

화면연동

1) 글쓰기(insert) 화면구현

지난 시간에 save라는 메서드로 db에 글을 저장하는 것까지 해보았다. postman으로 요청했을 시 해당하는 화면을 찾지 못해서 500번 에러가 발생했었다. 이제 sava에 해당하는 뷰를 만들고 연동해 보겠다.

#index.html

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <link rel="stylesheet" href="/css/home.css">

</head>

<body>

<h2>Hello Spring Boot</h2>

<a href="/save">글작성</a>

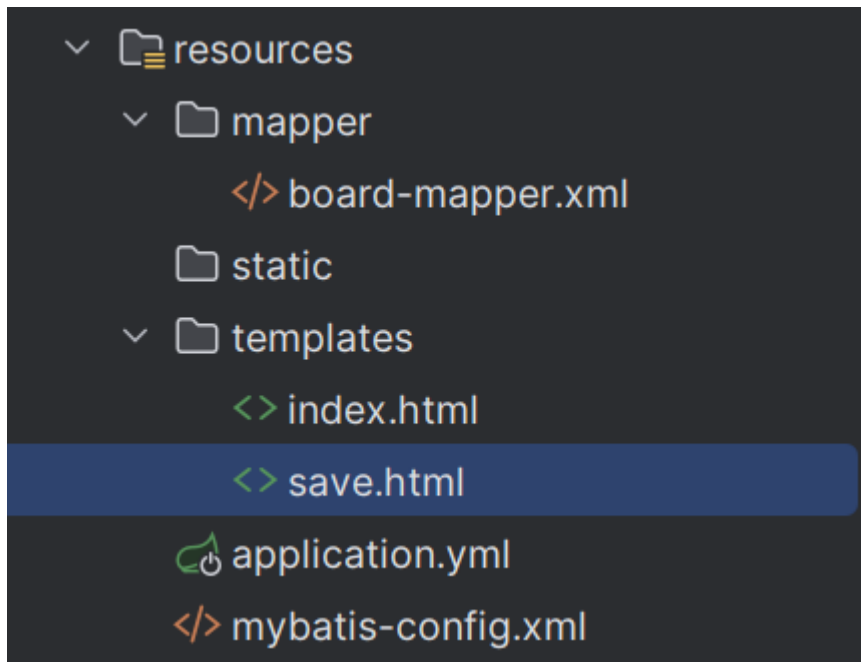
<a href="/list">글목록</a>

</body>

</html>
```

일단 맨 첫 번째 시간 스프링 프로젝트를 테스트하기 위해 만들었던 진입점인 index.html에서 글을 작성하는 페이지와 글목록을 볼 수 있는 페이지로 넘어가기 위해 버튼을 두 개 만들어준다.

그다음 글을 작성하게 될 화면인 save.html을 생성해 준다. 경로는 아래와 같다.



#save.html

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
    <meta charset="UTF-8">
```

```
    <title>Title</title>
```

```
</head>
```

```
<body>
```

```
<form action="/save" method="post" enctype="multipart/form-data">
```

```
    제목: <input type="text" name="boardTitle"> <br>
```

```
    작성자: <input type="text" name="boardWriter"> <br>
```

```
    비밀번호: <input type="text" name="boardPass"> <br>
```

```
    내용: <textarea name="boardContents" cols="30" rows="10"> </textarea>
```

```
    <input type="submit" value="작성">
```

```
</form>
```

```
</body>
```

추후 사진첨부/다중첨부 기능을 위해 form태그의 enctype를 multipart/form-data로 설정해 준다.

2) Controller와 화면 연동

#BoardController

@Controller

@RequestMapping

@Sf4j

public class BoardController {

private final BoardService boardService;

@GetMapping("/save")

public String save(){

return "save";

}

@PostMapping("/save")

public void save(BoardDTO boardDTO){

boardService.save(boardDTO);

}

}

마지막으로 컨트롤러에 save메서드를 작성해 준다. Get요청으로 들어오는 /save는 글작성 화면으로 넘어가는 메서드이고 Post요청으로 들어오는 /save는 글을 DB에 저장하는 기능을 해준다.

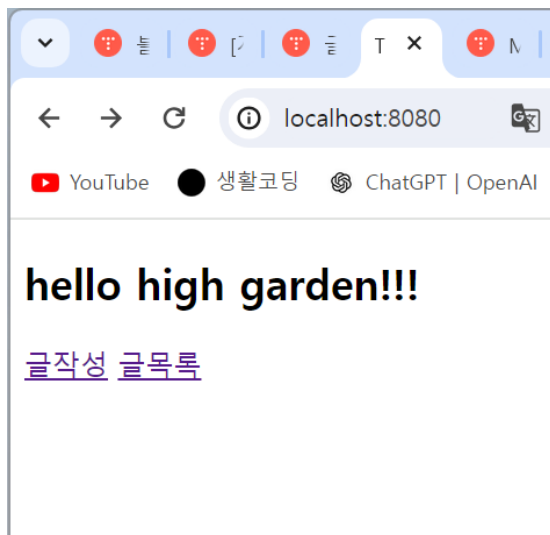
Java는 오버로딩을 제공하기 때문에 같은 이름의 메서드를 매개변수의 타입이나 개수에 따라서 다르게 선언할 수 있다. 그리고 같은 URL요청이라도 HTTP메서드의 종류에 따라서 다르게 선언할

수 있기 때문에 위와 같은 코드를 작성 가능하다.

3) 테스트

여기까지 작성을 했다면 글작성 기능과 화면의 연동이 완료된 것이다. 이제 테스트를 해보자.

스프링부트 서버를 작동시키고 브라우저에 localhost:8080으로 요청을 보내면 index화면이 나온다.
그다음 **글작성** 버튼을 누르게 되면 글쓰기 화면이 나오게 된다.



←

→

↻

i

localhost:8080/save

YouTube

생활코딩

ChatGPT | OpenAI

Overvi

제목:

작성자:

비밀번호:







내용:

화면연동 테스트입니다.

작성

인덱스 화면 / 글쓰기 화면

작성버튼을 눌러서 실제로 글을 써보면 아래와 같이 DB에 잘 저장된다.

Result Grid								
Filter Rows: <input type="text"/>								
Edit:   								
Export/Import:  								
Wrap Cell Content: 								
	id	boardTitle	boardWriter	boardPass	boardContents	boardHits	createdAt	fileAttached
▶	5	화면연동 테스트	하이가든	1234	화면연동 테스트입니다.	0	2024-06-05 14:58:54	0
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

db저장

사실상 게시판 프로젝트에서 중요한 뼈대 부분을 구현하는 것이 끝났다고 할 수 있다. 다음 시간에는 나머지 CRUD기능들을 구현해 보도록 하자.