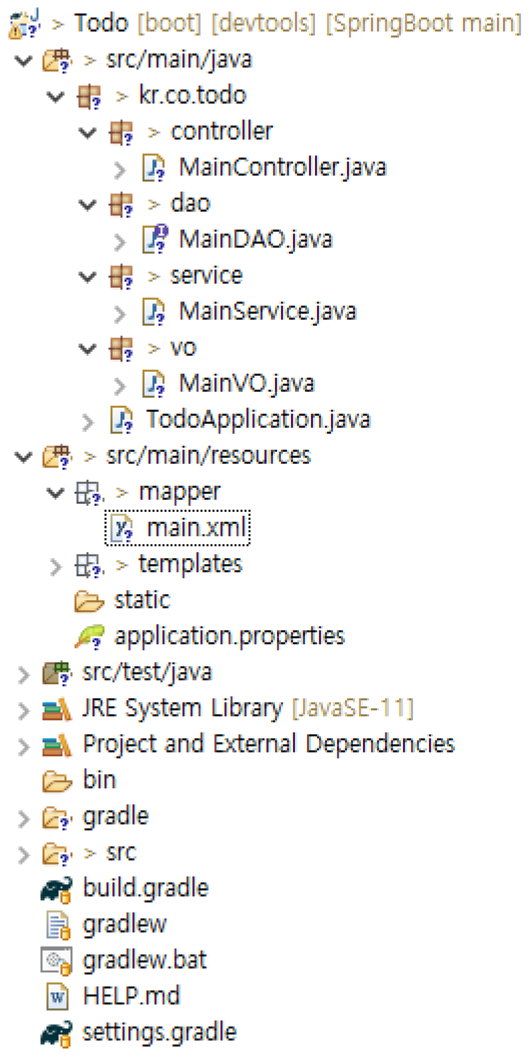
통합 구현
수행평가 실습보고서

2023/02/20 박가영

목차 table of contents

Part 1

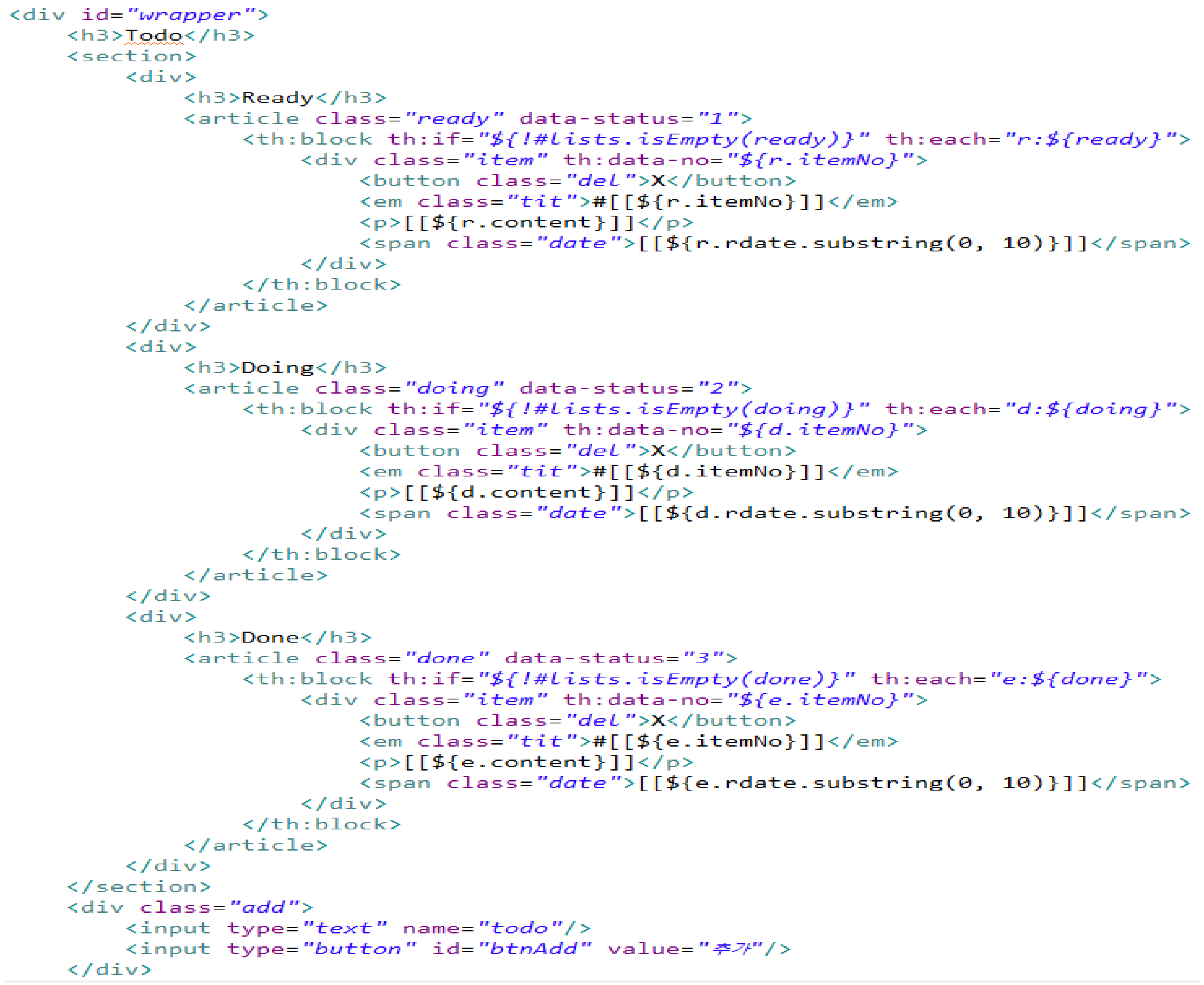
문제1. 프로젝트 생성



```
> Todo [boot] [devtools] [SpringBoot main]
  > src/main/java
    > kr.co.todo
      > controller
        MainController.java
      > dao
        MainDAO.java
      > service
        MainService.java
      > vo
        MainVO.java
      TodoApplication.java
  > src/main/resources
    > mapper
      main.xml
    > templates
    static
    application.properties
  src/test/java
  JRE System Library [JavaSE-11]
  Project and External Dependencies
  bin
  gradle
  > src
  build.gradle
  gradlew
  gradlew.bat
  HELP.md
  settings.gradle
```

문제2. 화면 구현

화면 레이아웃



```
<div id="wrapper">
    <h3>Todo</h3>
    <section>
        <div>
            <h3>Ready</h3>
            <article class="ready" data-status="1">
                <th:block th:if="${!#lists.isEmpty(ready)}" th:each="r:${ready}">
                    <div class="item" th:data-no="${r.itemNo}">
                        <button class="del">X</button>
                        <em class="tit">#[[${r.itemNo}]]</em>
                        <p>[[${r.content}]]</p>
                        <span class="date">[[${r.rdate.substring(0, 10)}]]</span>
                    </div>
                </th:block>
            </article>
        </div>
        <div>
            <h3>Doing</h3>
            <article class="doing" data-status="2">
                <th:block th:if="${!#lists.isEmpty(doing)}" th:each="d:${doing}">
                    <div class="item" th:data-no="${d.itemNo}">
                        <button class="del">X</button>
                        <em class="tit">#[[${d.itemNo}]]</em>
                        <p>[[${d.content}]]</p>
                        <span class="date">[[${d.rdate.substring(0, 10)}]]</span>
                    </div>
                </th:block>
            </article>
        </div>
        <div>
            <h3>Done</h3>
            <article class="done" data-status="3">
                <th:block th:if="${!#lists.isEmpty(done)}" th:each="e:${done}">
                    <div class="item" th:data-no="${e.itemNo}">
                        <button class="del">X</button>
                        <em class="tit">#[[${e.itemNo}]]</em>
                        <p>[[${e.content}]]</p>
                        <span class="date">[[${e.rdate.substring(0, 10)}]]</span>
                    </div>
                </th:block>
            </article>
        </div>
    </section>
    <div class="add">
        <input type="text" name="todo"/>
        <input type="button" id="btnAdd" value="추가"/>
    </div>
```

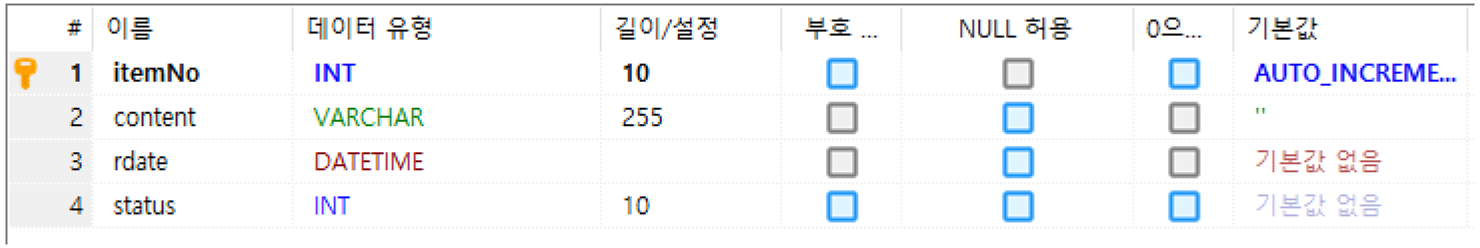
문제2. 화면 구현

style

```
*{margin:0; padding: 0;}
#wrapper {width:800px; height:auto; margin: 0 auto; overflow: hidden;}
section {width: 800px; height: auto; margin: 0 auto;}
h3 {margin-bottom: 10px;}

section > div{
    float: left;
    width: 33.33%;
    height: 100%;
    padding: 6px;
    border-radius: 10px;
    box-sizing: border-box;
}

article{
    width: 100%;
    height: 600px;
    padding: 6px;
    background: #f6f8fa;
    border: 1px solid #d8dee4;
    border-radius: 6px;
    box-sizing: border-box;
    overflow: hidden;
    overflow-y: auto;
}

.item{
    float: left;
    width: 100%;
    height: 100px;
    padding: 10px;
    margin-top: 6px;
    background: white;
    border: 1px solid #d8dee4;
    border-radius: 6px;
    box-sizing: border-box;
    z-index:10000;
}
```

```
.item > .del{
    float: right;
    background: none;
    border: none;
}

.add{
    padding: 6px;
    box-sizing: border-box;
}

.add > input{
    padding: 6px;
    box-sizing: border-box;
    outline: none;
}
```

문제3. 테이블 설계



#	이름	데이터 유형	길이/설정	부호 ...	NULL 허용	0으...	기본값
1	**itemNo**	**INT**	**10**	☐	☐	☐	**AUTO_INCREME...**
2	content	VARCHAR	255	☐	☐	☐	''
3	rdate	DATETIME		☐	☐	☐	기본값 없음
4	status	INT	10	☐	☐	☐	기본값 없음

Part 2

문제4. 기능 구현

controller

```java
@Controller
public class MainController {

    @Autowired
    private MainService service;

    @GetMapping("index")
    public String index(Model model) {
        Map<Integer, List<MainVO>> result = service.selectAll();
        List<MainVO> ready = result.get(1);
        List<MainVO> doing = result.get(2);
        List<MainVO> done = result.get(3);

        model.addAttribute("ready", ready);
        model.addAttribute("doing", doing);
        model.addAttribute("done", done);
        return "index";
    }

    @ResponseBody
    @PostMapping("insert")
    public Map<String, Object> insert(MainVO vo) {
        vo.setStatus(1);
        int result = service.insertContent(vo);
        Map<String, Object> resultMap = new HashMap<>();
        resultMap.put("result", result);
        resultMap.put("vo", vo);
        return resultMap;
    }

    @ResponseBody
    @GetMapping("delete")
    public Map<String, Integer> delete(int itemNo, int status) {
        int result = service.deleteContent(itemNo, status);
        Map<String, Integer> resultMap = new HashMap<>();
        resultMap.put("result", result);
        return resultMap;
    }
```

```java
@ResponseBody
@GetMapping("update")
public Map<String, Integer> update(int itemNo, int newstatus) {
    int result = service.updateContent(itemNo, newstatus);
    Map<String, Integer> resultMap = new HashMap<>();
    resultMap.put("result", result);
    return resultMap;
}
```

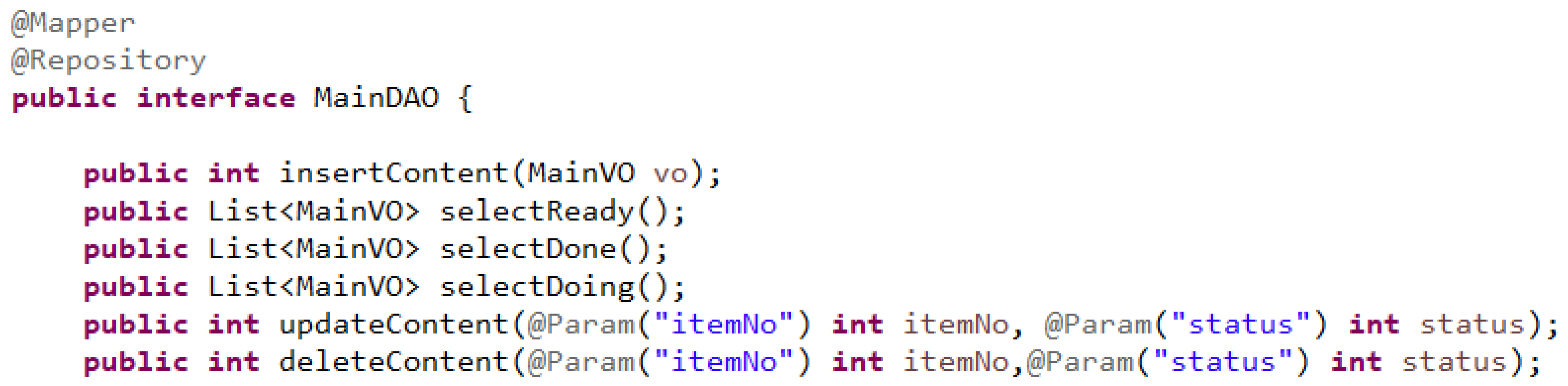
문제4. 기능 구현

MainService

```java
@Service
public class MainService {

    @Autowired
    private MainDAO dao;

    @Transactional
    public Map<Integer, List<MainVO>> selectAll() {
        List<MainVO> ready = dao.selectReady();
        List<MainVO> doing = dao.selectDoing();
        List<MainVO> done = dao.selectDone();

        Map<Integer, List<MainVO>> map = new HashMap<>();
        map.put(1, ready);
        map.put(2, doing);
        map.put(3, done);

        return map;
    }

    public int insertContent(MainVO vo) {
        return dao.insertContent(vo);
    }

    public int updateContent(int itemNo, int status) {
        return dao.updateContent(itemNo, status);
    }

    public int deleteContent(int itemNo, int status) {
        return dao.deleteContent(itemNo, status);
    }

}
```

문제4. 기능 구현

MainDAO



```
@Mapper
@Repository
public interface MainDAO {

    public int insertContent(MainVO vo);
    public List<MainVO> selectReady();
    public List<MainVO> selectDone();
    public List<MainVO> selectDoing();
    public int updateContent(@Param("itemNo") int itemNo, @Param("status") int status);
    public int deleteContent(@Param("itemNo") int itemNo,@Param("status") int status);
```

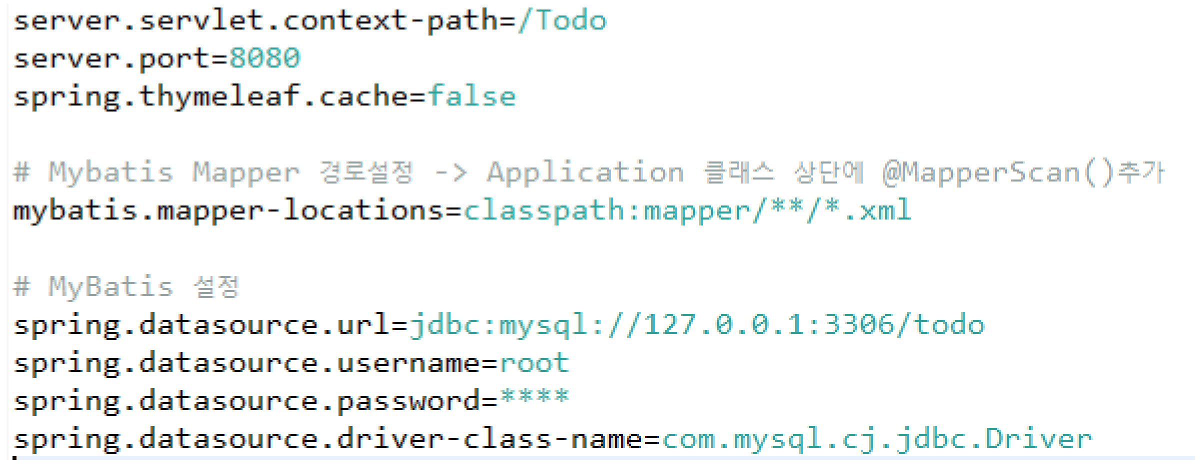
문제4. 기능 구현

main.xml

```
<mapper namespace="kr.co.todo.dao.MainDAO">
    <insert id="insertContent" parameterType="kr.co.todo.vo.MainVO"
            useGeneratedKeys="true" keyProperty="itemNo" keyColumn="itemNo">
        insert into `ready` set `content`=#{content}, `rdate`=NOW(), `status`=#{status};
    </insert>
    <select id="selectReady" resultType="kr.co.todo.vo.MainVO">
        select * from `ready` where `status`=1;
    </select>
    <select id="selectDone" resultType="kr.co.todo.vo.MainVO">
        select * from `ready` where `status`=3;
    </select>
    <select id="selectDoing" resultType="kr.co.todo.vo.MainVO">
        select * from `ready` where `status`=2;
    </select>
    <update id="updateContent">
        update `ready` set `status`=#{status} where `itemNo`=#{itemNo};
    </update>
    <delete id="deleteContent">
        delete from `ready` where `itemNo`=#{itemNo} and `status`=#{status};
    </delete>
</mapper>
```

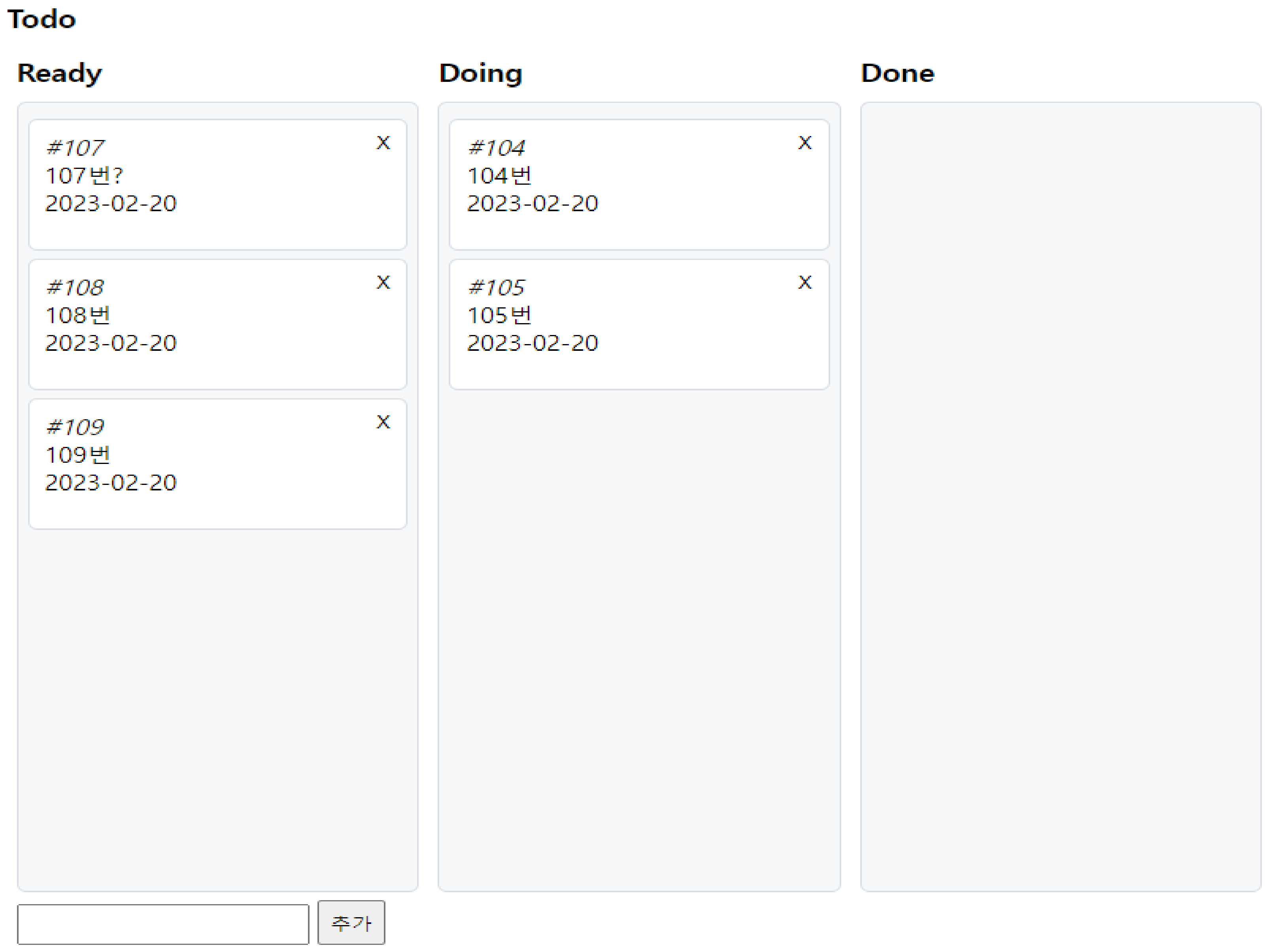
문제4. 기능 구현

application.properties



```
server.servlet.context-path=/Todo
server.port=8080
spring.thymeleaf.cache=false

# Mybatis Mapper 경로설정 -> Application 클래스 상단에 @MapperScan()추가
mybatis.mapper-locations=classpath:mapper/**/*.xml

# MyBatis 설정
spring.datasource.url=jdbc:mysql://127.0.0.1:3306/todo
spring.datasource.username=root
spring.datasource.password=****
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
```

문제4. 기능 구현



Todo

Ready

#107
107번?
2023-02-20
X

#108
108번
2023-02-20
X

#109
109번
2023-02-20
X

Doing

#104
104번
2023-02-20
X

#105
105번
2023-02-20
X

Done

이상으로 보고서를 마치겠습니다. 감사합니다.