




≡ 한 줄 소개	spring-boot를 이용하여 CRUD구현과 Front와 Backend를 분리하였습니다.
📅 진행 기간	@2022/09/26 → 2022/09/30
☰ Skills & Libraries	<div>ajaxcsshtmljavajavascriptjsonspring-boot</div>
🔗 Github	

 <Spring\_board 시스템>은 yaml과 DB연동을 하였으며,  
기본적인 MVC 패턴으로 CRUD 구현하였습니다.  
삭제 여부 컬럼을 추가하여 탈퇴한 회원은 Y 현재 회원은 N으로 관리하였습니다.



## API 명세서

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/e353af16-8e48-4020-99ea-7f3003637054/EMP%EC%A0%95%EB%B3%B4%EC%B2%B4%EA%B3%84\\_%EA%B8%B0%EB%8A%A5%EB%AA%85%EC%84%B8%EC%84%9C\\_%EA%B0%9C%EB%B0%9C%EC%9D%BC%EC%A0%95%EA%B4%80%EB%A6%AC.xlsx](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/e353af16-8e48-4020-99ea-7f3003637054/EMP%EC%A0%95%EB%B3%B4%EC%B2%B4%EA%B3%84_%EA%B8%B0%EB%8A%A5%EB%AA%85%EC%84%B8%EC%84%9C_%EA%B0%9C%EB%B0%9C%EC%9D%BC%EC%A0%95%EA%B4%80%EB%A6%AC.xlsx)

### ▼ GET

<http://localhost:8080/api/d1/emp> : 직원 수 / 평균 급여/ 부서 수 / 보너스 합

<http://localhost:8080/api/d1/emppageall?page=+pageNum> : 전체 직원 조회

<http://localhost:8080/api/d1/empall/+empno> : 선택한 직원 번호

<http://localhost:8080/api/d1/deptpage?page=+pageNum> : 부서 전체

<http://localhost:8080/api/d1/dept/+deptno> : 선택한 부서

<http://localhost:8080/api/d1/logs?page=+pageNum> : 접속자 히스토리

<http://localhost:8080/api/d1/logs/+logId> : 선택한 접속자

### ▼ POST

<http://localhost:8080/api/d1/emp> : 직원 추가

<http://localhost:8080/api/d1/dept> : 부서 추가

### ▼ PATCH

<http://localhost:8080/api/d1/emp> : 직원 정보 수정

<http://localhost:8080/api/d1/dept> : 부서 수정

<http://localhost:8080/api/d1/empall/empno/+empno> : 직원 삭제

데이터는 곧 자산 그러므로 데이터를 삭제하지 않고,  
삭제 여부 컬럼을 추가해서 탈퇴한 회원 Y 현재 회원 N로 관리한다.

### ▼ DELETE

<http://localhost:8080/api/d1/dept/+deptno> : 부서 삭제

## 기술 스택

- Spring Boot
- MySQL
- MyBatis
- HTML/CSS/JS
- json / Ajax

## 기능 구현

- Spring 및 SpringBoot 에서 사용되는 Http 요청과 관련된 어노테이션, MVC 어노테이션, MyBatis 어노테이션 등 활용
- 로그를 추가하여 접속 url 호출과 http 호출을 하여 접속자 히스토리를 구현
- 데이터는 자산이므로 삭제 여부 컬럼을 추가하여 Y , N으로 구분하였다.
- 사원 목록 전체를 엑셀로서 다운로드 할 수 있는 기능 구현

## Spring\_board 화면



