

2025년 상반기 K-디지털 트레이닝

# 영속,비즈니스 계층의 CRUD 구현

[KB] IT's Your Life

### ✓ 다음과 같이 프로젝트를 만드세요.

- template: MyBatisSpringLegacy
- project name: board
  
- settings.gradle에서 프로젝트명 변경
- Tomcat 설정

### ✓ 게시판용 테이블을 생성하고, 5개의 테스트 데이터를 추가하세요.

#### ○ 컬럼

- |               |     |
|---------------|-----|
| ▪ no          | PK  |
| ▪ title       | 제목  |
| ▪ content     | 본문  |
| ▪ writer      | 작성자 |
| ▪ reg_date    | 등록일 |
| ▪ update_date | 수정일 |

- ✓ tbl\_board 테이블을 위한 BoardVO 클래스를 정의하세요.
  - 패키지: org.scoula.board.domain

- ✓ **MyBatis 설정 파일에 mapUnderscoreToCamelCase 속성과 BoardVO에 대한 타입 앨리어스 속성을 설정하세요.**

### ✓ MyBatis 연동을 위한 기본 구조를 추가하세요.

- BoardMapper 인터페이스
  - 패키지: org.scoula.board.mapper
- RootConfig에 Mapper 스캔 정보 등록
- BoardMapper.xml 파일

- ✓ 게시판의 목록을 얻는 getList() 메서드를 인터페이스와 xml 매퍼에 추가하고, 단위 테스트를 진행하세요.

- ✓ 게시판의 글 하나를 얻는 `get(Long no)` 메서드를 인터페이스와 xml 매퍼에 추가하고, 단위 테스트를 진행하세요.



- ✓ 게시판의 글을 추가하는 `int create(BoardVO board)` 메서드를 인터페이스와 xml 매퍼에 추가하고, 단위 테스트를 진행하세요.
  - 글을 추가할 때 생성된 PK를 전달한 객체에 설정함

- ✓ 게시판의 글을 수정 하는 `int update(BoardVO board)` 메서드를 인터페이스와 xml 매퍼에 추가 하고, 단위 테스트를 진행하세요.
  - 수정항목: title, content, writer
  - `update_date`는 쿼리 실행한 날짜/시간으로 자동 설정

- ✓ 게시판의 글을 삭제하는 `int delete(Long no)` 메서드를 인터페이스와 xml 매퍼에 추가하고, 단위 테스트를 진행하세요.

- ✓ 지금까지 작성한 Mapper.xml을 재사용할 수 있도록 수정하여 파일 템플릿으로 저장하세요.