

서버 CRUD 개발 가이드

게시판 테이블(BOARD)을 이용해서 조회, 등록, 수정, 삭제하는 서버로직 개발 방법을 설명합니다.

실습을 위한 개발은 SAMPLE 프로젝트에 작성합니다.

SAMPLE 프로젝트를 구성하기 위한 프로젝트도 함께 이클립스에 Import 합니다.

- 프로젝트 리스트

프로젝트명	프로젝트 내용
SAMPLE	샘플 프로젝트
COM	공통 프로젝트
CONFIG	로컬 개발환경 설정 프로젝트
LIB	참조 라이브러리 프로젝트

- 서버로직 개발 순서 (일반)

- 1) DAO클래스 생성
- 2) DAO편집기를 이용하여 SQL작성 (DVO자동생성)
- 3) SVO클래스 생성 (완료 후 [UI 정의보기] 메뉴선택)
- 4) BVO클래스 생성
- 5) Biz클래스 생성
- 6) Svc인터페이스 생성 ([Healthcare Solution]-[Service Meta Manager] 메뉴를 선택하여 ServiceID를 저장)
- 7) Svc인터페이스를 구현한 SvcImpl클래스 생성
- 8) JUnit클래스 생성 (Option)

- 서버로직 개발 순서 (CommonService 이용)

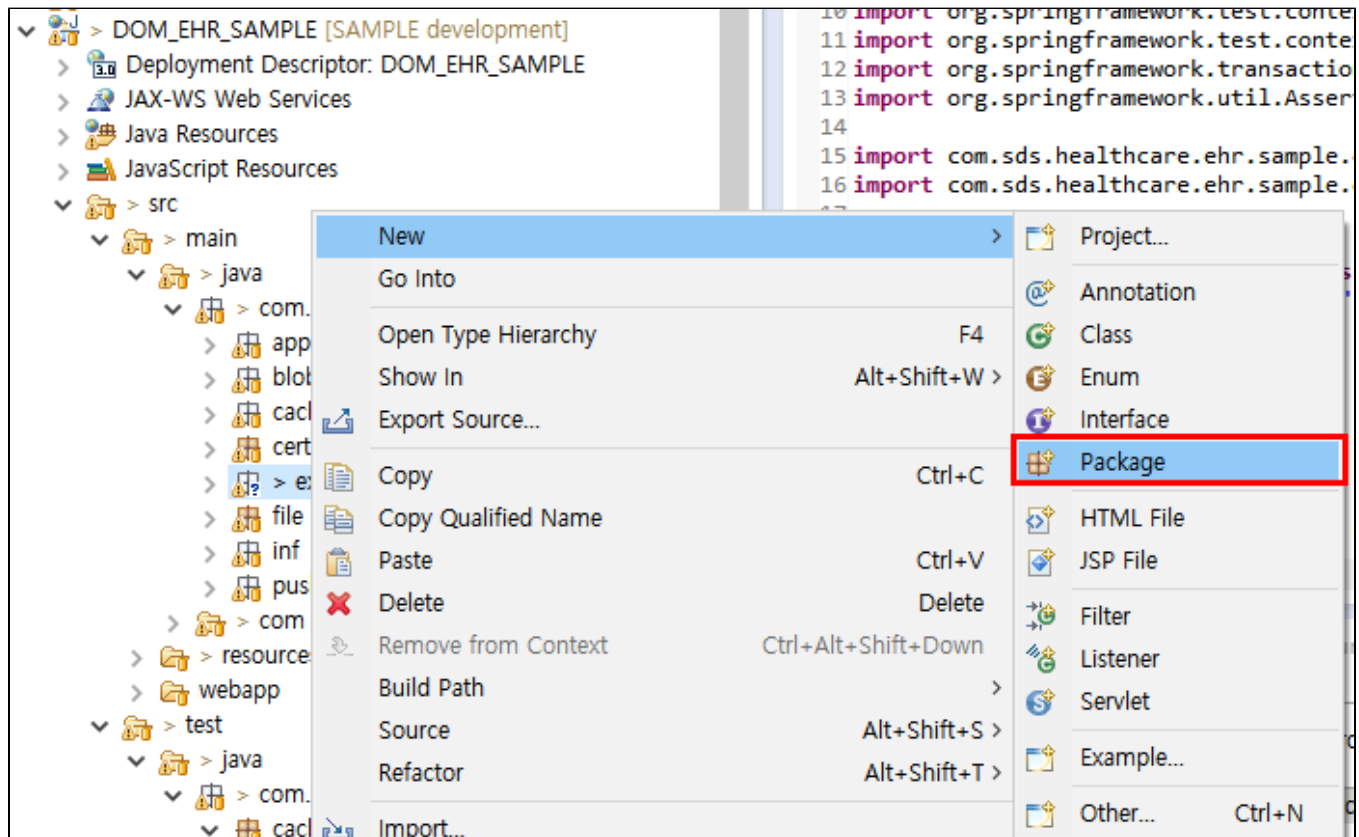
- 1) DAO클래스 생성
- 2) DAO편집기를 이용하여 SQL 작성 (DVO자동생성)
- 3) CommonService 등록
- 4) JUnit클래스 생성 (Option)

1. 조회 (다건)

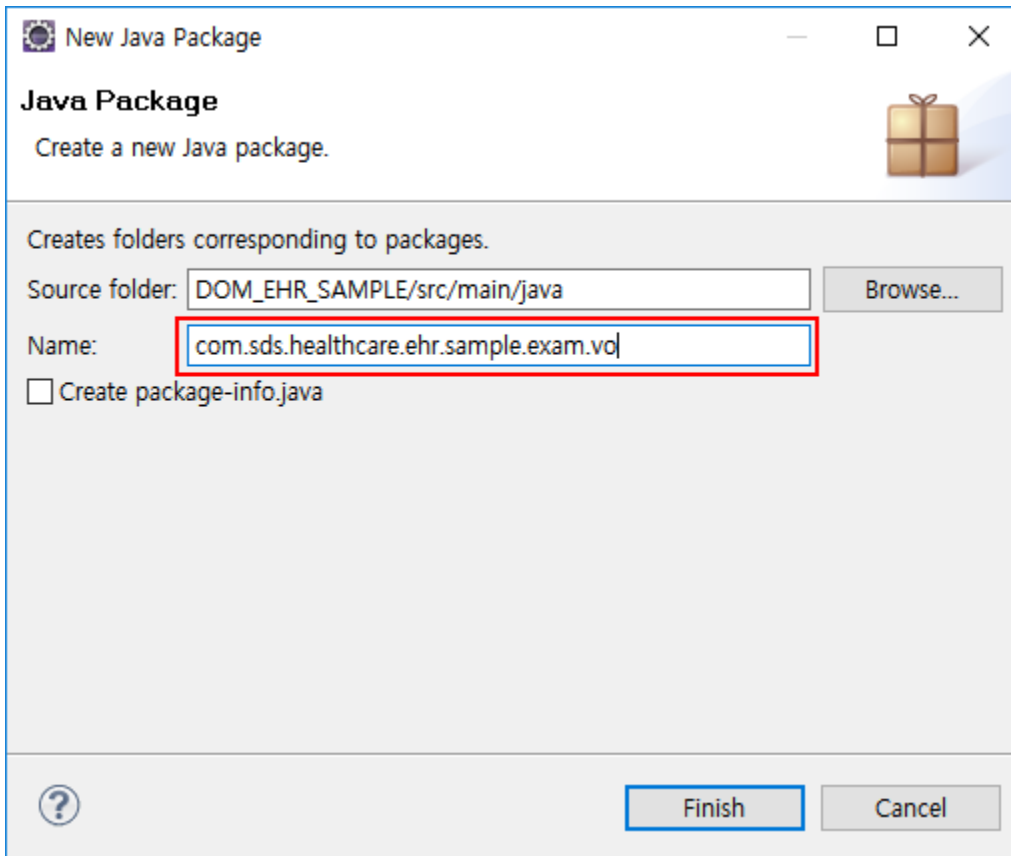
1) 패키지 추가

SAMPLE 프로젝트의 패키지를 추가.

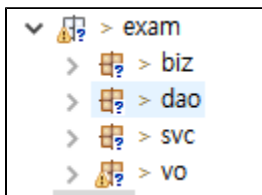
기본적으로 업무 최하위 패키지당 4개의 하위 패키지를 생성. (svc, biz, dao, vo)



업무 패키지, 클래스명, 메소드명 등 명명규칙 관련해서는 개발표준가이드를 참조합니다.

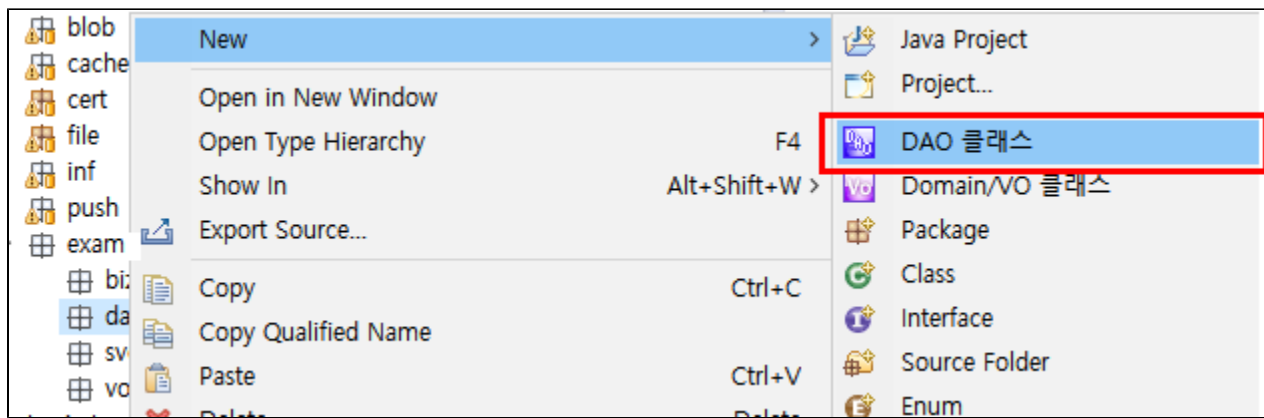


아래는 생성이 완료된 패키지 구조.



2) DAO 클래스 생성

com.sds.healthcare.ehr.sample.exam.dao 패키지를 선택 한 후 아래와 같이 DAO 클래스 메뉴를 선택.



DAO 이름을 입력한다. ExamBoardDAO 를 입력.

이름과 클래스 설명은 옵션항목으로 알맞은 내용으로 입력한다. 소스 생성시 Comment로 생성.

신규 DAO 클래스

신규 DAO 클래스
신규 DAO 클래스를 생성하세요.

Source folder: DOM_EHR_SAMPLE/src/main/java Browse...

Package: com.sds.healthcare.ehr.sample.exam.dao Browse...

Name: ExamBoardDAO

Superclass: com.anyframe.online.runtime.orm.mybatis.AbstractSqlSe Browse...

이름(한국어): 게시판 관리

클래스 설명

게시판의 내용을 조회, 등록, 수정, 삭제 처리한다.

?

< Back

Next >

Finish

Cancel

정상적으로 DAO 클래스가 생성된 경우에는 아래와 같은 DAO편집기 화면을 확인할 수 있습니다.

ExamBoardDAO.java

클래스 정보

선택된 메소드의 프라퍼티를 설정하세요.

클래스 명: ExamBoardDAO

패키지 명*: com.sds.healthcare.ehr.sample.exam.dao

클래스 [한글명]: 게시판 관리

클래스 설명: 게시판의 내용을 조회, 등록, 수정, 삭제 처리한다.

연결 프로파일: EHRV2.3

매핑 XML: [src/main/resources/sqlmap/oracle/com/sds/healthcare/ehr/sample/exam/dao/mapper-ExamBoardDAO.xml](#)

메소드 정보

DAO 클래스의 메소드(단, dmlType 메소드) 목록을 지정하세요.

3) DAO 메소스 추가

다건 조회이므로 추가버튼을 클릭한 후 list 항목을 선택.

클래스 정보

메소드 정보

DAO 클래스의 메소드(단, dmlType 메소드) 목록을 지정하세요.

추가....

제거

위로

아래로

CRUD 생성

DML 타입

DML 타입

insert

update

delete

selectOne

list

executeBatch

paging

execute

메소드 세부정보 탭을 클릭해서 확장한 후 메소드 이름을 입력한다. listBoard를 입력한 후 저장.

메소드 정보 및 SQL 아이디는 메소드명에 따라 자동으로 변경되므로 해당 항목정보를 변경하기 위해서는 메소드명을 변경.

클래스 정보
선택된 메소드의 프라퍼티를 설정하세요.

클래스명 ExamBoardDAO
패키지명 com.sds.healthcare.ehr.sample.exam.dao
클래스 [한글명] 게시판 관리
클래스 설명 게시판의 내용을 조회, 등록, 수정, 삭제 처리한다.
연결 프로파일 EHRV2.3
매핑 XML src/main/resources/sqlmap/oracle/com/sds/healthcare/ehr/sample/exam/dao/mapper-ExamBoardDAO.xml

메소드 정보
DAO 클래스의 메소드(단, dmlType 메소드) 목록을 지정하세요.

listBoard

메소드 세부정보
선택된 메소드의 프라퍼티를 설정하세요. 필수 필드는 "*"로 표시됩니다.

메소드명 listBoard
메소드 타입 list
입력 타입 VO com.sds.healthcare.ehr.sample.exam.vo.ExamBoardDVO
반환 타입 List com.sds.healthcare.ehr.sample.exam.vo.ExamBoardDVO
메소드 [한글명]
메소드 설명

SQL 문
선택된 메소드의 SQL 문을 나타냅니다. 필수 필드는 "*"로 표시됩니다.

SQL 아이디 com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-listBoard
/*SQL_ID: com-sds-healthcare-ehr-sample-exam-dao-TstExamBoradDAO-listBoard */

4) SQL 쿼리 작성

쿼리를 입력하기 위해 SQL의 편집 버튼을 클릭해서 SQL편집 마법사를 오픈.

SQL 편집 마법사

SQL 편집 마법사 페이지
SQL 문을 편집하고 결과 확인을 위해 SQL문을 실행하세요.

SQL 아이디 com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-listBoard 메소드 타입 list 연결 프로파일 EHRV2.3

실행 테스트

테이블 정보
테이블명 테이블 타입 스키마 주석

Column Index
이름 타입 길이 Scale PK 인덱스 주석

테스트 실행
테스트 값 테스트 실행 결과 계획 예러

번호 이름 Java 타입 테스트 값

추가...
제거
위로
아래로
값 추출
값 세팅

Finish Cancel

쿼리 입력창에 쿼리를 입력.

제목과 내용에 대한 검색조건을 다이나믹 쿼리로 입력.

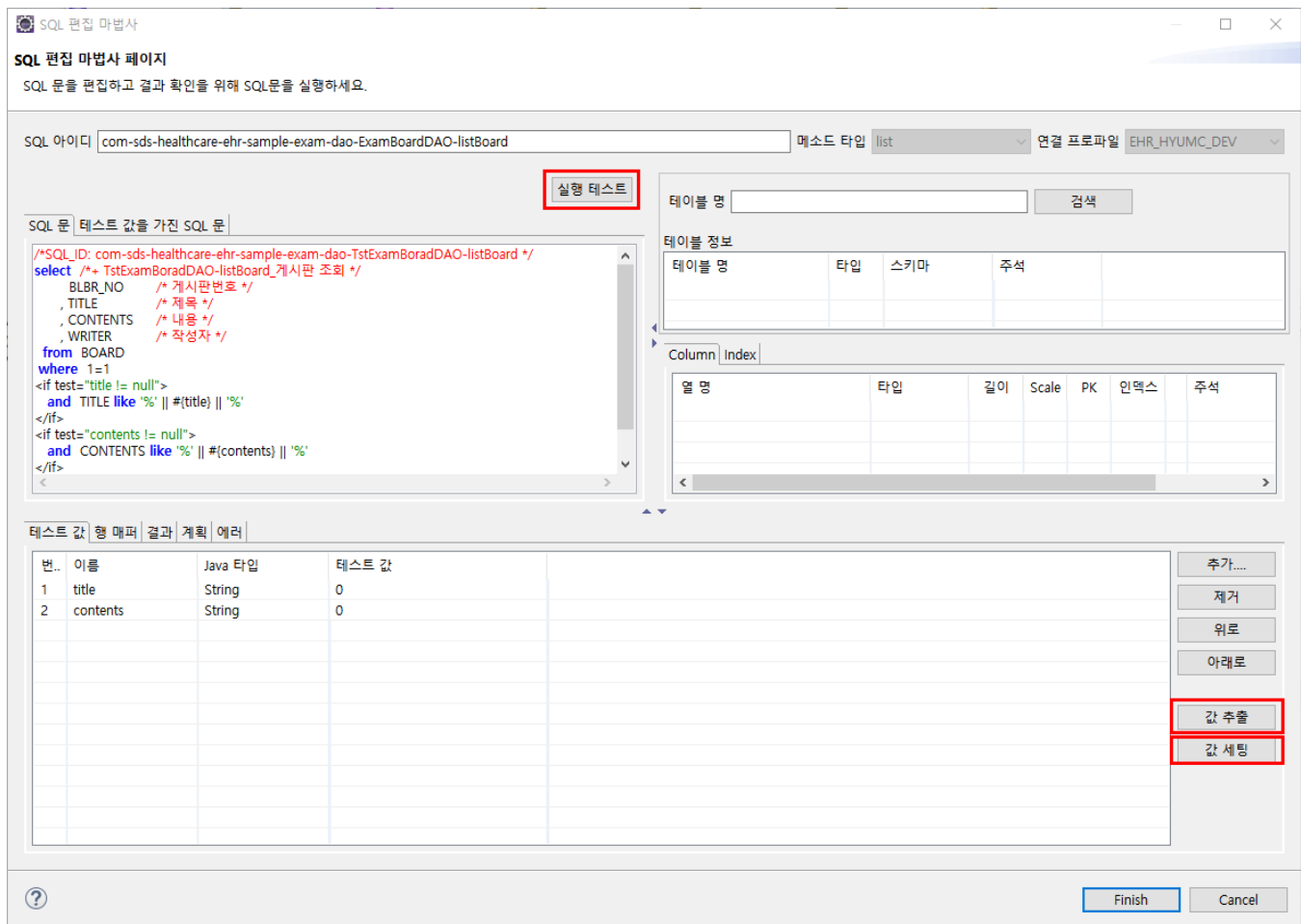
게시판 목록조회쿼리

```
/*SQL_ID: com-sds-healthcare-ehr-sample-exam-dao-ExamBoradDAO-listBoard */
select /*+ TstExamBoradDAO-listBoard_ */
      BLBR_NO      /* */
    , TITLE        /* */
    , CONTENTS     /* */
    , WRITER       /* */
  from BOARD
 where l=1
<if test='title != null'>
    and TITLE like '%' || #{title} || '%'
</if>
<if test='contents != null'>
    and CONTENTS like '%' || #{contents} || '%'
</if>
 order
    by BLBR_NO DESC
```

값 추출 → 값 세팅 → 실행 테스트 순으로 버튼을 클릭.

값 추출은 변수에 대한 바인딩 변수 이름을 추출하는 이벤트이고 값 세팅은 추출된 변수이름에 디폴트 값을 설정하는 이벤트입니다.

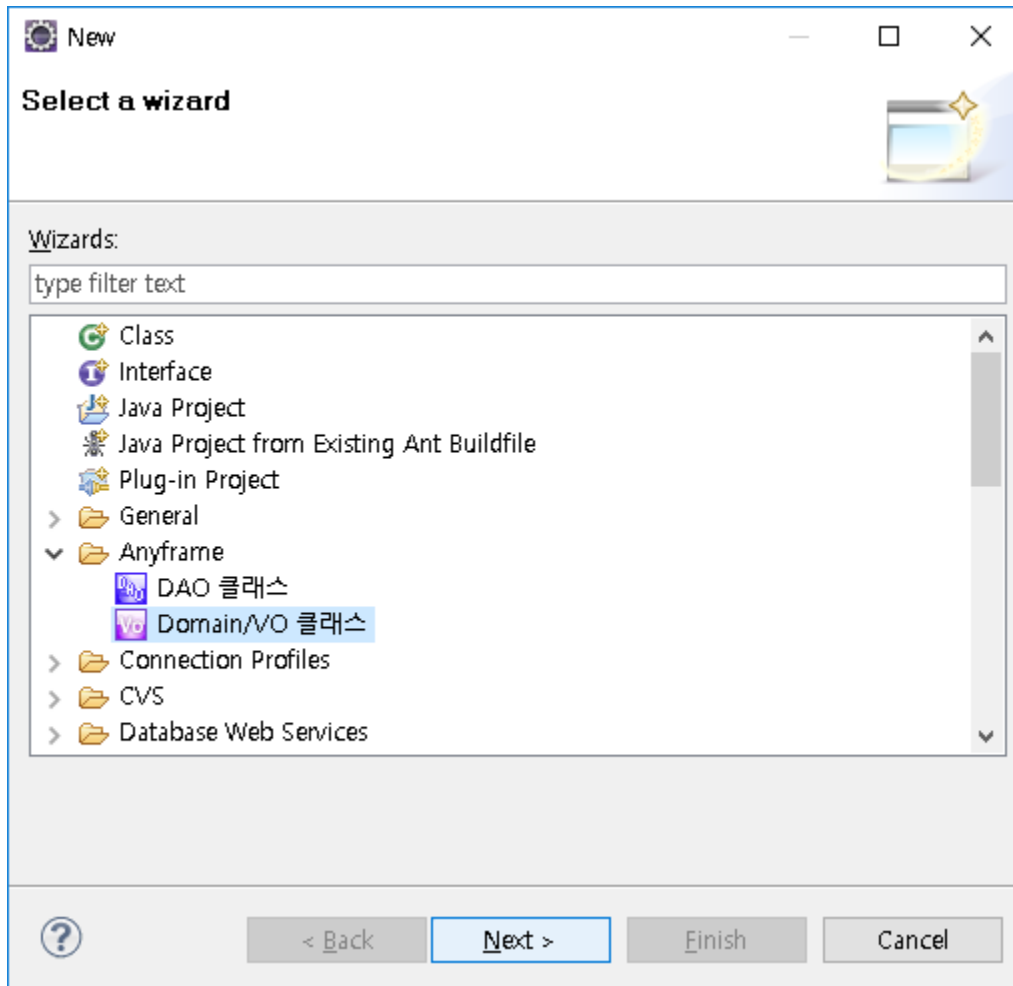
마지막으로 실행 테스트는 입력한 변수 값을 바인딩해서 쿼리를 실행해서 정상적인 쿼리인지를 판단하는 이벤트입니다.




정상적으로 실행된 화면입니다. 테스트 값을 변경해가면서 확인해 본 후 Finish버튼을 클릭.

com.sds.healthcare.ehr.sample.exam.vo패키지를 선택 한 후 Ctrl+N 을 입력.

Domain/VO 클래스를 선택.




이름에 ExamBoardSVO를 입력한 후 Finish 버튼을 클릭.


신규 Domain/VO 클래스 도메인

신규 **Domain/VO** 클래스

신규 Domain/VO 클래스를 생성하세요.



Source folder:


Package:

Name:

Superclass:

이름(한국어):

클래스 설명 :



삽입 메뉴를 통해 아래와 같이 VO항목을 추가.

title, contents, blbrNo 입력값 필드와 DVO List 출력값 필드를 정의 합니다.

출력 필드인 DVOList는 참조 클래스 컬럼의 버튼을 클릭해서 ExamBoardDVO클래스를 검색한 후 선택하여 입력.

클래스 정보

클래스 명

ExamBoardSVO

클래스 [한글명]

패키지 명

com.sds.healthcare.ehr.sample.exam.vo

입출력 타입

VO

클래스 설명

VO 편집

이름	한글명	타입	참조 클래스
title		String	
contents		String	
blbrNo	게시판번호	BigDecimal	
> examBoardDVOList	게시판 관리DVO	List	com.sds.healthcare.ehr.sample.exam.vo.ExamBoardDVO

삽입

복사

붙여넣기

삭제

위로 이동

아래로 이동

신규 행 삽입

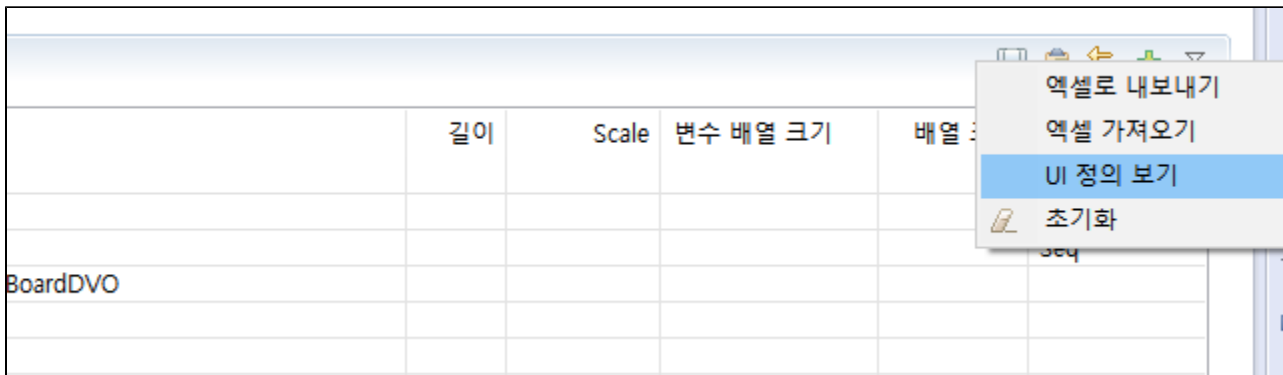
삽입할 행의 개수: 1

OK

Cancel

SVO 메타 정보 저장을 위해 아래와 같이 [\[UI 정의 보기\]](#) 메뉴를 선택.

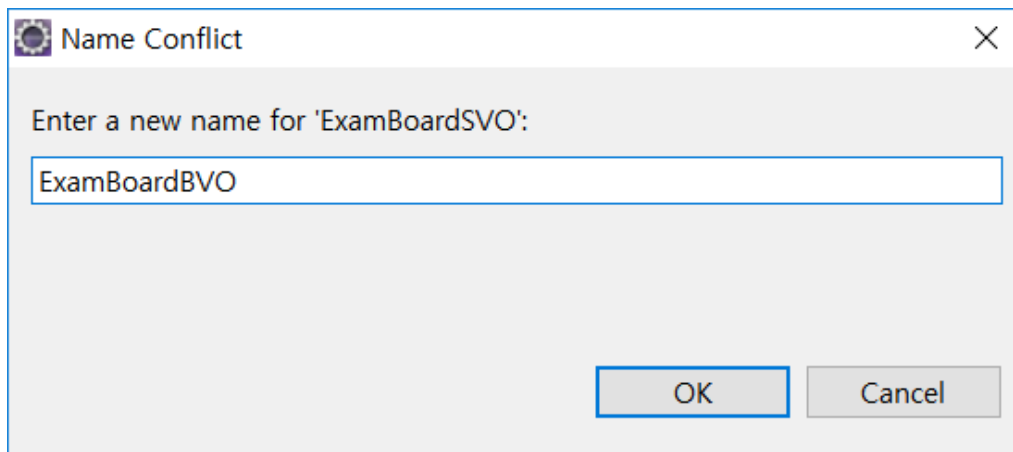
클라이언트 프로그램에서 해당 메타정보를 이용하여 필드를 정의하므로 추가, 변경, 삭제시 반드시 해야하는 작업입니다.



6) BVO 클래스 생성

생성된 SVO클래스를 복사해서 ExamBoardBVO 클래스를 생성합니다.

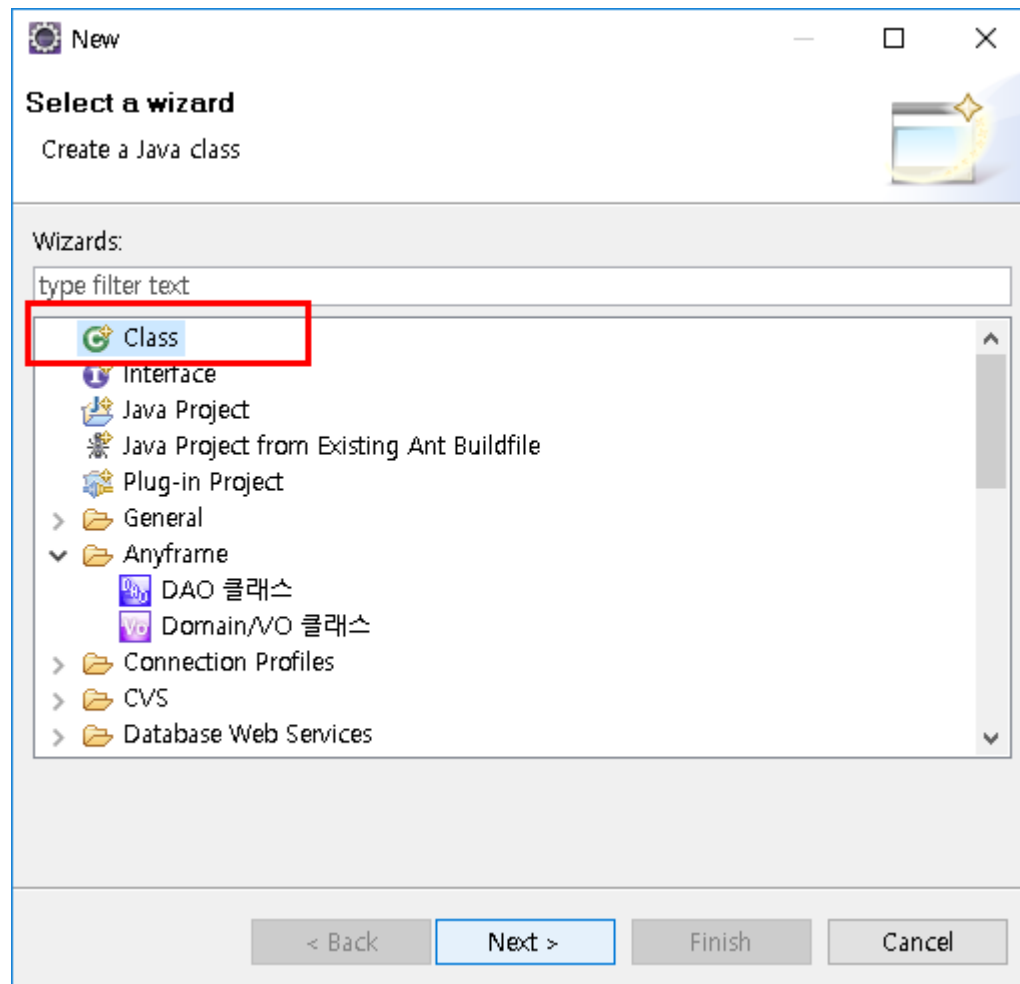
(SVO 클래스 선택하여 Ctrl+C, VO 패키지 선택하여 Ctrl+V 한 후, 아래와 같이 클래스 이름을 변경 한다.)



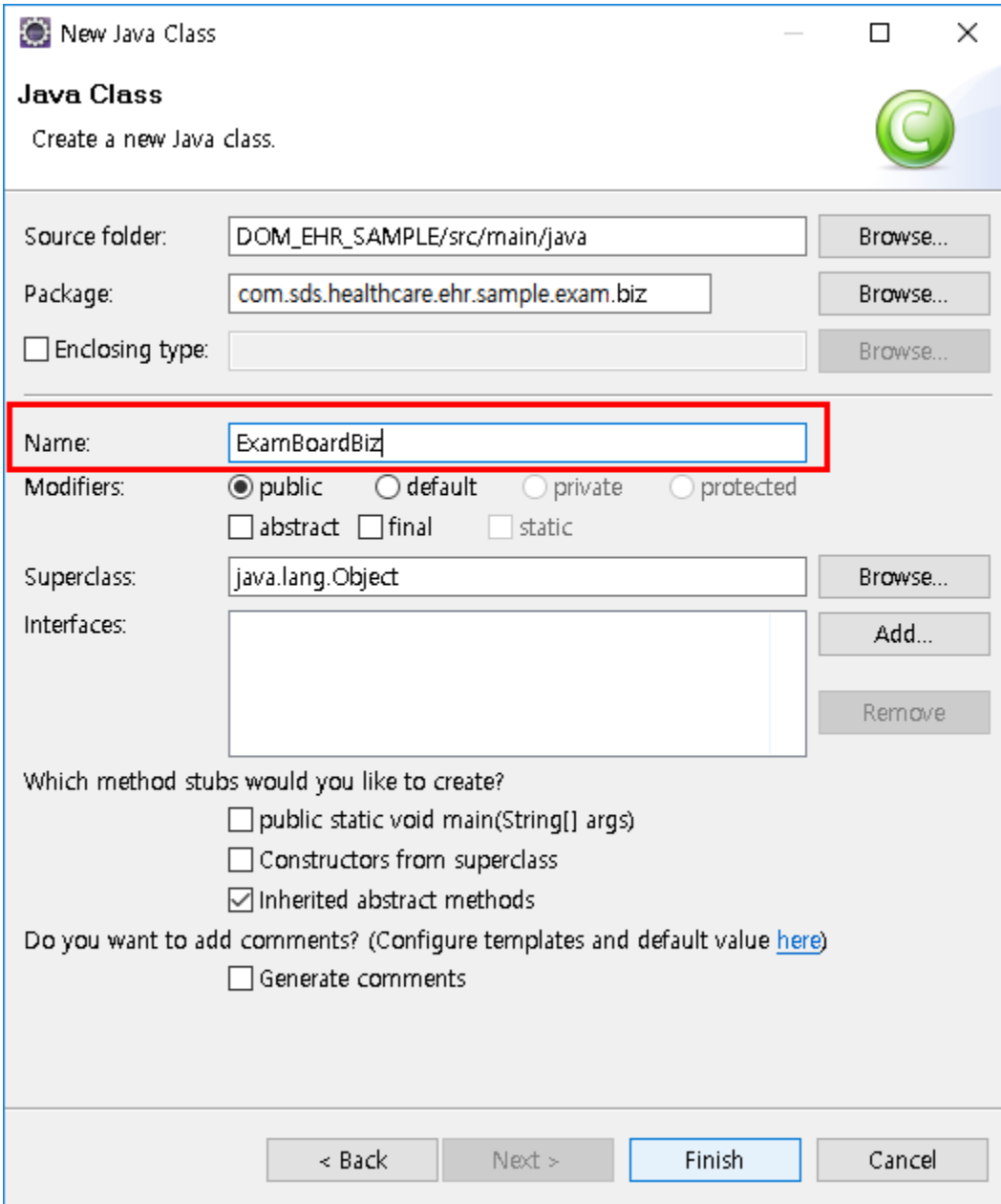
7) Biz 클래스 생성

com.sds.healthcare.ehr.sample.exam.biz 패키지를 선택 한 후 Ctrl+N 을 입력.

아래와 같이 Class를 선택.



이름에 ExamBoardBiz를 입력.



New Java Class

Java Class
Create a new Java class.

Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers: ☒ public ☐ default ☐ private ☐ protected
☐ abstract ☐ final ☐ static

Superclass:

Interfaces:

Which method stubs would you like to create?

☐ public static void main(String[] args)
☐ Constructors from superclass
☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))
☐ Generate comments

getBoardList 메소드를 추가하고 아래와 같이 코드를 입력합니다.

```

public class ExamBoardBiz {

    /**
     *
     * @param bvo ExamBoardBVO
     * @return ExamBoardBVO
     * @author
     */
    public ExamBoardBVO getBoardList(ExamBoardBVO bvo) {
        ExamBoardDVO dvo = new ExamBoardDVO();
        VoUtil.copy(bvo, dvo);
        bvo.setExamBoardDVOList(ExamBoardDAO.getInstance().listBoard(dvo));
        return bvo;
    }

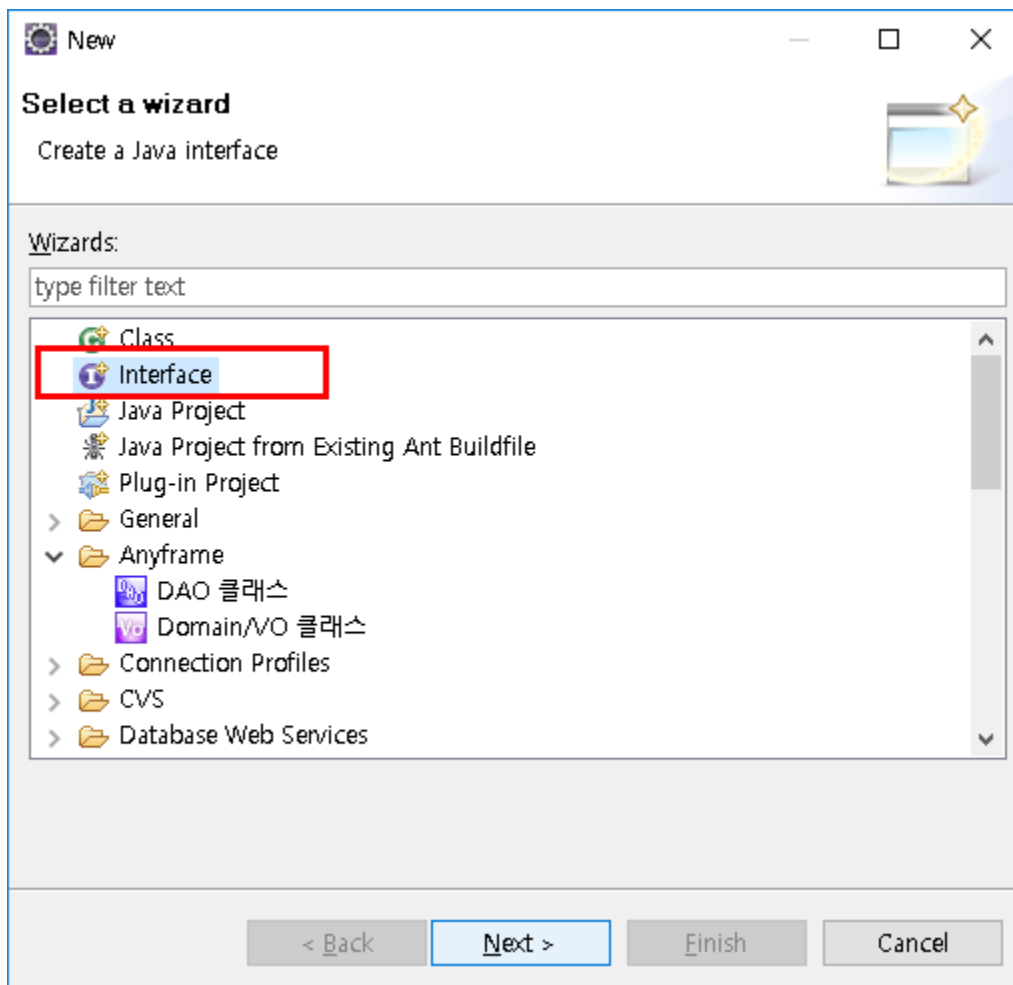
}

```

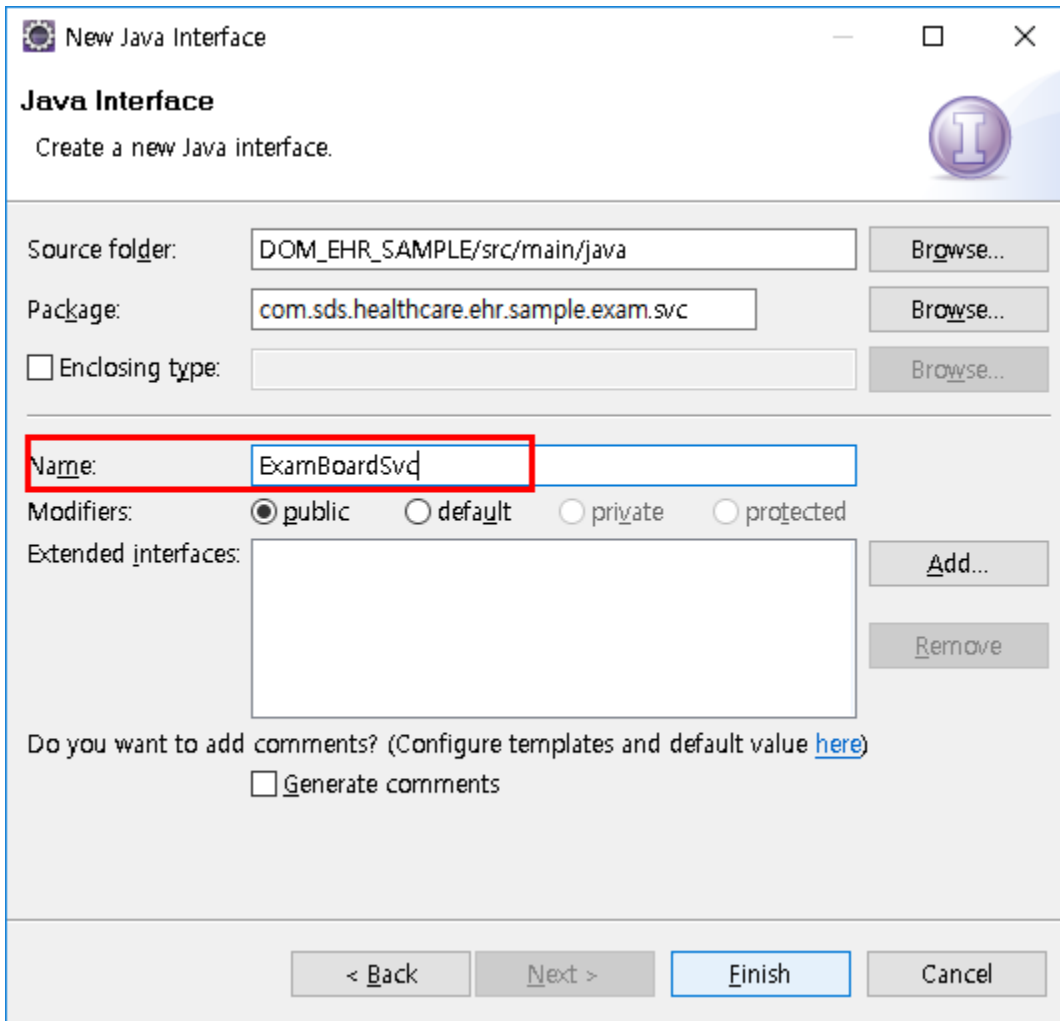
8) Svc 인터페이스 생성

com.sds.healthcare.ehr.sample.exam.svc 패키지를 선택 한 후 Ctrl+N 을 입력.

아래와 같이 Interface를 선택.



이름에 ExamBoardSvc를 입력.



New Java Interface

Java Interface

Create a new Java interface.

Source folder: DOM_EHR_SAMPLE/src/main/java Browse...

Package: com.sds.healthcare.ehr.sample.exam.svc Browse...

☐ Enclosing type: Browse...

Name: ExamBoardSvc

Modifiers: ☒ public ☐ default ☐ private ☐ protected

Extended interfaces: Add... Remove

Do you want to add comments? (Configure templates and default value [here](#))

☐ Generate comments

< Back Next > Finish Cancel

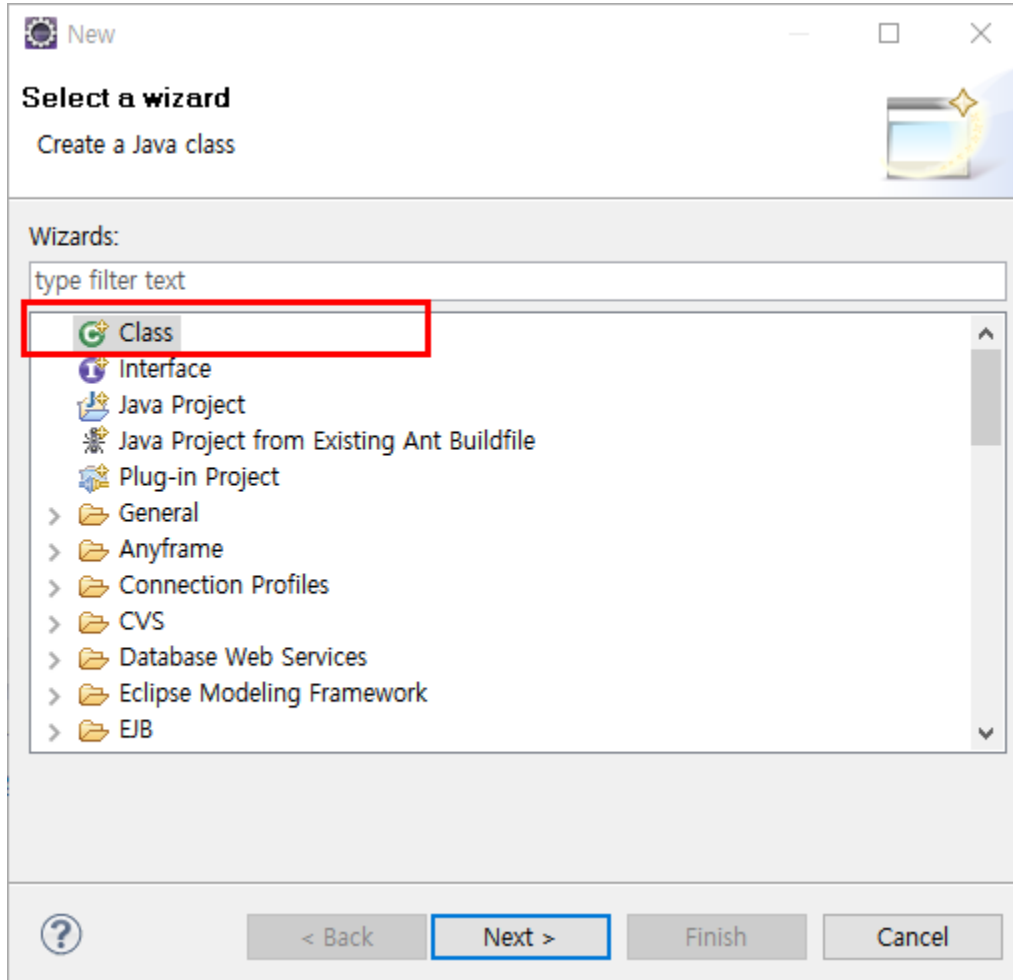
getBoardList 메소드를 추가하고 ServiceId를 ExmGetBoardListSID로 설정합니다.

```
public interface ExamBoardSvc {  
  
    /**  
     *  
     * @param svo ExamBoardSVO  
     * @return ExamBoardSVO  
     * @author  
     */  
    @ServiceIdMapping("ExmGetBoardListSID")  
    ExamBoardSVO getBoardList(ExamBoardSVO svo);  
}
```

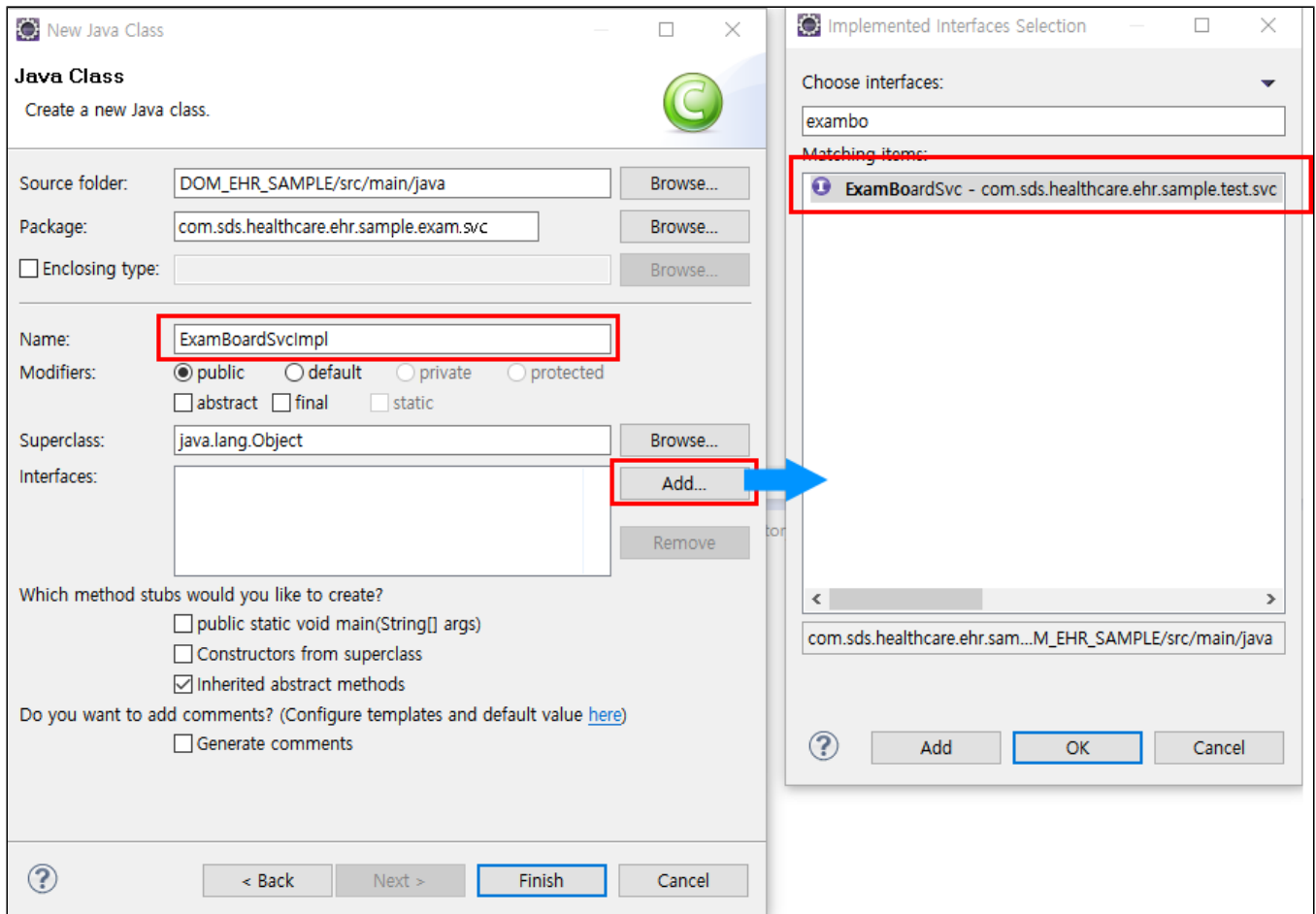

9) Svclmpl 클래스 생성

com.sds.healthcare.ehr.sample.exam.svc 패키지를 선택 한 후 Ctrl+N 을 입력.

아래와 같이 Class를 선택.



Name에 ExamBoardSvclmpl 을 입력하고 Interfaces 의 Add... 버튼을 클릭해서 ExamBoardSvc 인터페이스를 검색하여 선택합니다.



생성된 클래스 파일에 조희로직을 입력합니다.

```
@Service
public class ExamBoardSvcImpl implements ExamBoardSvc {

    @Override
    public ExamBoardSVO getBoardList(ExamBoardSVO svo) {
        ExamBoardBVO bvo = new ExamBoardBVO();
        VoUtil.copy(svo, bvo);

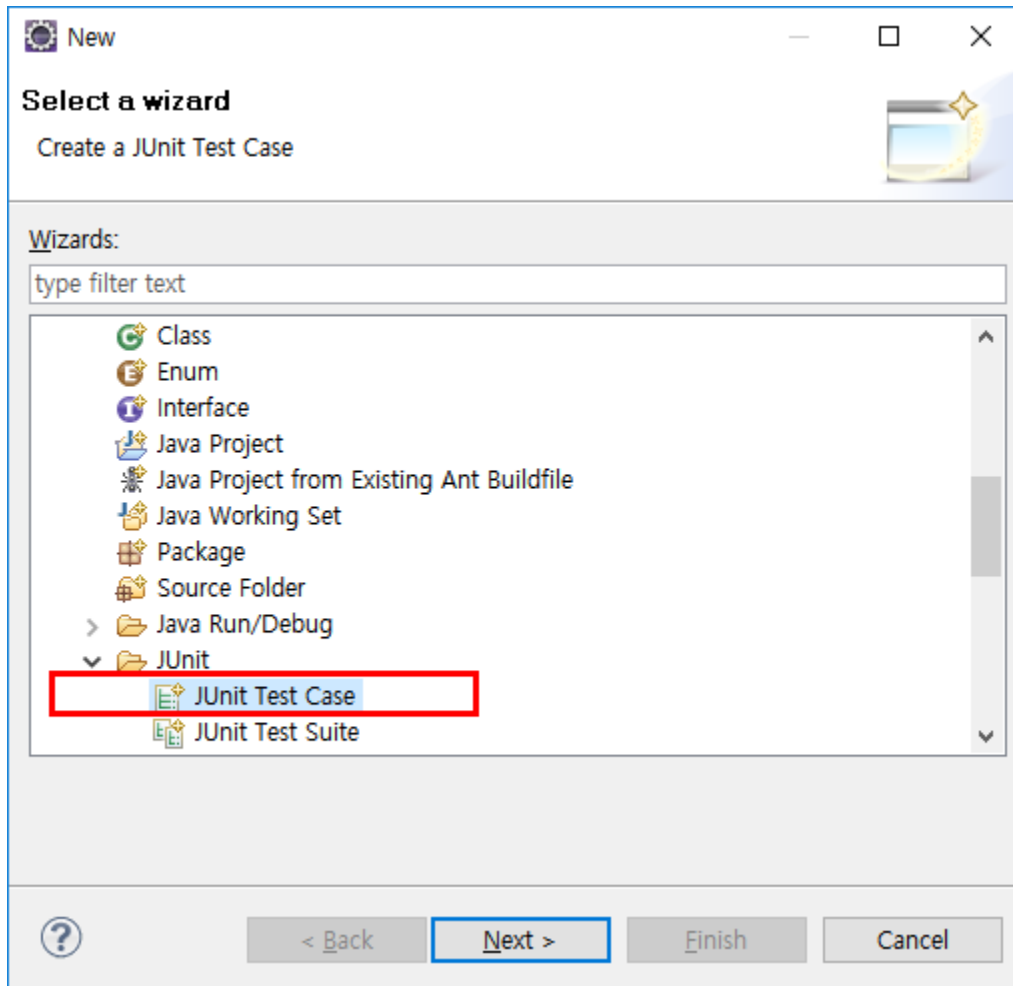
        ExamBoardBiz biz = new ExamBoardBiz();
        bvo = biz.getBoardList(bvo);
        VoUtil.copy(bvo, svo);

        return svo;
    }
}
```


10) JUnit 클래스 생성

src/test 소스 폴더에서 com.sds.healthcare.ehr.sample.exam패키지를 선택 한 후 Ctrl+N 을 입력.

아래와 같이 JUnit Test Case를 선택.




Name 에 ExamBoardSvcTest를 입력.

 New JUnit Test Case

JUnit Test Case

Select the name of the new JUnit test case. You have the options to specify the class under test and on the next page, to select methods to be tested.



☐ New JUnit 3 test

☒ New JUnit 4 test

Source folder:

DOM_EHR_SAMPLE/src/test/java

Browse...

Package:

com.sds.healthcare.ehr.sample.exam

Browse...

Name:

ExamBoardSvcTest

Superclass:

java.lang.Object

Browse...

Which method stubs would you like to create?

☐ setUpBeforeClass()

☐ tearDownAfterClass()

☐ setUp()

☐ tearDown()


☐ constructor

Do you want to add comments? (Configure templates and default value [here](#))

☐ Generate comments

Class under test:

Browse...



< Back

Next >

Finish

Cancel

생성된 클래스 파일에 조회 테스트 로직을 입력합니다.

```

@RunWith(SpringJUnit4ClassRunner.class)
@ContextConfiguration(locations = { "classpath:spring/context-*.xml" })
@Transactional("ehrTxManager")
public class ExamBoardSvcTest {

    @Resource
    ExamBoardSvc examBoardSvc;

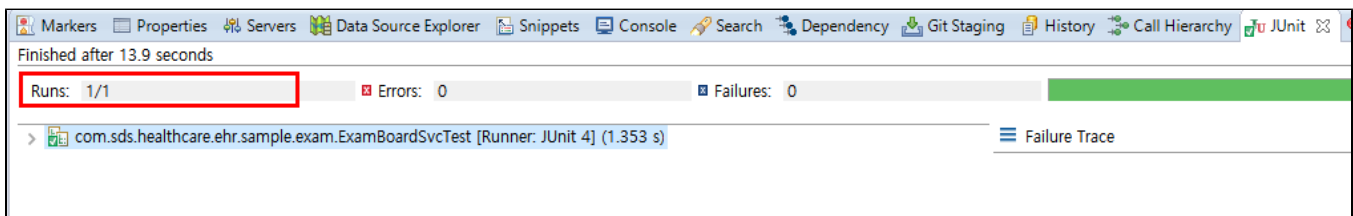
    @BeforeClass
    public static void setProperty() {
        System.setProperty("spring.profiles.active", "local_xxx");
    }

    @Test
    public void testGetBoardList() {
        ExamBoardSVO svo = new ExamBoardSVO();
        svo = examBoardSvc.getBoardList(svo);

        Assert.assertTrue(svo.getExamBoardDVOList().size() >= 0);
    }
}

```

테스트 코드를 실행하기 위해 Ctrl+F11을 키를 입력 합니다..

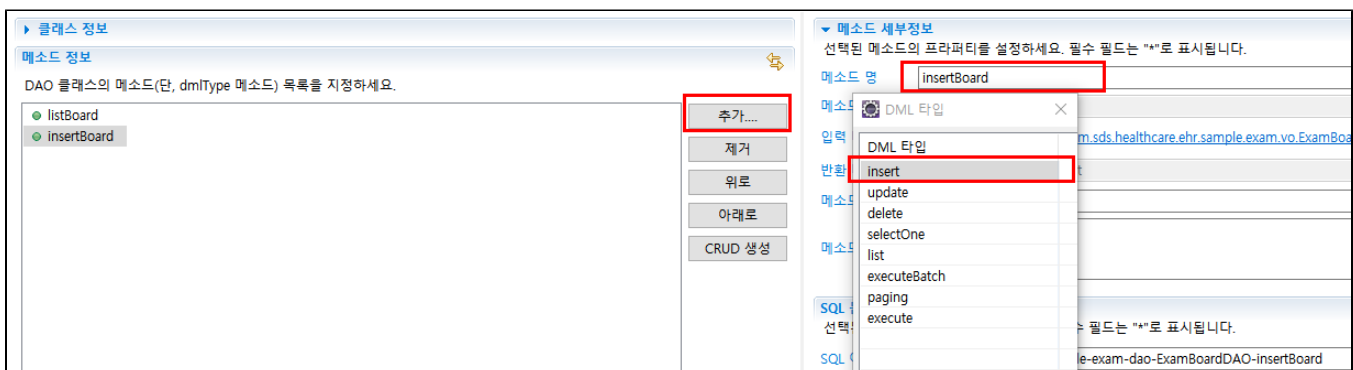


2. 등록

1) DAO 메소드 추가

ExamBoardDAO.java 소스를 오픈한 후 메소드 추가...버튼을 클릭합니다.

DML타입을 insert를 선택한 후 메소드 명을 insertBoard로 변경합니다.



2) SQL 쿼리 작성

SQL문의 편집 버튼을 클릭한 후 쿼리를 작성합니다.

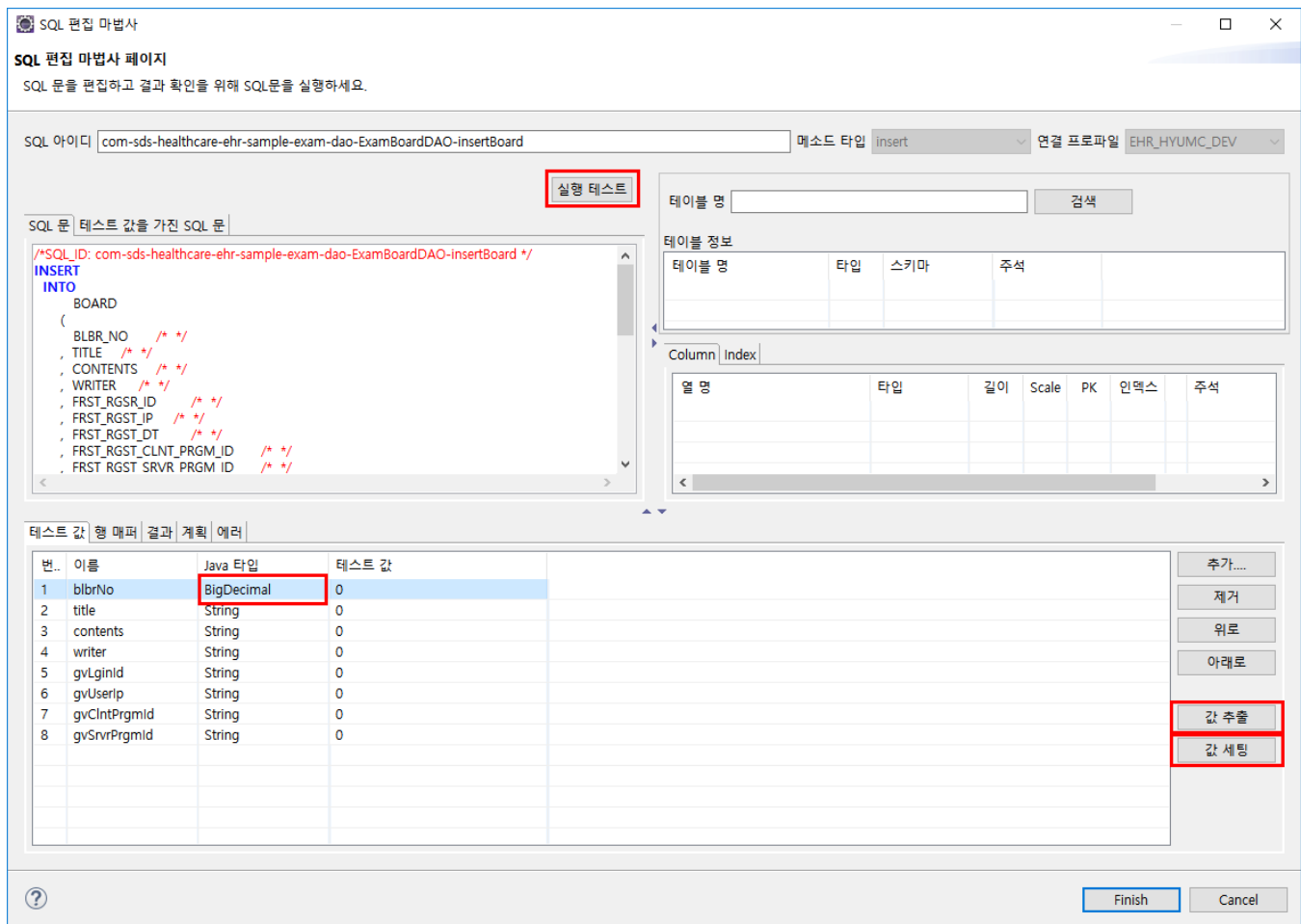
게시판등록쿼리

```
/*SQL_ID: com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-insertBoard */
INSERT
  INTO
    BOARD
  (
    BLBR_NO
    , TITLE
    , CONTENTS
    , WRITER
    , FRST_RGSR_ID
    , FRST_RGST_IP
    , FRST_RGST_DT
    , FRST_RGST_CLNT_PRGM_ID
    , FRST_RGST_SRVR_PRGM_ID
    , LAST_UPDR_ID
    , LAST_UPDT_IP
    , LAST_UPDT_DT
    , LAST_UPDT_CLNT_PRGM_ID
    , LAST_UPDT_SRVR_PRGM_ID
  )
  VALUES
  (
    #{blbrNo}
    , #{title}
    , #{contents}
    , #{writer}
    , #{gvLginId}
    , #{gvUserIp}
    , sysdate
    , #{gvClntPrgmId}
    , #{gvSrvrPrgmId}
    , #{gvLginId}
    , #{gvUserIp}
    , sysdate
    , #{gvClntPrgmId}
    , #{gvSrvrPrgmId}
  )
```

쿼리작성 완료되면 값추출, 값 세팅 버튼을 순서대로 클릭한 후 blbrNo의 Java 타입을 BigDecimal로 변경합니다. 실행 테스트 버튼을 클릭 후 Finish 버튼을 클릭합니다.

마지막으로 해당 DAO파일을 저장합니다.

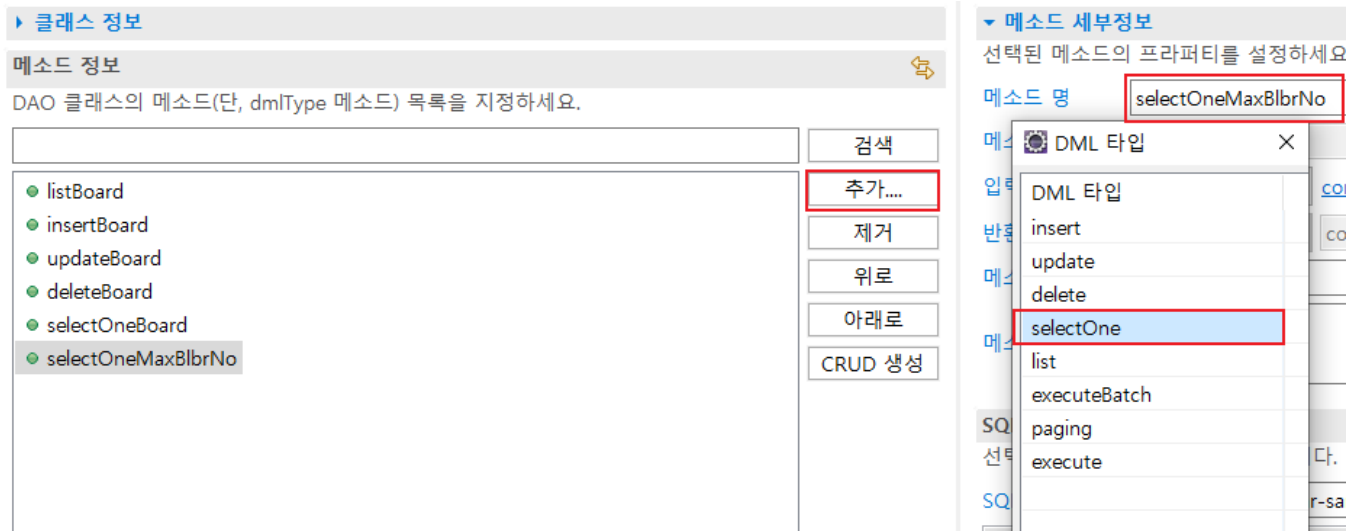
파라미터의 경우에는 테스트 값의 해당 Java 타입으로 DVO가 생성되므로 확인이 필요합니다. 값 추출시 최초에는 String으로 설정됩니다.



3) 게시판 번호 추출을 위한 DAO 메소드 추가

ExamBoardDAO.java 소스를 오픈한 후 메소드 추가...버튼을 클릭합니다.

DML타입을 selectOne를 선택한 후 메소드 명을 selectOneBlbrNo로 변경합니다.



4) SQL 쿼리 작성

SQL문의 편집 버튼을 클릭한 후 쿼리를 작성합니다.

게시판번호 추출

```
/*SQL_ID: com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-selectOneMaxBlbrNo */  
SELECT CAST(NVL(MAX(BLBR_NO),0) + 1 AS NUMBER(10)) AS "BLBR_NO"  
FROM BOARD
```

5) SVO 클래스 수정

SVO필드에서 등록시 사용되는 파라미터가 있는지 확인한 후 없는 경우 추가한다. SVO인 경우 gv로 시작하는 필드는 자동으로 값이 세팅되는 필드이므로 추가할 필요가 없습니다.

여기서는 writer필드가 존재하지 않으므로 writer필드를 추가합니다.

클래스 정보

클래스 명
ExamBoardSVO

클래스 [한글명]

패키지 명
com.sds.healthcare.ehr.sample.exam.vo

입출력 타입
VO

클래스 설명

[illegible]

6) BVO 클래스 수정

SVO와 동일한 방법으로 BVO를 VO에디터로 오픈 한 후 writer필드를 추가합니다.

7) Biz 클래스 메소드 추가

ExamBoardBiz.java 소스에 등록 처리를 위한 insBoard 메소드를 추가합니다.

```

/**
 *
 * @param bvo ExamBoardBVO
 * @return ExamBoardBVO
 * @author
 */
public ExamBoardBVO insBoard(ExamBoardBVO bvo) {
    ExamBoardDVO dvo = new ExamBoardDVO();
    VoUtil.copy(bvo, dvo);

    ExamBoardDVO blbrNoDVO = ExamBoardDAO.getInstance().selectOneMaxBlbrNo(dvo);
    dvo.setBlbrNo(blbrNoDVO.getBlbrNo());

    int insCount = ExamBoardDAO.getInstance().insertBoard(dvo);
    if ( insCount != 1 ) {
        throw new EHRBizException("EC14501");           // .
    }
    return bvo;
}

```

8) Svc 인터페이스 메소드 추가

ExamBoardSvc.java 소스에 insBoard라는 메소드를 추가한다. 서비스ID는 ExmInsBoardSID로 설정한다.

```
/**
 *
 * @param svo ExamBoardSVO
 * @return ExamBoardSVO
 * @author
 */
@ServiceIdMapping("ExmInsBoardSID")
ExamBoardSVO insBoard(ExamBoardSVO svo);
```

9) SvcImpl 클래스 메소드 추가

위에 Svc 인터페이스에 정의된 insBoard 메소드를 ExamBoardSvcImpl.java 소스에 구현한다.

```
@Override
public ExamBoardSVO insBoard(ExamBoardSVO svo) {
    ExamBoardBVO bvo = new ExamBoardBVO();
    VoUtil.copy(svo, bvo);

    ExamBoardBiz biz = new ExamBoardBiz();
    biz.insBoard(bvo);

    return svo;
}
```

10) JUnit 메소드 추가

조희시 생성했던 ExamBoardSvcTest.java소스에 아래와 같이 testInsBoard 메소드를 추가한 후 테스트 코드를 입력한다.

gv로 시작하는 값은 실제 런타임시 설정되는 값이나 테스트 코드에서는 자동으로 설정되지 않으므로 입력해 주어야 합니다. (해당 컬럼은 필수 값이므로 입력하지 않는 경우 오류가 발생합니다.)

Run을 실행(Ctrl+F11) 하여 정상적으로 실행되는지 테스트 합니다.

```

@Test
    public void testInsBoard() {
        ExamBoardSVO svo = new ExamBoardSVO();
        svo.setBlbrNo(new BigDecimal(1));
        svo.setTitle("title");
        svo.setContents("contents");
        svo.setWriter("SDS");

        svo.setGvUserIp("localhost");
        svo.setGvUserId("700000");
        svo.setGvLginId("700000");
        svo.setGvClntPrgmId("SDS");
        svo.setGvSrvrPrgmId("ExmInsBoardSID");

        examBoardSvc.insBoard(svo);
    }

```

다음 수정, 삭제 실패를 위해 데이터를 1건 등록합니다. 테스트 코드를 이용하여 해당 데이터를 등록합니다.

@Rollback(false)코드를 등록 테스트 메소드에 입력하여 rollback하지 않도록 설정합니다. Run 실행 후 해당 어노테이션은 삭제합니다.

```

@Test
@Rollback(false)
    public void testInsBoard() {
        ...
    }

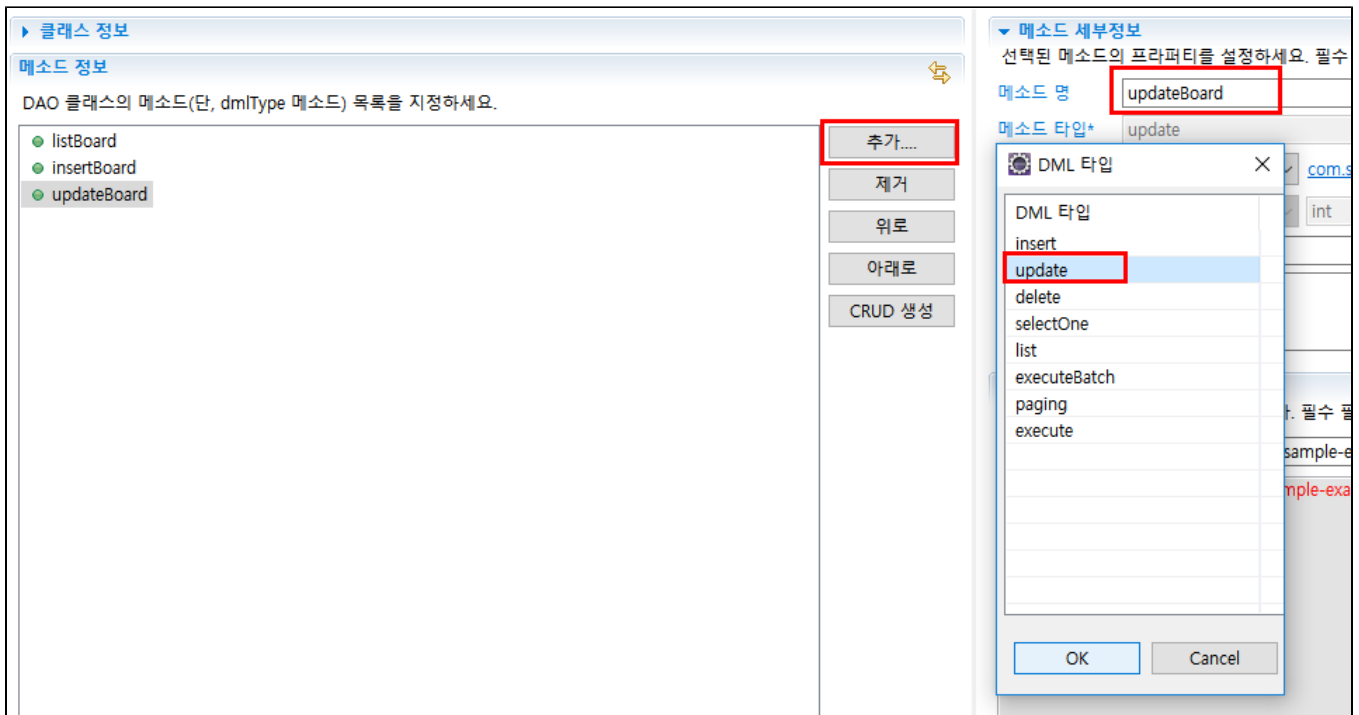
```

3. 수정

1) DAO 메소드 추가

ExamBoardDAO.java 소스를 오픈한 후 메소드 추가...버튼을 클릭합니다.

DML타입을 update를 선택한 후 메소드 명을 updateBoard로 변경합니다.



2) SQL 쿼리 작성

SQL문의 편집 버튼을 클릭한 후 쿼리를 작성합니다.

게시판수정쿼리

```

/*SQL_ID: com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-updateBoard */
UPDATE
    BOARD
SET
    TITLE                = #{title}                /* */
    , CONTENTS            = #{contents}              /* */
    , WRITER              = #{writer}                /* */
    , LAST_UPDR_ID        = #{gvLginId}              /* ID */
    , LAST_UPDT_IP        = #{gvUserIp}              /* IP */
    , LAST_UPDT_DT        = sysdate                  /* */
    , LAST_UPDT_CLNT_PRGM_ID = #{gvClntPrgmId}        /* ID */
    , LAST_UPDT_SRVR_PRGM_ID = #{gvSrvrPrgmId}        /* ID */
WHERE
    BLBR_NO = #{blbrNo}

```

쿼리작성 완료되면 값추출, 값 세팅 버튼을 순서대로 클릭한 후 blbrNo의 Java 타입을 BigDecimal로 변경합니다.

실행 테스트 버튼을 클릭 후 Finish 버튼을 클릭합니다. 마지막으로 해당 DAO파일을 저장합니다.

SQL 편집 마법사

SQL 편집 마법사 페이지

SQL 문을 편집하고 결과 확인을 위해 SQL문을 실행하세요.

SQL 아이디

com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-updateBoard

메소드 타입

update

연결 프로파일

EHR_HYUMC_DEV

실행 테스트

테이블 명

검색

테이블 정보

테이블 명	타입	스키마	주석

Column Index

열 명	타입	길이	Scale	PK	인덱스	주석

SQL 문

테스트 값을 가진 SQL 문

```

/*SQL_ID: com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-updateBoard */
UPDATE
BOARD
SET
TITLE           = #{title} /* 타이틀 */
, CONTENTS      = #{contents} /* 내용 */
, WRITER        = #{writer} /* 작성자 */
, LAST_UPDR_ID  = #{gvLginId}
, LAST_UPDT_IP  = #{gvUserIp}
, LAST_UPDT_DT   = sysdate
, LAST_UPDT_CLNT_PRGM_ID = #{gvClntPrgmId}
, LAST_UPDT_SRVR_PRGM_ID = #{gvSrvrPrgmId}
WHERE
BLBR NO = #{blbrNo}

```

테스트 값

행 매핑

결과

계획

여러

번..	이름	Java 타입	테스트 값
1	title	String	0
2	contents	String	0
3	writer	String	0
4	gvLginId	String	0
5	gvUserIp	String	0
6	gvClntPrgmId	String	0
7	gvSrvrPrgmId	String	0
8	blbrNo	BigDecimal	0

추가...

제거

위로

아래로

값 추출

값 세팅

Finish

Cancel

3) Biz 클래스 메소드 추가

ExamBoardBiz.java 소스에 수정 처리를 위한 updBoard 메소드를 추가합니다.

```

/**
 *
 * @param bvo ExamBoardBVO
 * @return ExamBoardBVO
 * @author
 */
public ExamBoardBVO updBoard(ExamBoardBVO bvo) {
    ExamBoardDVO dvo = new ExamBoardDVO();
    VoUtil.copy(bvo, dvo);

    int updCount = ExamBoardDAO.getInstance().updateBoard(dvo);
    if ( updCount != 1 ) {
        throw new EHRBizException("EC14502"); // .
    }
    return bvo;
}

```

4) Svc 인터페이스 메소드 추가

ExamBoardSvc.java 소스에 updBoard라는 메소드를 추가한다. 서비스ID는 ExmUpdBoardSID로 설정한다.

```
/**
 *
 * @param svo ExamBoardSVO
 * @return ExamBoardSVO
 * @author
 */
@ServiceIdMapping("ExmUpdBoardSID")
ExamBoardSVO updBoard(ExamBoardSVO svo);
```

5) SvcImpl 클래스 메소드 추가

위에 Svc 인터페이스에 정의된 updBoard 메소드를 ExamBoardSvcImpl.java 소스에 구현한다.

```
@Override
public ExamBoardSVO updBoard(ExamBoardSVO svo) {
    ExamBoardBVO bvo = new ExamBoardBVO();
    VoUtil.copy(svo, bvo);

    ExamBoardBiz biz = new ExamBoardBiz();
    biz.updBoard(bvo);

    return svo;
}
```

6) JUnit 메소드 추가

조희시 생성했던 ExamBoardSvcTest.java 소스에 아래와 같이 testUpdBoard 메소드를 추가한 후 테스트 코드를 입력한다.

Run을 실행(Ctrl+F11) 하여 정상적으로 실행되는지 테스트 합니다.

```
@Test
public void testUpdBoard() {
    ExamBoardSVO svo = new ExamBoardSVO();
    svo.setBlbrNo(new BigDecimal(1));
    svo.setTitle("title-update");
    svo.setContents("contents-update");
    svo.setWriter("SDS");

    svo.setGvUserIp("localhost");
    svo.setGvUserId("700000");
    svo.setGvLginId("700000");
    svo.setGvClntPrgmId("SDS");
    svo.setGvSrvrPrgmId("ExmInsBoardSID");

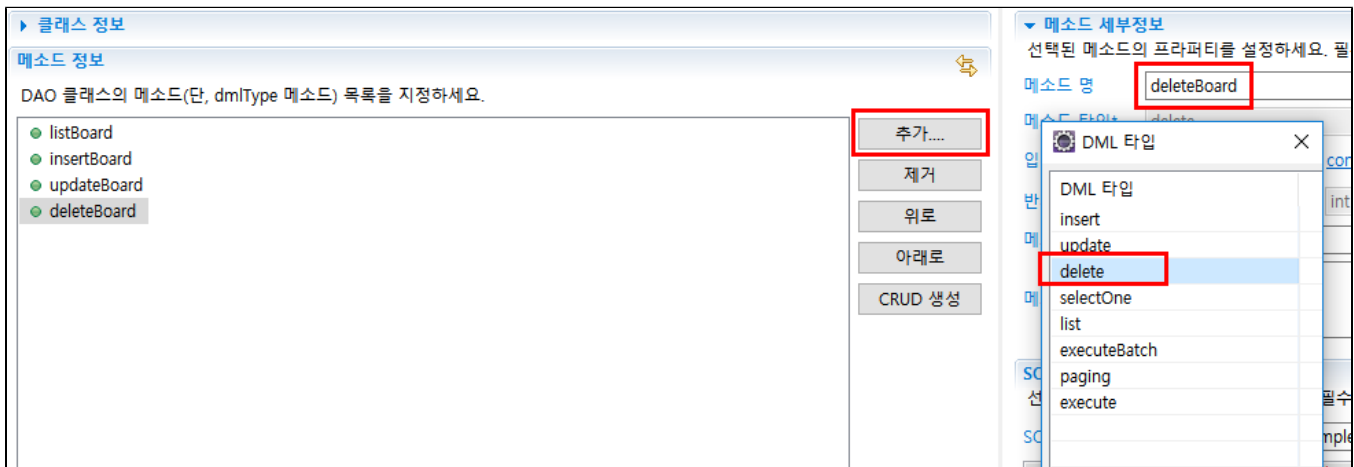
    examBoardSvc.updBoard(svo);
}
```

4. 삭제

1) DAO 메소드 추가

ExamBoardDAO.java 소스를 오픈한 후 메소드 추가...버튼을 클릭합니다.

DML타입을 delete를 선택한 후 메소드 명을 deleteBoard로 변경합니다.



2) SQL 쿼리 작성

SQL문의 편집 버튼을 클릭한 후 쿼리를 작성합니다.

게시판삭제쿼리

```
/*SQL_ID: com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-deleteBoard */
DELETE
FROM
    BOARD
WHERE  BLBR_NO = #{blbrNo}
```

쿼리작성 완료되면 값추출, 값 세팅 버튼을 순서대로 클릭한 후 blbrNo의 Java 타입을 BigDecimal로 변경합니다.

실행 테스트 버튼을 클릭 후 Finish 버튼을 클릭합니다. 마지막으로 해당 DAO파일을 저장합니다.

SQL 편집 마법사

SQL 편집 마법사 페이지

SQL 문을 편집하고 결과 확인을 위해 SQL문을 실행하세요.

SQL 아이디

com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-deleteBoard

메소드 타입

delete

연결 프로파일

EHR_HYUMC_DEV

실행 테스트

테스트 문

테스트 값을 가진 SQL 문

```

/*SQL_ID: com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-deleteBoard */
DELETE
FROM
  BOARD
WHERE BLBR_NO = 0

```

테이블 정보

테이블 명

타입

스키마

주석

Column

Index

열 명

타입

길이

Scale

PK

인덱스

주석

테스트 값

행 매핑

결과

계획

예러

번..	이름	Java 타입	테스트 값
1	blbrNo	BigDecimal	0

추가...

제거

위로

아래로

값 추출

값 세팅

Finish

Cancel

3) Biz 클래스 메소드 추가

ExamBoardBiz.java 소스에 수정 처리를 위한 delBoard 메소드를 추가합니다.

```

/**
 *
 * @param bvo ExamBoardBVO
 * @return ExamBoardBVO
 * @author
 */
public ExamBoardBVO delBoard(ExamBoardBVO bvo) {
    ExamBoardDVO dvo = new ExamBoardDVO();
    VoUtil.copy(bvo, dvo);

    int delCount = ExamBoardDAO.getInstance().deleteBoard(dvo);
    if ( delCount != 1 ) {
        throw new EHRBizException("EC99166"); // .
    }
    return bvo;
}

```


4) Svc 인터페이스 메소드 추가

ExamBoardSvc.java 소스에 delBoard라는 메소드를 추가한다. 서비스ID는 ExmDelBoardSID로 설정한다.

```
/**
 *
 * @param svo ExamBoardSVO
 * @return ExamBoardSVO
 * @author
 */
@ServiceIdMapping("ExmDelBoardSID")
ExamBoardSVO delBoard(ExamBoardSVO svo);
```

5) SvcImpl 클래스 메소드 추가

위에 Svc 인터페이스에 정의된 delBoard 메소드를 ExamBoardSvcImpl.java 소스에 구현한다.

```
@Override
public ExamBoardSVO delBoard(ExamBoardSVO svo) {
    ExamBoardBVO bvo = new ExamBoardBVO();
    VoUtil.copy(svo, bvo);

    ExamBoardBiz biz = new ExamBoardBiz();
    biz.delBoard(bvo);

    return svo;
}
```

6) JUnit 메소드 추가

조회시 생성했던 ExamBoardSvcTest.java소스에 아래와 같이 testDelBoard 메소드를 추가한 후 테스트 코드를 입력한다.

Run을 실행(Ctrl+F11) 하여 정상적으로 실행되는지 테스트 합니다.

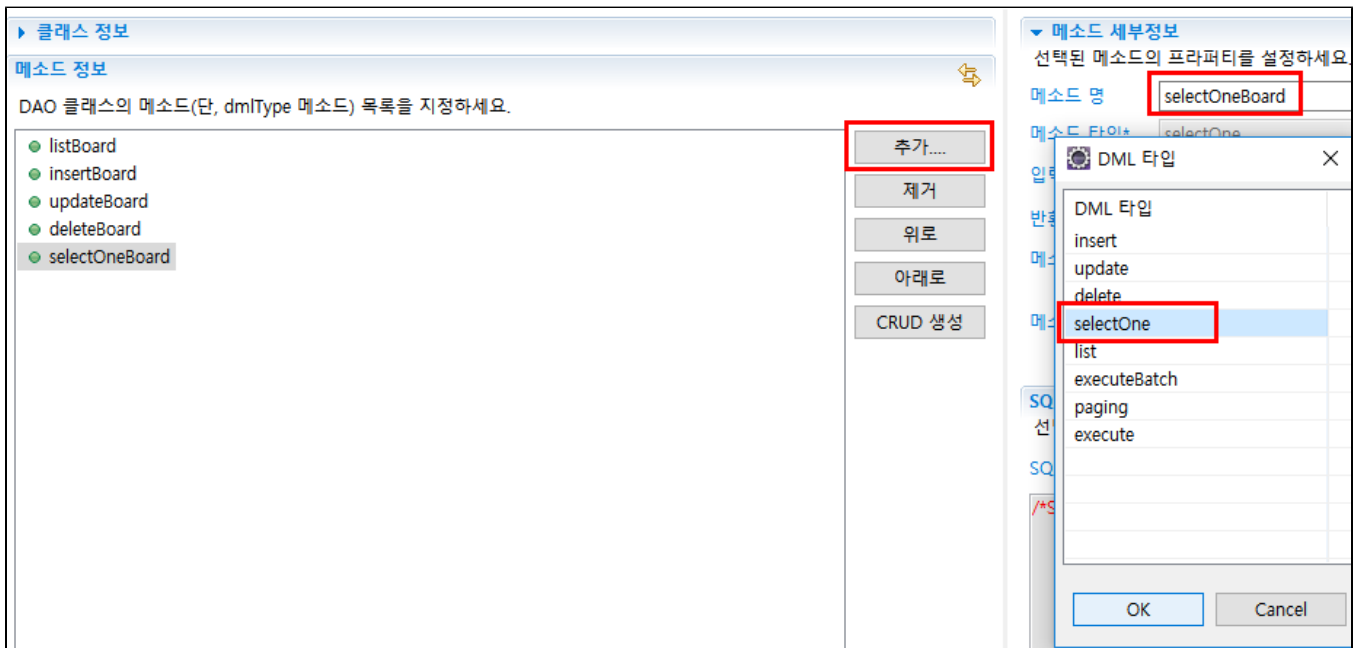
```
@Test
public void testDelBoard() {
    ExamBoardSVO svo = new ExamBoardSVO();
    svo.setBlbrNo(new BigDecimal(1));
    examBoardSvc.delBoard(svo);
}
```

5.조회(단건) - CommonService 이용

1) DAO 메소드 추가

ExamBoardDAO.java 소스를 오픈한 후 메소드 추가...버튼을 클릭합니다.

DML타입을 selectOne를 선택한 후 메소드 명을 selectOneBoard로 변경합니다.



2) SQL 쿼리 작성

SQL문의 편집 버튼을 클릭한 후 쿼리를 작성합니다.

게시판 단건조회쿼리

```
/*SQL_ID: com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-selectOneBoard */
SELECT /*+ TstExamBoraddDAO-selectOneBoard_ */
      BLBR_NO          /* */
    , TITLE            /* */
    , CONTENTS         /* */
    , WRITER           /* */
FROM   BOARD
WHERE  BLBR_NO = #{blbrNo}
```

쿼리작성 완료되면 값추출, 값 세팅 버튼을 순서대로 클릭한 후 blbrNo의 Java 타입을 BigDecimal로 변경합니다.

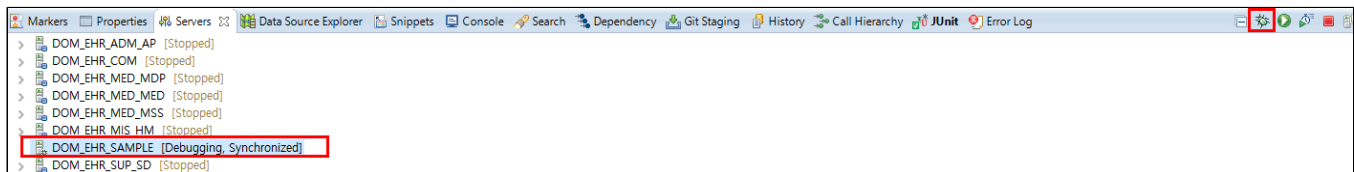
실행 테스트 버튼을 클릭 후 Finish 버튼을 클릭합니다. 마지막으로 해당 DAO파일을 저장합니다.

Sql Id : com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-selectOneBoard

The image shows a 'Common Service Manager' window. At the top, there's a 'Common Service' section with input fields for 'Service Id' (containing 'ExmGetBoardCSID') and 'Sql Id' (containing 'com-sds-healthcare-ehr-sample-exam-dao-ExamBoardDAO-selectOneBoard'). Below these are radio buttons for 'Sql Type' with 'SELECT' selected. To the right are 'Insert', 'Update', and 'Delete' buttons. Below this is a search section with 'Service Id', 'Sql Id', and 'Sql Type' (set to 'BOTH') filters, and a 'Search' button. The main area is a table with columns: 'Service Id', 'Sql Id', 'Sql Type', 'User Id', and 'Last Modified'. The table is currently empty.

4) 테스트 화면에서 해당 ExmGetBoardCSID 호출 테스트

이클립스에서 DOM_EHR_SAMPLE 프로젝트를 이용해서 서버를 구동합니다.



웹 브라우저를 오픈한 후 http://localhost:8080/DOM_EHR_SAMPLE/anyframe.jsp 로 이동합니다.

Service Type을 Simple로 선택하고 Input Data를 아래와 같이 입력합니다.

InputData : {"ExmGetBoardCSID":{"blbrNo":"1","gvUserIp":"localhost","gvUserId":"700000","gvCntPrgmId":"NONE","gvSvrPrgmId":"

ExmGetBoardCSID","userId":"700000","gvLginId":"700000"}}

전송버튼을 클릭하면 Output Data에 해당 blbrNo가 1인 데이터가 조회 됩니다.

Service Type Simple ▾

Input Data

```
{"ExmGetBoardCSID": {"blbrNo": "1", "gvUserIp": "localhost", "gvUserId": "700000", "gvCntPrgmId": "NONE", "gvSvrPrgmId": "ExmGetBoardCSID", "userId": "700000", "gvLginId": "700000"}}
```

전송

Output Data

```
{  "ExmGetBoardCSID": {    "pageSize": 0,    "blbrNo": 1,    "title": "1_Title",    "contents": "1_Contents",    "writer": "1_Tester"  },  "errorcode": "0",  "errmsg": []}
```

Error