

spring 한스푼(5주차)

4주차 review/ java 익히기/
CRUD 게시판 (변수.ver)/DB연결 /CRUD 게시판

4주차 review

스프링의 구조

- 의존성 주입(build.gradle)
 - 설정파일(application.properties)
 - 프로젝트 시작(???application)
 - 테스트(test 폴더)
-
- 파라미터(Controller)(ex : www.youtube.com/(영상주소))
 - 컨트롤러 서비스 구현(service)
 - 테이블관리(entity) (JDBC)
 - DB관리 (repository) (라이브러리 : JPA)
- 
-
- 보안설정 (security) (라이브러리 : spring security)
 - 그 외 (JWT, handling, ...)

의존성 주입(Build.gradle)

필요한 라이브러리가 있을 경우 추가

의존성을 추가 했거나 제거시 꼭 마지막엔 코끼리 아이콘을 눌러
프로젝트 빌드

설정 파일(application.properties)

프로젝트에 대한 설정 및 라이브러리에 대한 설정을 하는 곳이다.

DEBUG 같이 레벨 설정을 하여 실행 창에서 DEBUG의 로그를 볼 수 있다.

ex) `logging.level.root=DEBUG` default : info(정보만 보여줌)

어노테이션

프로그램 코드의 일부가 아닌 프로그램에 관한 데이터를 제공하고, 코드에 정보를 추가하는 정형화된 방법이다.

라이브러리마다 다르기에 라이브러리를 검색하여 사용법을 찾아야한다.

ex) LOMBOK의 @Getter/@Setter (게터/세터)를 통해 해당 클래스의 멤버들을 자동으로 할당해 줌.

Java 익히기!

뒤죽박죽한 데이터들 정렬해보기!

main 코드를 참고하여 Sorting 클래스를 구현해보자

```
public static void main(String[] args) {  
    int len = 10;  
    int max = 100;  
    int list[] = new int[len];  
    for (int i = 0; i < len; i++) {  
        double Value = Math.random();  
        list[i] = (int)(Value * max);  
    }  
    Sorting s = new Sorting(list);  
    s.print();  
    s.sort();  
    s.print();  
}
```

hint :

Sorting 클래스안엔
private int ??[] 멤버변수가
있고
생성자 함수
print메서드
sort메서드가 있다.

11 32 97 53 31 47 82 37 58 25 정렬 중입니다...

정렬 되었습니다!

11 25 31 32 37 47 53 58 82 97

종료 코드 0(으)로 완료된 프로세스

CRUD 게시판(변수.ver)

저번 시간 -> R 중 list를 보내기까지 했다.

이번 시간

C -> POST를 통해 int id를 통해 id가 없을 경우 저장

R -> GET을 통해 list에서 id값을 받아 Board 전송

U -> C쪽 POST에서 id가 있을 경우 id를 통한 Board저장.

D -> Delete에서 id값을 받아 찾고 id를 통한 Board 삭제

DB 설정

의존성 추가 - MySql, JPA

-application.properties-

```
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver  
spring.datasource.url=jdbc:mysql://192.168.0.26:3306/spring?  
useSSL=false&useUnicode=true&serverTimezone=Asia/Seoul  
spring.datasource.username=sos  
spring.datasource.password=root  
spring.jpa.show-sql=true  
spring.jpa.hibernate.ddl-auto=update  
spring.jpa.properties.hibernate.format_sql=true
```

적기만 하면 DB 연결 완료. 이후 application 실행하여 오류가 없을 경우 성공.

CRUD 게시판

C -> POST를 통해 Long id를 통해 id가 없을 경우 저장

R -> GET을 통해 list에서 id값을 받아 Board 전송

U -> C쪽 POST에서 id가 있을 경우 id를 통한 Board저장.

D -> DELETE를 통해 id값을 받아 찾고 id를 통한 Board 삭제

repository 추가하여 DB를 써보자.