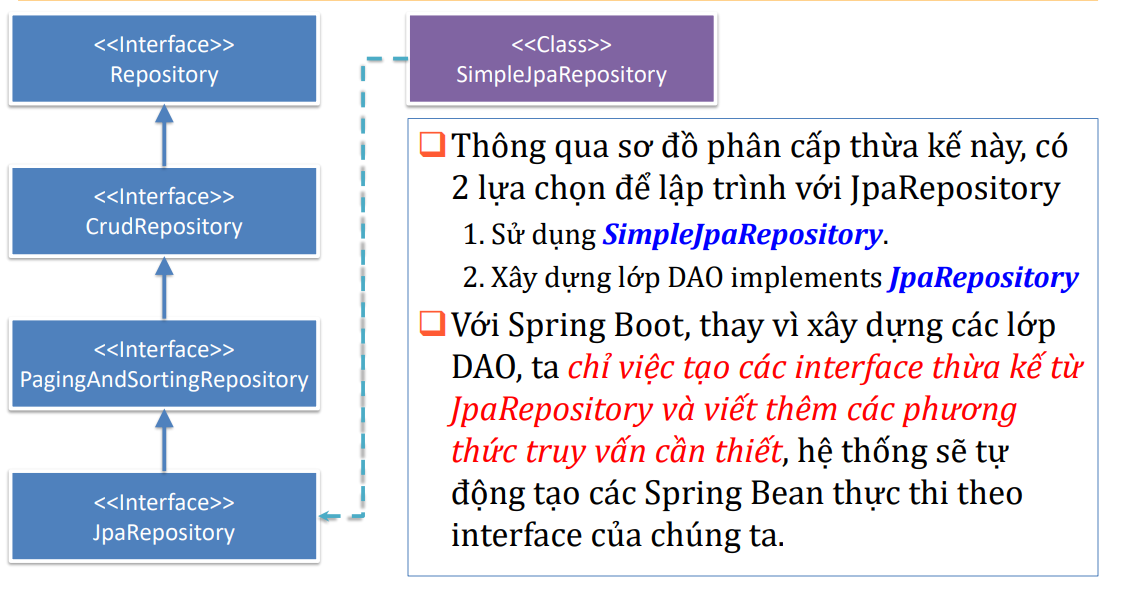
Public MyController(@Qualifierp(“bean1) Acount acount){}

-thực chất của IOC để làm gì?

-đối với các ứng dụng có độ tuỳ biến cao,có nhiều lời gọi phương thức trong chương trình mà không biết nó thực hiện chức năng gì vì còn phụ thuộc nhiều yếu tố trong tương lai

-để tránh sự phụ thuộc vào mã trong class,khi tiêm người ta sử dụng interface.Điều này có nghĩa nếu chúng ta thay lớp này bằng lớp khác thì ứng dụng sẽ thực hiện chức năng theo mong muốn của làm mới(đây chính là mục tiêu của IOC-đảo ngược sự điều khiển)

Bài 5:



Sort: cung cấp các tiêu chí sắp xếp

Pageable:cung cấp các tiêu chí sắp xếp và phân trang (nếu cần)

Page: chứa dữ liệu của trang đã được truy vấn thông qua kết quả phân trang

SimpleJpaRepository(Class, EntityManager):Phương thức khởi dựng này tạo một đối tượng DAO tương ứng với một Entity.Từ đó có thể gọi các phương thức phù hợp để làm việc với CSDL

BÀI 6:

-JpaRepositoty API cung cấp @Query giúp viết mã truy vấn cho các phương thức khai báo trong interface JpaRepository

-Spring sẽ tự động sinh mã thưc thi cho phương thức có @Query để thực hiện câu lệnh JpQL