

게시판 테이블 생성

- ▶ HeideSQL 이나 Workbench 를 이용해서 직접 생성하는 방법과 스크립트를 로딩하는 방법이 있다.
- ▶ SQL 스크립트

```
CREATE TABLE `t_board_2` (  
  `board_idx` INT(11) NOT NULL AUTO_INCREMENT COMMENT '글번호',  
  `title` VARCHAR(300) NOT NULL COMMENT '제목' COLLATE 'utf8_general_ci',  
  `contents` MEDIUMTEXT NULL DEFAULT NULL COMMENT '내용' COLLATE 'utf8_general_ci',  
  `hit_cnt` INT(11) NULL DEFAULT NULL COMMENT '조회수',  
  `created_datetime` DATETIME NULL DEFAULT NULL COMMENT '작성시간',  
  `creator_id` VARCHAR(50) NULL DEFAULT NULL COMMENT '생성자' COLLATE 'utf8_general_ci',  
  `updated_datetime` DATETIME NULL DEFAULT NULL COMMENT '수정시간',  
  `updater_id` VARCHAR(50) NULL DEFAULT NULL COMMENT '수정자' COLLATE 'utf8_general_ci',  
  `deleted_yn` CHAR(1) NULL DEFAULT NULL COMMENT '삭제여부' COLLATE 'utf8_general_ci',  
  PRIMARY KEY (`board_idx`) USING BTREE,  
  INDEX `title` (`title`) USING BTREE  
)  
COMMENT='게시판'  
COLLATE= 'utf8_general_ci'  
ENGINE=InnoDB  
AUTO_INCREMENT=1
```

마이바티스 설정

/board/src/main/resources/application.properties

← 컬럼명을 카멜 표기법으로 변경

Mybatis.configuration.map-underscore-to-camel-case=true

/board/src/main/java/kr/ac/inha/configuration/DatabaseConfiguration.java

@Bean

@ConfigurationProperties(prefix="mybatis.configuration")

← application.properties 읽어 오는 부분

```
public org.apache.ibatis.session.Configuration mybatisConfig(){  
    return new org.apache.ibatis.session.Configuration();  
}
```

@Bean

```
public SqlSessionFactory sqlSessionFactory(DataSource dataSource) throws Exception {
```

```
    SqlSessionFactoryBean sqlSessionFactoryBean = new SqlSessionFactoryBean();
```

```
    sqlSessionFactoryBean.setDataSource(dataSource);
```

```
    sqlSessionFactoryBean.setMapperLocations(applicationContext.getResources("classpath:/mapper/**/*.xml"));
```

```
    sqlSessionFactoryBean.setConfiguration(mybatisConfig());
```

← 추가(환경설정 적용)

```
    return sqlSessionFactoryBean.getObject();
```

```
}
```

DTO 만들기

- ▶ Dto 의 변수는 기본적으로 카멜 표기법을 사용한다.

```
/board/src/main/java/kr/ac/inha/board/board/dto/BoardDto.java
```

```
package kr.ac.inha.board.board.dto;
```

```
import lombok.Data;
```

```
@Data
```

```
public class BoardDto {
```

```
    private int boardIdx;
```

```
    private String title;
```

```
    private String contents;
```

```
    private int hitCnt;
```

```
    private String createdDatetime;
```

```
    private String creatorId;
```

```
    private String updateDatetime;
```

```
    private String updaterId;
```

```
    private String deletedYn;
```

```
}
```

테이블당 하나씩 생성

변수명은 카멜 표기법 사용

1. 카멜 표기법

- "camelCase"
- "단봉낙타" 표기법^{[1][2]}
- 각 단어의 첫문자를 대문자로 표기하고 붙여쓰되, 맨처음 문자는 소문자로 표기함
- 띄어쓰기 대신 대문자로 단어를 구분하는 표기 방식
- 예시: backgroundColor, typeName, iPhone

2. 파스칼 표기법

- "PascalCase"
- 첫 단어를 대문자로 시작하는 표기법
- 예시: BackgroundColor, TypeName, PowerPoint

Controller 작성

```
/board/src/main/java/kr/ac/inha/board/board/controller/BoardController.java  
package kr.ac.inha.board.board.controller;
```

```
import java.util.List;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.stereotype.Controller;  
import org.springframework.web.bind.annotation.RequestMapping;  
import org.springframework.web.servlet.ModelAndView;  
  
import kr.ac.inha.board.board.dto.BoardDto;
```

<CTRL>+<SHIFT>+"O" 하면 import 문 자동생성됨

@Controller

```
public class BoardController {  
    @Autowired  
    private BoardService boardService;  
  
    @RequestMapping("/board/openBoardList.do")  
    public ModelAndView openBoardList() throws Exception {  
        ModelAndView mv = new ModelAndView("/board/boardList");  
  
        List<BoardDto> list = boardService.selectBoardList();  
        mv.addObject("list", list);  
  
        return mv;  
    }  
}
```

Service 작성

```
/board/src/main/java/kr/ac/inha/board/board/service/BoardService.java
```

```
package com.qhedge.online.board.board.service;
```

```
...
```

```
public interface BoardService {  
    List<BoardDto> selectBoardList() throws Exception;  
}
```

```
/board/src/main/java/kr/ac/inha/board/board/service/BoardServiceImpl.java
```

```
package com.qhedge.online.board.board.service;
```

```
...
```

```
@Service  
public class BoardServiceImpl implements BoardService{  
    @Autowired  
    private BoardMapper boardMapper;  
  
    @Override  
    public List<BoardDto> selectBoardList() throws Exception {  
        return boardMapper.selectBoardList();  
    }  
}
```

Mapper 작성

```
/board/src/main/java/kr/ac/inha/board/board/mapper/BoardMapper.java
```

```
package com.qhedge.online.board.board.mapper;
```

```
import java.util.List;
```

```
import org.apache.ibatis.annotations.Mapper;
```

```
import com.qhedge.online.board.board.dto.BoardDto;
```

```
@Mapper
```

```
public interface BoardMapper {
```

```
    List<BoardDto> selectBoardList() throws Exception;
```

```
}
```

SQL 작성

/board/src/main/resources/mapper/sql-board.xml

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
```

```
<mapper namespace="kr.ac.inha.board.board.mapper.BoardMapper">
```

```
<select id="selectBoardList" resultType="kr.ac.inha.board.board.dto.BoardDto">
```

```
<![CDATA[
```

```
SELECT
```

```
    board_idx,
```

```
    title,
```

```
    hit_cnt,
```

```
    DATE_FORMAT(created_datetime, '%Y-%m-%d %H:%i:%s') AS created_datetime
```

```
FROM
```

```
    t_board
```

```
WHERE
```

```
    deleted_yn = 'N'
```

```
ORDER BY board_idx DESC
```

```
]]>
```

```
</select>
```

```
</mapper>
```

구분기호	역할	구분기호	역할
%Y	4자리 년도	%m	숫자 월 (두자리)
%y	2자리 년도	%c	숫자 월(한자리는 한자리)
%M	긴 월(영문)	%d	일자 (두자리)
%b	짧은 월(영문)	%e	일자(한자리는 한자리)
%W	긴 요일 이름(영문)	%I	시간 (12시간)
%a	짧은 요일 이름(영문)	%H	시간(24시간)
%i	분	%r	hh:mm:ss AM,PM
%T	hh:mm:ss	%S	초