5/15 백엔드

RestController의 액션 메소드는, 데이터를 클라이언트에 전송하기 때문에, 뷰(view)가 필요 없다.

그래서 뷰의 이름을 리턴할 필요없고, 클라이언트에게 전송하기 위한 데이터를 리턴한다

액션 메소드의 리턴값은 프론트엔드에 전달할 데이터

### @Entity

JPA Entity 클래스 앞에 @Entity 어노테이션을 붙여야 한다.

### @Id

기본키(primary key)에 해당하는 멤버 변수에 @Id 어노테이션을 붙여야 한다.

### @GeneratedValue(strategy = GenerationType.IDENTITY)

기본키가 Auto Increment 필드이거나 Identity 필드인 경우에

이 어노테이션을 붙여야 한다.

public interface DepartmentRepository extends JpaRepository<Department, Integer> {}

Department: 엔터티 클래스 이름 Integer: primary key의 타입

**->** Department department = departmentRepository.findById(id).get();

department 테이블에서 기본키 id 값으로 레코드를 조회한다.

Optional<Department> 타입의 객체가 리턴된다.

이 객체의 get 메소드를 호출하면 Department 객체가 리턴된다.

department 와 student 관계는 1 대 다 관계이다.

student 테이블의 departmentId 필드는 소속 학과를 가르키는 외래키(foreign key) 이다.

외래키는 무조건 1 대 다 중 무조건 다에 속한다.

Single-Valued Association 구현

학생클래스

@ManyToOne // 학생 여러명이 학과 한 개에 소속되어 있다.

@JoinColumn(name = "departmentId") // 학생 테이블의 외래 키는 departmentId 필드이다

Department department; // 학생이 속한 학과는 한 개이다.

Collection-avlued Association 구현

학과클래스

@OneToMany (mappedBy=”department”) //학생클래스 조인정보

List<Student> students;

Student 와 Department 관계는 Many To One

${ st.department.name }

student 객체에 department 속성이 있고, 이 속성의 값은 Department 객체이다.

Department 객체에는 name 속성값이 있다.

getter/setter는 속성

<form method="post" th:object="${student}">

<input type="text" th:field="\*{studentNo}" />

student 이름으로 전달된 model attribute의

studentNo 속성값을 입력하기 위한 input

<select th:field="\*{department.id}">

student 이름으로 전달된 model attribute의

studentNo 속성 객체의 id 속성값을 입력하기 위한 select

<option th:each="dt : ${ departments }"

th:value="${ dt.id }" th:text="${ dt.name }">

</option>

학과를 선택하기 위한 option 태그를 출력하기 위한 소스 코드.

departments 이름으로 전달된 model attribute 학과 목록 객체.

departments 학과 목록의 학과 객체 각각을 dt 변수에 대입하며, option 태그를 반복 출력한다.

id, name 부분은 학과 객체의 속성이다.

<select th:field="\*{department.id}">

선택한 option 태그의 value 값이, 즉 학과 id 값이

"department.id" 이름의 request parameter가 되어 서버에 전달된다.

student/edit 입력폼이 처음 뜰 때,

student 이름으로 전달된 model attribute 객체의

department 속성의 id 속성값과 일치하는 option 태그가 선택된 상태가 된다.

th:each="변수 : ${ 목록}" 반복 지시자가 중접되었다.

<div th:each="dp : ${ departments }"> departments 학과 목록의 학과 각각에 대해서 반복

<tr th:each="st : ${ dp.students }"> dp.students 학생 목록의 학생 각각에 대해서 반복

1) DB에서 학번으로 학생 조회하는 작업을 빠르게 하려면 무엇을 해야 하는가?

-> 학번 필드에 인덱스 만들기

5) 아래 st thymeleaf 변수에 대입된 Student 객체의 studentNo 값이 "202014199" 이고, id 값이 3일 때,

   아래 소스 코드에 의해서 출력되는 html 태그를 쓰시오.

<td><ath:text="${ st.studentNo }"th:href="${ 'edit?id=' + st.id }"></a></td>

-> <td><ahref="edit?id=3">202014199</a></td>

6) 다음 SQL은 어떤 레코드를 조회하는가?

SET @email = '';

SELECT \* FROM student WHERE @email = '';

-> 모든레코드