

TO-DO LIST 구현 하기

1. 프로젝트 개요

메인화면 동작

- TODO, DOING, DONE 3가지 상태로 구분.
- 세가지 상태는 동일한 크기와 동일한 간격으로 배치.
- 각 카드UI간의 간격은 위/아래모두 동일한 margin으로 유지.
- 새로운 카드(할 일)가 추가되면 아래에 동일한 카드가 추가됨.
- 카드의 갯수는 계속 증가할 수 있음.
- 각 카드에는 할 일 제목/등록날짜/담당자/우선순위를 표현.
- 먼저 등록된 카드(할 일)가 위에 노출된다. 나중에 등록한 카드는 아래 추가가 되어서 노출된다.
- 각 카드에서 오른쪽 화살표를 누르면 오른쪽 상태로 이동한다. 예를들어 TODO에서 각 카드의 화살표를 누르면 Doing으로 이동한다. done상태로가면 오른쪽 화살표가 없어진다. (버튼을 눌렀을때 카드가 이동하는 건 화면의 새로고침 없이 자바스크립트만으로 구현 되어 한다.)

나의 해야할 일들

새로운 TODO 등록

TODO	DOING	DONE
자바스크립트 공부하기 등록날짜: 2018.03.10, 홍길동, 우선순위 1	다음달 계획서 작성방법 고민하기 등록날짜: 2018.01.22, 홍길동, 우선순위 1	CSS학습 등록날짜: 2018.01.10, 홍길동, 우선순위 1
리포트 제출하기 등록날짜: 2018.02.20, 홍길동, 우선순위 1	데모페이지 개발 등록날짜: 2018.01.10, 홍길동, 우선순위 1	JAVA 객체지향 설계 복습 등록날짜: 2018.01.11, 홍길동, 우선순위 1
	웹 관련 책 찾아보기 등록날짜: 2018.02.20, 홍길동, 우선순위 1	이사할 곳 알아보기 등록날짜: 2018.01.03, 홍길동, 우선순위 1
	앱폰트 공부하기 등록날짜: 2018.01.20, 홍길동, 우선순위 1	

할 일 등록

- 할 일제목, 담당자, 우선순위 를 입력해야 함.
- 할 일제목은 24자까지 입력해야 함.
- 할 일제목과 담당자는 text 형태로 입력받음.
- 우선순위는 라디오 박스로 한가지만 선택가능.
- 제출을 누르면 서버로 데이터가 전송된다. 전송된 이후에는 메인페이지에 추가된 내용이 포함 되어 노출됨.
- 할 일 등록시 필요한 데이터는 '할 일제목', '담당자', '우선순위', '등록날짜' 이다. 이중 등록날짜는 서버에서 자동으로 현재 날짜를 DB 에 저장한다.
- 내용지우기(reset)를 누르면 모든 내용이 다시 초기화됨.
- 이전을 누르면 메인페이지로 화면이 전환됨.
- 3가지 항목은 모두 필수로 입력되어야 한다.

할 일 등록

어떤일인가요?

swift 공부하기(24자까지)

누가 할일인가요?

홍길동

우선순위를 선택하세요

☐ 1순위 ☐ 2순위 ☐ 3순위

< 이전 제출 내용지우기

2. 프로젝트의 개발스펙

웹프론트엔드 기술요구사항

- 총 2개의 화면이 존재합니다. - 할 일 목록 화면 (리스트) - 할 일 등록 화면 (쓰기)
- CSS는 기획서와 동일한 수준으로 만들어야 합니다.

- 따라서 HTML 엘리먼트간의 배치와 간격은 일정하고 반듯해야 합니다.
- 글자의 크기는 일정한 수준을 유지합니다.
- CSS는 외부 라이브러리(부트스트랩)을 사용하지 않습니다.
- jQuery를 사용하지 않고, querySelector, addEventListener, innerHTML을 사용해서 DOM, EVENT 처리를 합니다.
- Ajax는 XMLHttpRequest를 사용합니다.

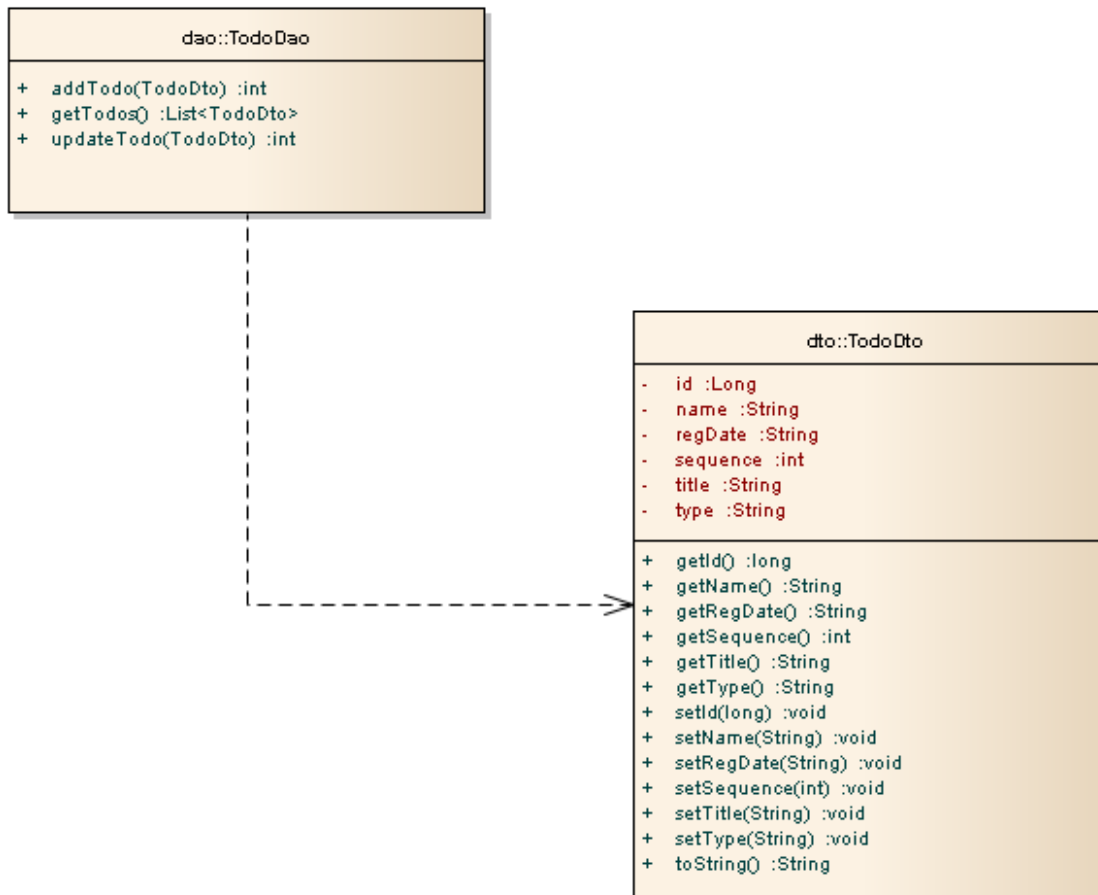
웹백엔드 기술요구사항

- 프로젝트는 maven프로젝트로 생성합니다.
- 제공된 테이블 생성 SQL을 이용해서 테이블을 생성합니다.
- TodoDto 클래스와 TodoDao클래스를 주어진 스펙에 맞게 작성합니다.
- 메인화면을 보여주기 위한 MainServlet과 main.jsp를 작성합니다.
- MainServlet은 TodoDao를 이용해 결과를 조회해서 main.jsp 에 전달합니다.
- 새로운todo등록 버튼을 클릭하면 해당 요청을 서블릿이 받아서 jsp로 포워딩하여 할 일 등록 화면을 보여줍니다.
- 할일등록폼에서 값을 입력하고 제출 버튼을 누르면 post 방식으로 요청하게 합니다.
- 해당 요청을 서블릿이 받아서 처리하게하고, 요청에 대한 모든 일이 끝나면 메인화면으로 리다이렉트 합니다.
- 메인화면에서 todo 상태변경 버튼(->)을 클릭하여 요청을 보낼 때, Todo 의 Id와 상태값을 전달하여 다음 상태로 (현재 상태가 Todo라면 doing으로 doing 이라면 done) 상태를 나타내는 컬럼값을 변경하고 응답결과로 "success"를 보냅니다.

3. 프로젝트 개발과정 참고

BE 개발 - MySql설치

1. 프로젝트 이름을 Todo로해서 maven프로젝트를 생성합니다.
2. 제공된 테이블 생성 SQL을 이용해서 테이블을 생성합니다.
3. Todo 테이블에 정보 한 건을 담을 수 있는 TodoDto 클래스를 주어진 스펙에 맞게 작성합니다.
4. Todo 테이블에 입력, 수정, 조회하는 TodoDao클래스를 주어진 스펙에 맞게 작성합니다.
5. 메인화면을 보여주기 위한 MainServlet과 main.jsp를 작성합니다.
6. MainServlet은 TodoDao를 이용해 결과를 조회해서 main.jsp 에 전달합니다.
7. main.jsp에서는 전달받은 결과를 JSTL 과 EL을 이용해 출력합니다.
8. 새로운todo등록 버튼을 클릭하면 TodoFormServlet이 실행되고, TodoFormServlet은 todoForm.jsp로 포워딩하여 할 일 등록 화면을 보여줍니다.
9. 할 일 등록폼에서 값을 입력하고 제출 버튼을 누르면 post 방식으로 TodoAddServlet으로 값이 전달되고, TodoAddServlet에서는 TodoDao를 이용해서 테이블에 저장하고 메인화면으로 리다이렉트합니다.
10. 메인화면에서 todo 상태변경버튼(->)을 클릭하면 TodoTypeServlet에게 Todo 의 Id와 상태 값을 전달하여 다음 상태로 (현재 상태가 TODO라면 DOING으로 DOING 이라면 DONE로) TodoDao를 이용해서 변경합니다.



테이블 생성 SQL

```
CREATE TABLE todo ( id BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT, title VARCHAR(255) NOT NULL, name VARCHAR(100) NOT NULL, sequence INT(1) NOT NULL, type VARCHAR(20) DEFAULT 'TODO', regdate DATETIME DEFAULT NOW(), PRIMARY KEY (id) );
```

프로그램 관련 SQL(참고용)

- 입력

```
insert into todo(title, name, sequence) values('자바스크립트공부하기', '홍길동', 1);
insert into todo(title, name, sequence) values('리포트 제출하기', '홍길동', 1);
```

- 수정

```
update todo set type = 'DOING' where id = 1; update todo set type = 'DONE' where id = 1;
```

- 조회

```
select id, title, name, sequence, type, regdate from todo order by regdate desc
select id, title, name, sequence, type, regdate from todo where type = 'TODO'
order by regdate desc
```

TodoDao와 TodoDto

주어진 클래스 다이어그램과 같이 필드와 메소드를 가지는 TodoDto와 TodoDao클래스를 작성합니다.

평가기준표

기능

기 준	세부항목
메 인 페 이 지 - 카 드 노 출	- 카드의 갯수는 계속 증가할 수 있다. - 먼저 등록한 카드(할일)가 위에 노출된다. 나중에 등록한 카드는 아래 추가되서 노출된다.
메 인 페 이 지 - 카 드 이 동	- 각 카드에서 오른쪽 화살표를 누르면 오른쪽 상태로 이동한다. 예를들어 TODO에서 각 카드의 화살표를 누르면 Doing으로 이동한다. done상태로가면 오른쪽 화살표가 없어진다. (버튼을 눌렀을때 카드가 이동하는 건 화면의 새로고침 없이 자바스크립트만으로 구현 되어 한다.)

소스코드

기준	세부항목
JAVA 가독성	- 조건문의 경우 긍정적이고 흥미로운 (주 흐름에 해당하는) 경우가 앞쪽에 위치하도록 한다. - 삼항연산자, do-while문장은 코드 가독성을 떨어트리니 되도록 사용하지 않는다. - 코드는 빈줄을 이용해 커다란 블록을 논리적인 문단으로 구분한다. - 필요하지 않은 코드는 제거한다. - 불필요한 주석문은 사용하지 않는다.
JAVA - Dao 관련	- 클래스 다이어그램에서 제시한 것과 같이 TodoDao와 TodoDto 클래스의 필드와 메소드를 작성합니다. - 클래스 다이어그램에서 제시한 것과 같이 TodoDao 클래스에 getTodos(), addTodo(), updateTodo() 메소드를 작성합니다. - 클래스 다이어그램에서 제시한 것과 같이 TodoDao 클래스의 메소드 파라미터와 리턴타입을 작성합니다.
JAVA - Servlet 관련	- MainServlet은 실행 후 그 결과를 main.jsp에게 포워딩한다. - TodoAddServlet은 사용자가 입력한 제목, 이름, 순서를 받아들여 todo 테이블에 저장한다. - TodoAddServlet은 저장 후, 메인페이지로 이동한다. - 해야할일의 ">" 버튼을 클릭하면 해당 해야할일의 id와 현재 type값을 TodoTypeServlet에게 전달한다. - TodoTypeServlet은 id에 해당하는 해야할일의 type을 TODO라면 DOING으로, DOING이라면 DONE으로 변경한다.
JAVA - main.jsp	- main.jsp는 Todo의 type에 따라서 TODO, DOING, DONE을 구분하여 출력한다.