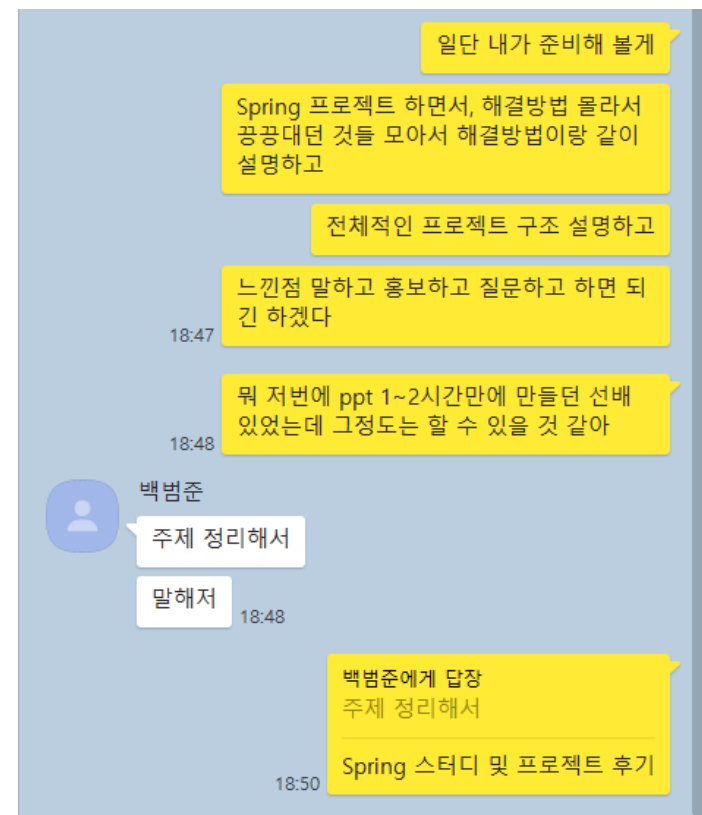
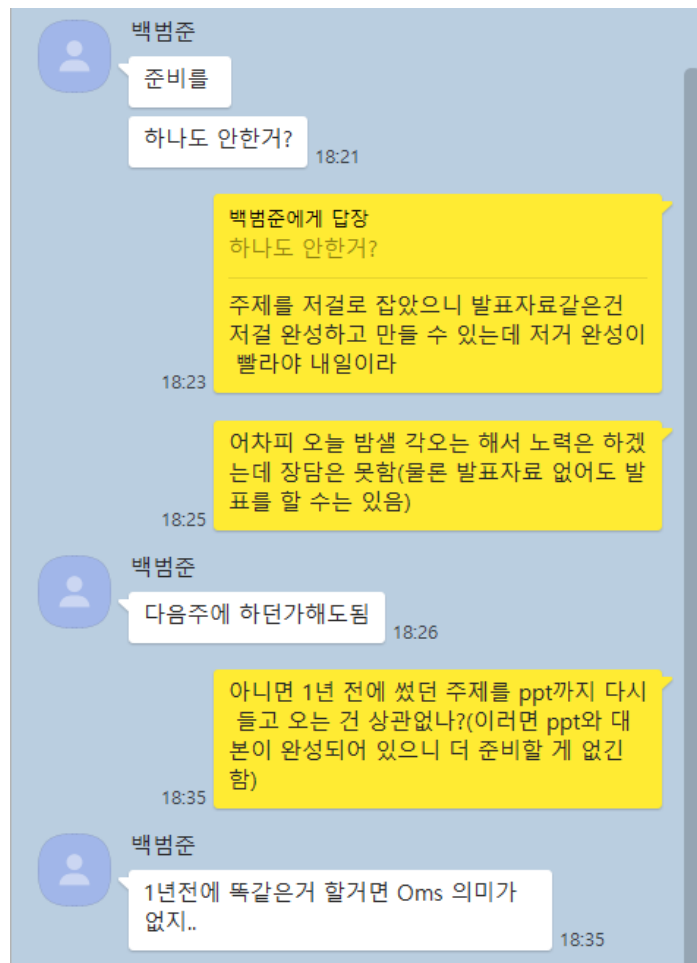
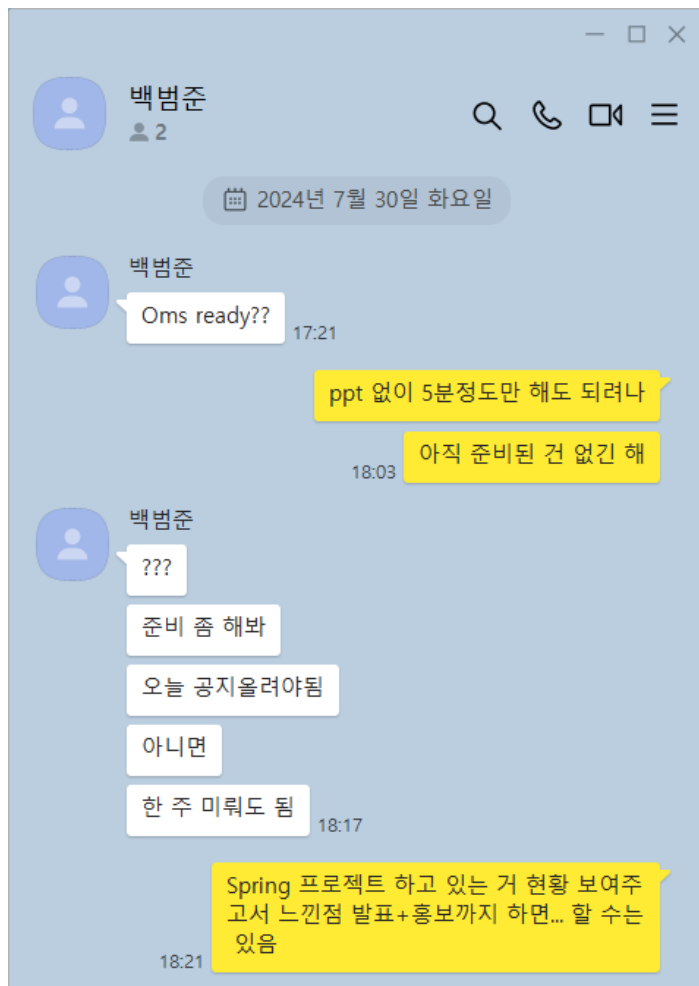


Spring 스터디 및 프로젝트 후기

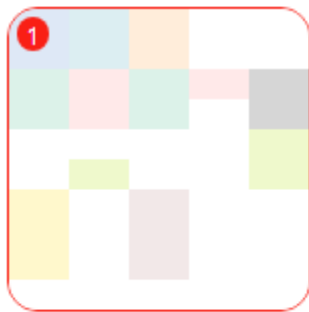
Zeropage 33기 정현승



결과 보기

[돌아가기](#)

38,880개의 경우의 수 중에서
1,514개의 시간표가 만들어졌습니다.
마음에 드는 시간표가 있다면 저장해주세요.



그룹 3 ⊕ 🗑

과목번호-분반	과목명	담당교수	강의실/강의시간	삭제
40458-02	오토마타와형식언어 (영어A강의)		수6 / 월5,6 / 310관 723호 / 729호	×
40458-01	오토마타와형식언어 (영어A강의)		화1,2 / 목2 / 310관 723호	×

그룹 4 ⊕ 🗑

과목번호-분반	과목명	담당교수	강의실/강의시간	삭제
34692-04	데이터베이스설계	이성훈	금2,3,4 / 310관 723호	×
34692-02	데이터베이스설계 (영어A강의)	고윤용	목6 / 금5,6 / 310관 723호	×
34692-01	데이터베이스설계	강현철	월5 / 수5,6 / 00 000000	×
34692-03	데이터베이스설계 (영어A강의)	이재용	수7,8,9 / 310관 723호	×

그룹 5 ⊕ 🗑

과목번호-분반	과목명	담당교수	강의실/강의시간	삭제
49156-02	객체지향프로그래밍 (영어A강의)	손봉수	월3,4 / 수3 / 310관 728호	×
49156-03	객체지향프로그래밍 (영어A강의)	손봉수	월6 / 수5,6 / 310관 728호 / 729호	×
49156-01	객체지향프로그래밍 (영어A강의)	손봉수	월2 / 수1,2 / 310관 728호	×

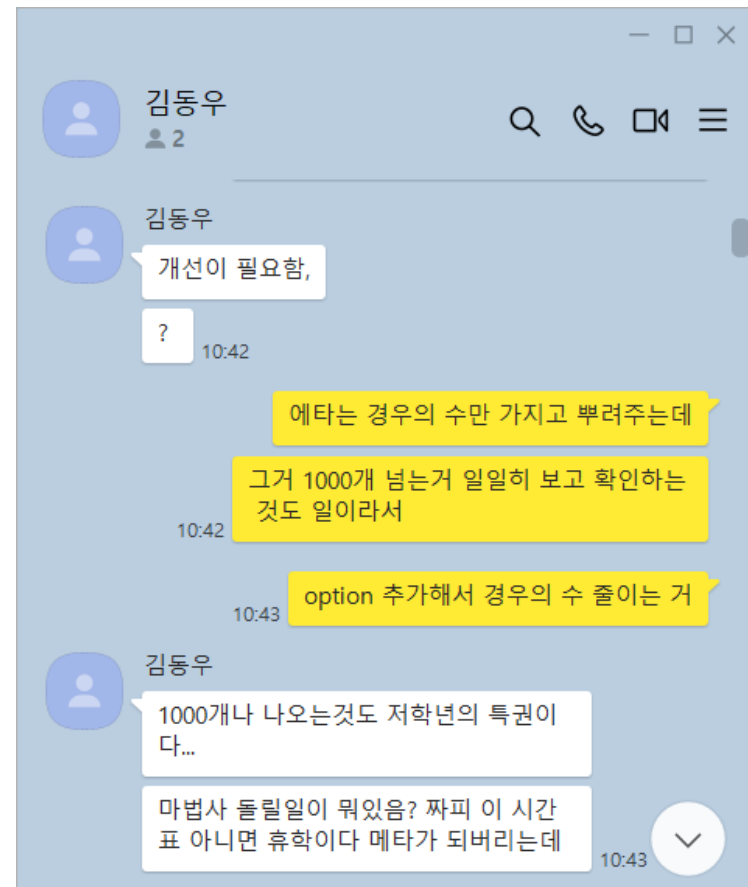
그룹 6 ⊕ 🗑

과목번호-분반	과목명	담당교수	강의실/강의시간	삭제
---------	-----	------	----------	----

3개 그룹 중 2개 과목만 수강 희망

핵교 5개 영역 중 2~3개 영역만 수강 희망

최소 하루는 공강 만들고 싶음



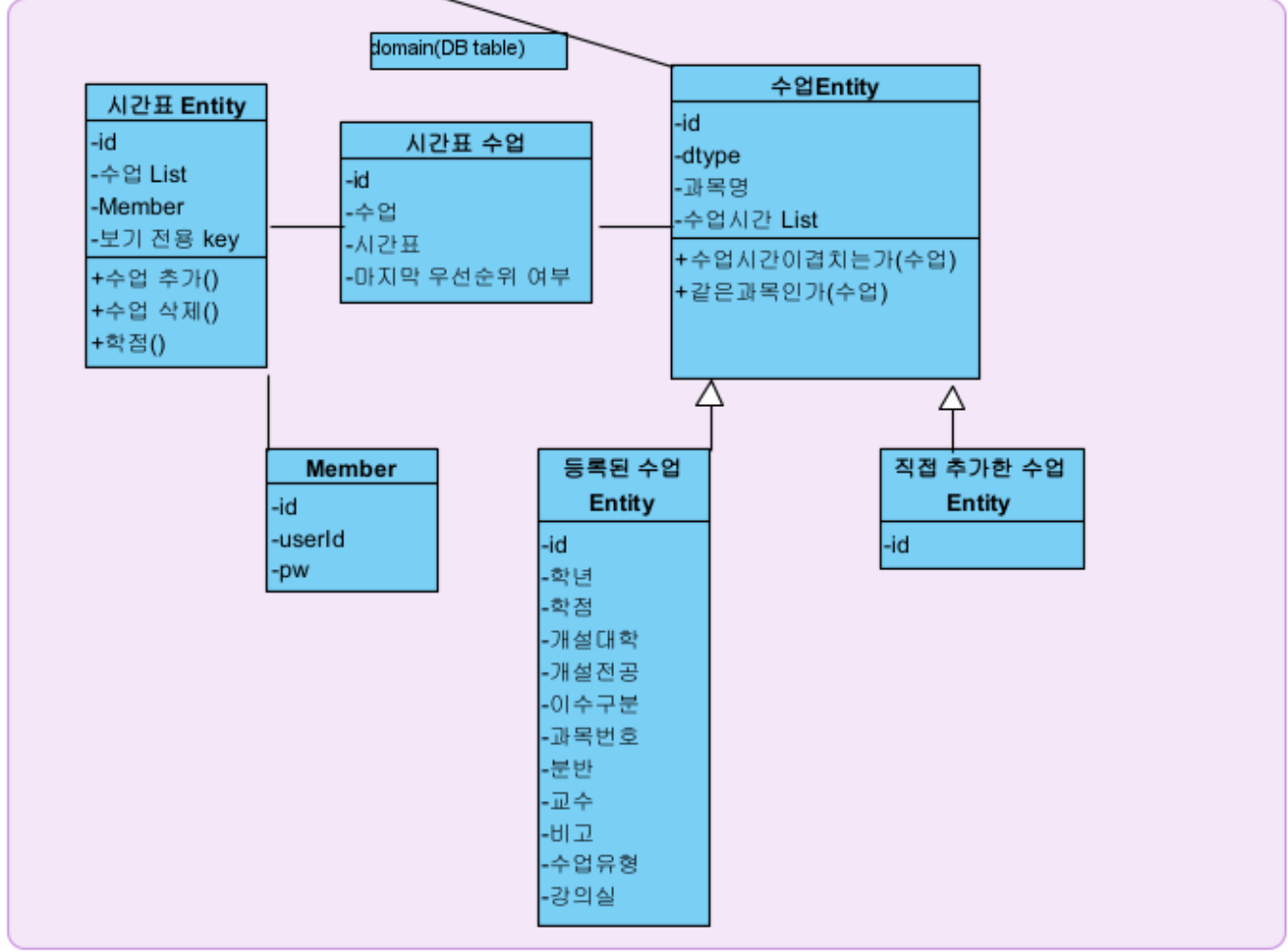
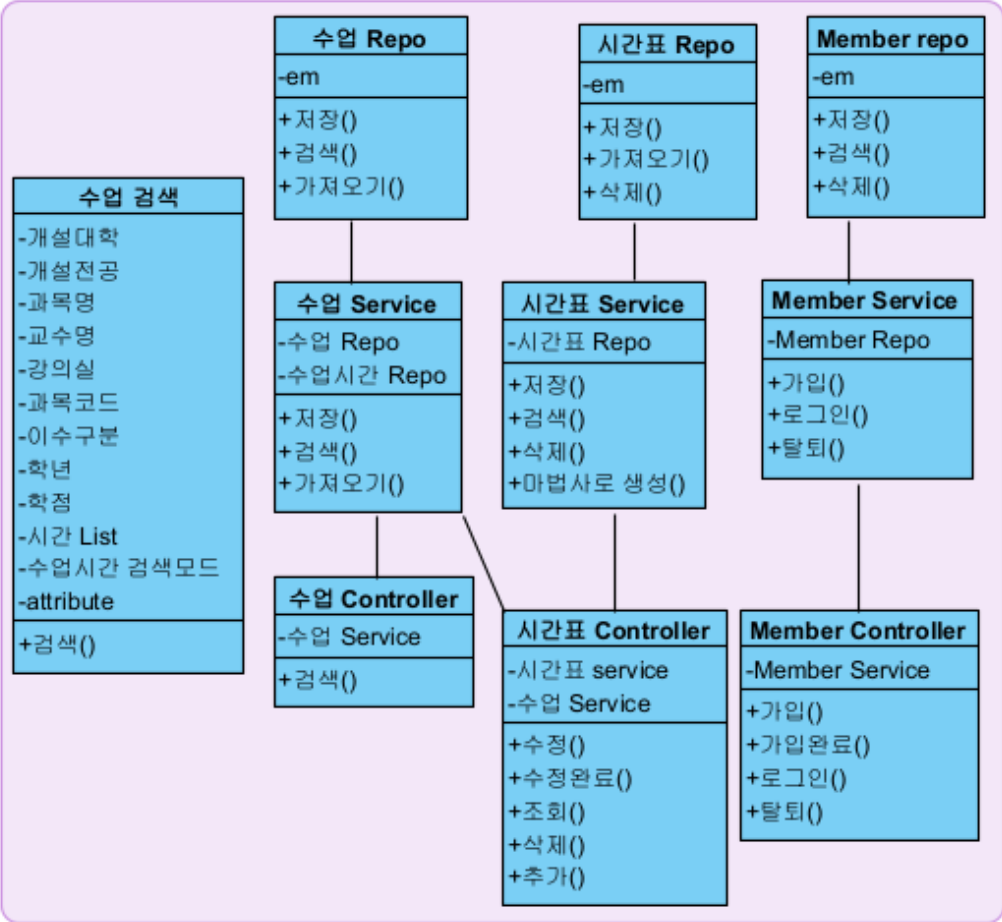
이것들은 DB
에 저장하지
않는 것

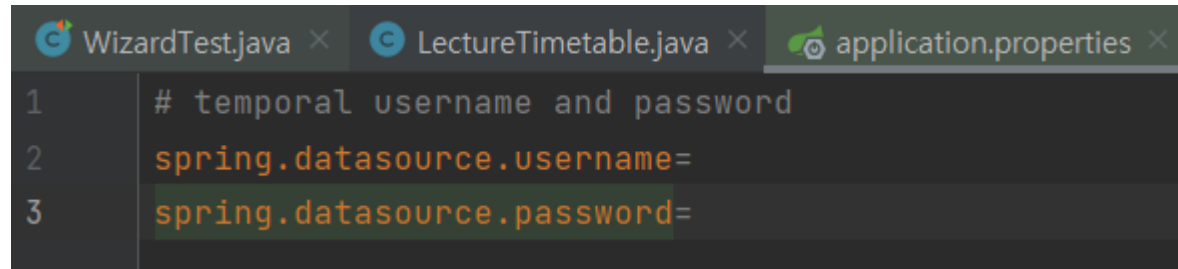
시간표 마법사
-수업 목록
-공강 요일 List
-공강 요일 수 최소
-강의 시작 희망 시각
-강의 종료 희망 시각
-토요일 공강 계산
-영상강의 조건제외
-비대면강의 공강제외
-비대면강의 조건제외
-마지막 우선순위 전체결과표시
+안되는 시간 추가()
+결과()

그룹 목록(GroupOfGroup)
-수업 목록
-이름
-목록 조합
-마지막 우선순위 여부

학교 5가지 영역 중 몇 가
지 영역 선택하는 사례 등
을 대비해 그룹을 한 번
더 그룹으로 감쌌

수업 목록(LectureGroup)
-이름
-강의 목록
-선택하는 강의 수





```
1 # temporal username and password
2 spring.datasource.username=
3 spring.datasource.password=
```

시간 조회, 검색 어떻게 하는가

Before

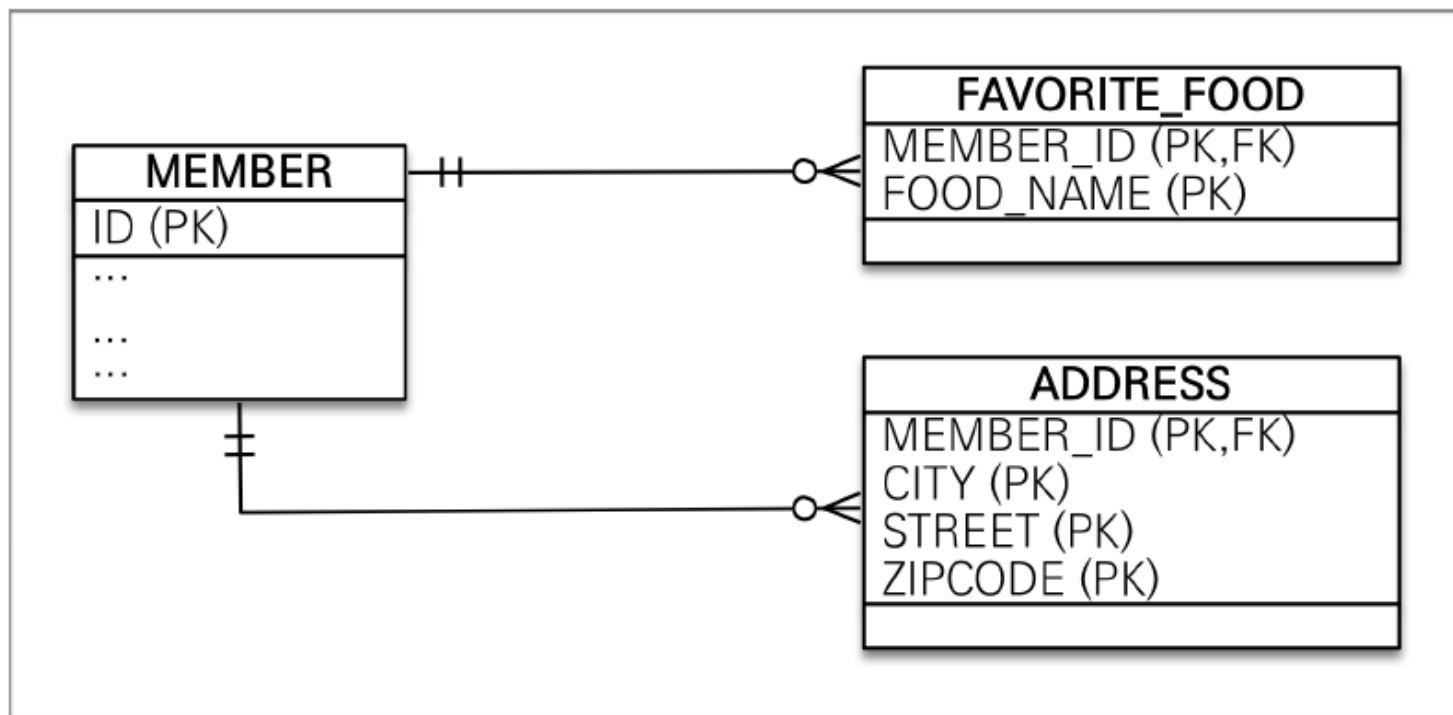
```
/**
 * Save by splitting one week by 30 minutes.
 * 0 is Monday 00:00~00:30,
 * 1 is Monday 00:30~01:00,
 * 47 is Monday 23:30~24:00,
 * 48 is Tuesday 00:00~00:30,
 * 95 is Tuesday 23:30~24:00,
 * 143 is Wednesday 23:30~24:00,
 * 191 is Thursday 23:30~24:00,
 * 239 is Friday 23:30~24:00,
 * 287 is Saturday 23:30~24:00.
 * Didn't expected Sunday because there are no enrolled lecture on Sunday.
 */
@ElementCollection
private List<Integer> classHours = new ArrayList<>();
```

DB에 별도의 column 형성

➔ 검색 시 페이징 불가(물론 문제를 해결한 지금도 이건 생략하긴 함)

➔ 매우 비효율적

값 타입 컬렉션



시간 조회, 검색 어떻게 하는가

Before

```
/**
 * Save by splitting one week by 30 minutes.
 * 0 is Monday 00:00~00:30,
 * 1 is Monday 00:30~01:00,
 * 47 is Monday 23:30~24:00,
 * 48 is Tuesday 00:00~00:30,
 * 95 is Tuesday 23:30~24:00,
 * 143 is Wednesday 23:30~24:00,
 * 191 is Thursday 23:30~24:00,
 * 239 is Friday 23:30~24:00,
 * 287 is Saturday 23:30~24:00.
 * Didn't expected Sunday because there are no enrolled lecture on Sunday.
 */
@ElementCollection
private List<Integer> classHours = new ArrayList<>();
```

After

```
* 한 주를 30분으로 나누어 저장함. ', '으로 구분.
* 0은 월요일 00:00~00:30,
* 1은 월요일 00:30~01:00,
* 47은 월요일 23:30~24:00,
* 48은 화요일 00:00~00:30,
* 95는 화요일 23:30~24:00,
* 143은 수요일 23:30~24:00,
* 191은 목요일 23:30~24:00,
* 239는 금요일 23:30~24:00,
* 287은 토요일 23:30~24:00.
* 일요일에는 강의가 없으므로 생각하지 않았음. (어느 정도 대비를 해 놓긴 함)
* 별도의 Entity를 쓰면 join 문제가 발생해 String으로 임시 대체
*/
private String classHours;
```

DB에 별도의 column 형성

➔ 검색 시 페이징 불가(물론 문제를 해결한 지금도 이건 생략하긴 함)

➔ 매우 비효율적

```

int classHoursSize = options.getClassHours().size();
int classHoursQuerySize = 0; // 아래 if문 이후로는 0이면 classHours 조건이 없는 것임
if (classHoursSize != 0 && classHoursSize != 24 * 2 * 7
    // 모든 시간을 전부 선택하면 의미가 없으므로 그것도 넘김
    && options.getTimeSearchMode() != null) {

    jpql += " and not l.classHours=:emptyClassHours";
    switch (options.getTimeSearchMode()) {
        case COVER -> {
            // 이 이외의 시간을 전부 빼는 방법으로 구현
            classHoursQuerySize = 24 * 2 * 7 - classHoursSize;
            for (int i = 0; i < classHoursQuerySize; i++)
                jpql += " and l.classHours not like :classHours" + i;
        }
        case INCLUDE -> {
            classHoursQuerySize = classHoursSize;
            jpql += " and (l.classHours like :classHours0";
            for (int i = 1; i < classHoursQuerySize; i++)
                jpql += " or l.classHours like :classHours" + i;
            jpql += ")";
        }
    }
}
}

```

```

if (classHoursQuerySize != 0) {
    query = query.setParameter( name: "emptyClassHours", value: "");
    switch (options.getTimeSearchMode()) {
        case COVER -> {
            Set<Integer> targetClassHours = new HashSet<>(options.getClassHours());
            Integer targetClassHoursConut = 0;
            for (int i = 0; i < classHoursQuerySize; i++, targetClassHoursConut++) {
                while (targetClassHours.contains(targetClassHoursConut))
                    targetClassHoursConut++;
                query = query.setParameter( name: "classHours" + i,
                    value: "%," + targetClassHoursConut.toString() + ",%");
            }
        }
        case INCLUDE -> {
            List<Integer> classHours = options.getClassHours();
            for (int i = 0; i < classHoursQuerySize; i++)
                query = query.setParameter( name: "classHours" + i,
                    value: "%," + classHours.get(i).toString() + ",%");
            }
    }
}
}

```

<div> <div>캠퍼스: 서울</div> <div>전공/영역: 전체</div> <div>검색어: 없음</div> <div>정렬: 기본</div> <div>시간: 월11-13시 ×</div> <div>학년: 전체</div> <div>구분: 전체</div> <div>학점: 전체</div> <div>닫기</div> </div>											
학년	이수구분	과목번호-분반	과목명	담당교수	학점	시간	강의실/강의시간	강의평	담은 인원	개설학과	유의사항
전체	교양	54341-01	구기스포스	정두식	2	2	B112호	★★★★★	9		인력-융합 플쓰
전체	교양	39967-01	ENGLISH WRITING(1) (영어A강의)	사라황	2	2	월3,4 / 303관(법학관) 902-1호	★★★★★	5		혼합형(원격70%이상)+대면수업, 선택-소통
1	전공	51200-01	육상지도법2	이정민	1	2	월3,4 / 305관(교수연구동 및 체육관) 220호	★★★★★	21	체육교육과	
1	전공기초	17465-02	일반생물학실험(2)	이성철	1	2	월3,4 / 104관(수림과학관) 601호	★★★★★	12	생명과학과	
1	전공기초	17441-01	일반물리학(2)	윤육자	2	2	월3,4 / 310관 414호	★★★★★	24	사회기반시스템공학부	건설/MSK
1	자유선택	51097-06	CAU세미나(2)	정도진	0.5	.5	월4 / 310관 926호	★★★★★	0	경영학전공	
1	자유선택	51097-14	CAU세미나(2)	권혁인	0.5	.5	월4 / 310관 932호	★★★★★	1	경영학전공	
1	자유선택	51097-26	CAU세미나(2)	유시용	0.5	.5	월4 / 310관 927호	★★★★★	0	경영학전공	
1	전공	41970-01	GF 참세미나(1) (영어A강의)	정준영	1	1	월4 / 310관 928호	★★★★★	33	글로벌금융전공	글로벌금융생 외 수강불가
2	전공	57333-01	연구학위학위	김재현	1	2	월3,4 / 102관(약학대학 및 R&D센터) 312	★★★★★	12	약학부	연구생만 수강가능

포함 안됨

2	전공	49156-02	객체지향프로그래밍 (영어A강의)	손봉수	3	3	월3,4 / 수3 / 310관 728호	★★★★★	84	소프트웨어학부	공학주제(설계2), 코드웨어(전전,융합)
---	----	----------	-------------------	-----	---	---	-----------------------	-------	----	---------	------------------------

```

int classHoursSize = options.getClassHours().size();
int classHoursQuerySize = 0; // 아래 if문 이후로는 0이면 classHours 조건이 없는 것임
if (classHoursSize != 0 && classHoursSize != 24 * 2 * 7
    // 모든 시간을 전부 선택하면 의미가 없으므로 그것도 넘김
    && options.getTimeSearchMode() != null) {

    jpql += " and not l.classHours=:emptyClassHours";
    switch (options.getTimeSearchMode()) {
        case COVER -> {
            // 이 이외의 시간을 전부 빼는 방법으로 구현
            classHoursQuerySize = 24 * 2 * 7 - classHoursSize;
            for (int i = 0; i < classHoursQuerySize; i++)
                jpql += " and l.classHours not like :classHours" + i;
        }
        case INCLUDE -> {
            classHoursQuerySize = classHoursSize;
            jpql += " and (l.classHours like :classHours0";
            for (int i = 1; i < classHoursQuerySize; i++)
                jpql += " or l.classHours like :classHours" + i;
            jpql += ")";
        }
    }
}
}

```

```

if (classHoursQuerySize != 0) {
    query = query.setParameter( name: "emptyClassHours", value: "");
    switch (options.getTimeSearchMode()) {
        case COVER -> {
            Set<Integer> targetClassHours = new HashSet<>(options.getClassHours());
            Integer targetClassHoursConut = 0;
            for (int i = 0; i < classHoursQuerySize; i++, targetClassHoursConut++) {
                while (targetClassHours.contains(targetClassHoursConut))
                    targetClassHoursConut++;
                query = query.setParameter( name: "classHours" + i,
                    value: "%," + targetClassHoursConut.toString() + ",%");
            }
        }
        case INCLUDE -> {
            List<Integer> classHours = options.getClassHours();
            for (int i = 0; i < classHoursQuerySize; i++)
                query = query.setParameter( name: "classHours" + i,
                    value: "%," + classHours.get(i).toString() + ",%");
            }
    }
}
}

```

216,217만 뽑아내기 시행

Query: l.classHours not like %21%

Column: **216,217**

???

✓ 테스트 통과: 1 / 1개 테스트 - 944ms

✓ 테스트 통과: 1 / 1개 테스트 - 944ms

```
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (313:VARCHAR) <- [% ,312,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (314:VARCHAR) <- [% ,313,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (315:VARCHAR) <- [% ,314,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (316:VARCHAR) <- [% ,315,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (317:VARCHAR) <- [% ,316,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (318:VARCHAR) <- [% ,317,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (319:VARCHAR) <- [% ,318,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (320:VARCHAR) <- [% ,319,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (321:VARCHAR) <- [% ,320,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (322:VARCHAR) <- [% ,321,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (323:VARCHAR) <- [% ,322,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (324:VARCHAR) <- [% ,323,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (325:VARCHAR) <- [% ,324,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (326:VARCHAR) <- [% ,325,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (327:VARCHAR) <- [% ,326,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (328:VARCHAR) <- [% ,327,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (329:VARCHAR) <- [% ,328,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (330:VARCHAR) <- [% ,329,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (331:VARCHAR) <- [% ,330,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (332:VARCHAR) <- [% ,331,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (333:VARCHAR) <- [% ,332,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (334:VARCHAR) <- [% ,333,%]
-- [ Test worker] org.hibernate.orm.jdbc.bind : binding parameter (335:VARCHAR) <- [% ,334,%]
```

DAY 1

6월 28일 18:00 ~ 23:00

홍혁준님 | 학부생으로서 어떻게 공부해야 하는가

임지민님 | Selenium : 웹 브라우저 자동화 도구

❓ 연사자의 요청에 따라 일정이 변경될 수 있습니다. - 제로페이지 -

```
time.sleep(2)
```

```
# 없어진 학부를 지우기 위해 교양, 교직만 있거나 코드세어 과목만 있다면 교양 tab이 아닌 이상 지움
```

```
lectures = driver.find_elements(By.CSS_SELECTOR,  
                                "#nbContext > div.nb-contents > div > div.B0_system.nb-q.ng-scope > "  
                                "section.section-gap > div > div > div > div.sp-grid-body > div > div")
```

```
toSave = False
```

```
for lecture in lectures:
```

```
    classification = lecture.find_element(By.CSS_SELECTOR,  
                                          "div:nth-child(5) > div.sp-grid-data.text-center > span > span").t
```

```
    remark = lecture.find_element(By.CSS_SELECTOR,  
                                  "div:nth-child(13) > div.sp-grid-data.text-center > span > span").text
```

```
    if (classification != "교양" and classification != "교직" and  
        remark != "경영경제대학 재학생 및 다전공생의 수강 불가" and remark != "공대 코드세어 과목"  
        # 코드세어 과목은 임시조치로 때문, 비교란으로 잡아내기 때문에 비교 내용 변경되면 추적이 안됨
```

```
) or college == colleges[0]:
```

```
    toSave = True
```

```
    break
```

```
if toSave:
```

```
    print(f"{college.text} {dept.text}: {len(lectures)}")
```

```
    driver.find_element(By.CSS_SELECTOR, "#nbContext > div.nb-contents > div > div.B0_system.nb-q.ng-scope >  
                                          "section.nb-search > div.nb-btn-default.nb-theme > div > div > "  
                                          "button").click()
```

```
    while not glob.glob(originalFilepath):
```

```
        nothing = "to do" # 파일 다운로드 대기
```

```
    os.rename(originalFilepath, os.path.join(downloadPath, f"{college.text}.{dept.text}.xlsx"))
```

```

// URL 열기
String url = "https://mportal.cau.ac.kr/std/usk/sUskSif001/index.do?type=1";
driver.get(url);

// 로그인
wait.until(ExpectedConditions.visibilityOfElementLocated(By.cssSelector("#txtUserID")))
    .sendKeys(userId);
wait.until(ExpectedConditions.visibilityOfElementLocated(By.cssSelector("#txtPwd")))
    .sendKeys(userPw);
driver.findElement(By.cssSelector("#form1 > div > div > div.login-wrap > a")).click();

// 옵션 선택
new Select(driver.findElement(By.cssSelector("#sel_year"))).selectByValue("2024");
new Select(driver.findElement(By.cssSelector("#sel_shtm"))).selectByValue("2");
new Select(driver.findElement(By.cssSelector("#sel_camp"))).selectByValue("1");

// 대학 선택
// 로딩 화면이 보이지 않을 때까지 대기
wait.until(ExpectedConditions.invisibilityOfElementLocated(By.cssSelector(
    "body > div.sp-loading-wrap.none.ng-isolate-scope")));
WebElement collegeSelect = wait.until
    (ExpectedConditions.visibilityOfElementLocated(By.cssSelector("#sel_colg")));
List<WebElement> colleges = collegeSelect.findElements(By.tagName("option"));
for (int i = 1; i < colleges.size(); i++) { // 첫 번째 요소는 비어 있으므로 제외
    WebElement college = colleges.get(i);
    new Select(collegeSelect).selectByValue(college.getAttribute("value"));

    // 로딩 화면이 보이지 않을 때까지 대기
    wait.until(ExpectedConditions.invisibilityOfElementLocated(By.cssSelector(
        "body > div.sp-loading-wrap.none.ng-isolate-scope")));

    // 학과 선택

```

```

getLectureFile.get( userId: "", userPw: "",
    downloadPath: "");

```

downloadPath 상대경로 불가
Windows에서 directory /로 설정하면 안됨
반드시 'ww'로 설정해야 함


```

/**
 * Calculate possible timetables with options.
 * 설정을 기반으로 가능한 시간표를 만들음.
 * @return result of wizard 마법사 결과
 */
public List<Timetable> result(){
    ArrayList<Timetable> result = new ArrayList<>();
    ArrayList<GroupOfGroup> groupOfGroups = new ArrayList<>(groups);
    GroupOfGroup firstGroup = groupOfGroups.remove( index: 0);
    firstGroup.calculateResult( options: this, result, new ArrayList<>(), groupOfGroups);
    return result;
}

```

```

void calculateResult(Wizard options,
                    ArrayList<Timetable> resultList,
                    ArrayList<LectureGroup> selectedGroups,
                    ArrayList<GroupOfGroup> remainingGroups) {
    calculateResult(options, startIndex: 0, numOfGroups, resultList,
                    selectedGroups, remainingGroups);
}

```

```

private void calculateResult(Wizard options, int startIndex, int numOfLeftGroup,
                            ArrayList<Timetable> resultList,
                            ArrayList<LectureGroup> selectedGroups,
                            ArrayList<GroupOfGroup> remainingGroups) {
    for (int i = startIndex; i <= lectureGroups.size() - numOfLeftGroup; i++) {
        ArrayList<LectureGroup> newlySelectedGroups = new ArrayList<>(selectedGroups);
        newlySelectedGroups.add(lectureGroups.get(i));
        if (numOfLeftGroup != 1) {
            //이 GroupOfGroup에서 LectureGroup를 더 골라야 함
            //예를 들어 학교 5개 영역 중 2개를 고르도록 했다면, 하나를 방금 골랐고 나머지 하나를 골라야 함
            calculateResult(options, startIndex: i + 1, numOfLeftGroup: numOfLeftGroup - 1,
                            resultList, newlySelectedGroups, remainingGroups);
        } else {
            //이 group에서 LectureGroup 생성이 끝남(방금 고른 게 마지막)
            if (remainingGroups.size() != 0) {
                //다음 group이 있어 넘겨 주어야 함
                ArrayList<GroupOfGroup> nextRemainingGroups = new ArrayList<>(remainingGroups);
                GroupOfGroup nextGroup = nextRemainingGroups.remove( index: 0);
                nextGroup.calculateResult(options, resultList,
                                          newlySelectedGroups, nextRemainingGroups);
            } else {
                //모든 LectureGroup 생성이 끝남
                ArrayList<LectureGroup> finallySelectedGroups =
                    new ArrayList<>(newlySelectedGroups);
                LectureGroup firstGroup = finallySelectedGroups.remove( index: 0);
                firstGroup.calculateResult(options, resultList,
                                          new ArrayList<Lecture>(),
                                          finallySelectedGroups);
            }
        }
    }
}

```



```

        ArrayList<LectureGroup> selectedGroups,
        ArrayList<GroupOfGroup> remainingGroups) {
    for (int i = startIndex; i <= lectureGroups.size() - numOfLeftGroup; i++) {
        ArrayList<LectureGroup> newlySelectedGroups = new ArrayList<>(selectedGroups);
        newlySelectedGroups.add(lectureGroups.get(i));
        if (numOfLeftGroup != 1) {
            //이 GroupOfGroup에서 LectureGroup를 더 골라야 함
            //예를 들어 학교 5개 영역 중 2개를 고르도록 했다면, 하나를 방금 골랐고 나머지 하나를 골라야 함
            calculateResult(options, startIndex: i + 1, numOfLeftGroup: numOfLeftGroup - 1,
                resultList, newlySelectedGroups, remainingGroups);
        } else {
            //이 group에서 LectureGroup 생성이 끝남(방금 고른 게 마지막)
            if (remainingGroups.size() != 0) {
                //다음 group이 있어 넘겨 주어야 함
                ArrayList<GroupOfGroup> nextRemainingGroups = new ArrayList<>(remainingGroups);
                GroupOfGroup nextGroup = nextRemainingGroups.remove(index: 0);
                nextGroup.calculateResult(options, resultList,
                    newlySelectedGroups, nextRemainingGroups);
            } else {
                //모든 LectureGroup 생성이 끝남
                ArrayList<LectureGroup> finallySelectedGroups =
                    new ArrayList<>(newlySelectedGroups);
                LectureGroup firstGroup = finallySelectedGroups.remove(index: 0);
                firstGroup.calculateResult(options, resultList,
                    new ArrayList<Lecture>(),
                    finallySelectedGroups);
            }
        }
    }
}
}
}

```

```
private void calculateResult(Wizard options, int startIndex, int numOfLeftGroup,
                             List<Lecture> currentTimetableLectureList,
                             List<Timetable> result, List<LectureGroup> remainingGroups) {
    for (int i = startIndex; i <= lectureList.size() - numOfLeftGroup; i++) {
        Lecture targetLecture = lectureList.get(i);

        //강의 추가 가능한지(다른 강의와 겹치거나 option에 맞지 않는 게 있는지) 확인
        //강의 추가 불가시 continue

        //options 확인 및 overlap 점검
        boolean isNotAbleToAdd = false;
        for (Lecture lecture : currentTimetableLectureList) {
            if (targetLecture.isClassOverlaped(lecture)) {
                isNotAbleToAdd = true;
                break;
            }
        }
        if (isNotAbleToAdd) continue;
    }
}
```

```

if (targetLecture instanceof RegisteredLecture registeredLecture) {
    if ((!registeredLecture.getCourseType().isVideo() ||
        !Objects.requireNonNullElse(options.getExcludeVideoLecture(), defaultObj: false)) &&
        (!registeredLecture.getCourseType().isUntact() ||
        !Objects.requireNonNullElse(options.getExcludeLivevideoLectureAll(),
            defaultObj: false))) {
        //조건에 무시되는 게 아니라면
        isNotAbleToAdd = checkIfNotAbleToAddByTime(options, registeredLecture);
        if (isNotAbleToAdd) continue;

        //공강 option
        if (!registeredLecture.getCourseType().isUntact() ||
            !Objects.requireNonNullElse(
                options.getExcludeLivevideoLectureOnEmpty(), defaultObj: false)) {
            //공강 조건 제외가 아니면(이게 걸려 있어서 시작, 종료시간을 먼저 계산한 것)
            isNotAbleToAdd = checkIfNotAbleToAddByEmptyDay(options, registeredLecture);
        }
    }
} else if (targetLecture instanceof CustomLecture) {
    isNotAbleToAdd = checkIfNotAbleToAddByTime(options, targetLecture);
    if (isNotAbleToAdd) continue;
    isNotAbleToAdd = checkIfNotAbleToAddByEmptyDay(options, targetLecture);
}

if(isNotAbleToAdd) continue;

```

```

static boolean checkIfNotAbleToAddByTime
    (Wizard options, Lecture lecture) {
    if (options.getLectureStartTime() != null) { //시작 시간 조건
        for (int j = 0; j < 7; j++) {
            LocalTime lectureStartTime = options.getLectureStartTime();
            int lectureStartTimeForInt = lectureStartTime.getHour() * 2;
            if (lectureStartTime.getMinute() > 30) {
                lectureStartTimeForInt += 2;
            } else if (lectureStartTime.getMinute() > 0) {
                lectureStartTimeForInt++;
            }
            List<Integer> beforeStart = IntStream
                .range(24 * 2 * j, 24 * 2 * j + lectureStartTimeForInt)
                .boxed().collect(Collectors.toList());
            beforeStart.retainAll(lecture.getClassHoursByList());
            if (beforeStart.size() != 0)
                return true;
        }
    }

    if (options.getLectureEndTime() != null) { //종료 시간 조건
        for (int j = 0; j < 7; j++) {
            LocalTime lectureEndTime = options.getLectureEndTime();
            int lectureEndTimeForInt = lectureEndTime.getHour() * 2;
            if (lectureEndTime.getMinute() > 30) {
                lectureEndTimeForInt++;
            }
            List<Integer> afterEnd = IntStream
                .range(24 * 2 * j + lectureEndTimeForInt, 24 * 2 * (j + 1))
                .boxed().collect(Collectors.toList());
            afterEnd.retainAll(lecture.getClassHoursByList());
            if (afterEnd.size() != 0)
                return true;
        }
    }

    return false;
}

```

```
/**  
 * 테스트를 위해 private을 해제함  
 */
```

2개 사용 위치

```
static boolean checkIfNotAbleToAddByEmptyDay(Wizard options, Lecture lecture) {  
    for (DayOfWeek day : options.getEmptyDaysList()) {  
        List<Integer> fullDay = IntStream  
            .range(24 * 2 * (day.getValue() - 1), 24 * 2 * day.getValue())  
            .boxed().collect(Collectors.toList());  
        fullDay.retainAll(lecture.getClassHoursByList());  
        if (fullDay.size() != 0)  
            return true;  
    }  
    return false;  
}
```

```

        ArrayList<Lecture> newTimetableLectureList =
            new ArrayList<>(currentTimetableLectureList);
        newTimetableLectureList.add(targetLecture);
        if (numOfLeftGroup != 1) {
            calculateResult(options, startIndex: i + 1, numOfLeftGroup: numOfLeftGroup - 1,
                newTimetableLectureList, result, remainingGroups);
        } else {
            if (remainingGroups.size() != 0) {
                ArrayList<LectureGroup> nextRemainingGroups =
                    new ArrayList<>(remainingGroups);
                LectureGroup nextGroup = nextRemainingGroups.remove(index: 0);
                nextGroup.calculateResult(options, result,
                    newTimetableLectureList, nextRemainingGroups);
            } else {
                //이제 시간표 하나가 완성됨

                //공강 날짜 수 세는 건 여기서
                //검사 과정이 복잡하니 검사 필요없을 때는 별도로 뺌
                if (options.getNumOfLeastEmptyDay() == 0) {
                    result.add(new Timetable(newTimetableLectureList, numOfLastPriority: 0));
                } else {
                    int numOfEmptyDay = getNumOfEmptyDay(options, newTimetableLectureList);

                    if (numOfEmptyDay >= options.getNumOfLeastEmptyDay())
                        result.add(new Timetable(newTimetableLectureList, numOfLastPriority: 0));
                }
            }
        }
    }
}

```

- [로그인](#)
- [회원 가입](#)
- [시간표 마법사](#)
- [시간표 공유](#)

시간표 마법사 설정

Groups of Groups

Add Group

Add Group



Group Name

Add Group

Add Lecture



Lecture Name

Add Lecture

공강 설정

공강 날짜:

- ☐ MONDAY
- ☐ TUESDAY
- ☐ WEDNESDAY
- ☐ THURSDAY
- ☐ FRIDAY
- ☐ SATURDAY
- ☐ SUNDAY

Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Wed Jul 31 12:14:54 KST 2024

There was an unexpected error (type=Internal Server Error, status=500).

An error happened during template parsing (template: "class path resource [templates/fragments/layout.html]")

org.thymeleaf.exceptions.TemplateInputException: An error happened during template parsing (template: "class path resource [templates/fragments/layout.html]")

```
at org.thymeleaf.templateparser.markup.AbstractMarkupTemplateParser.parse(AbstractMarkupTemplateParser.java:241)
at org.thymeleaf.templateparser.markup.AbstractMarkupTemplateParser.parseStandalone(AbstractMarkupTemplateParser.java:100)
at org.thymeleaf.engine.TemplateManager.parseAndProcess(TemplateManager.java:666)
at org.thymeleaf.TemplateEngine.process(TemplateEngine.java:1103)
at org.thymeleaf.TemplateEngine.process(TemplateEngine.java:1077)
at org.thymeleaf.spring6.view.ThymeleafView.renderFragment(ThymeleafView.java:372)
at org.thymeleaf.spring6.view.ThymeleafView.render(ThymeleafView.java:192)
at org.springframework.web.servlet.DispatcherServlet.render(DispatcherServlet.java:1431)
at org.springframework.web.servlet.DispatcherServlet.processDispatchResult(DispatcherServlet.java:1167)
at org.springframework.web.servlet.DispatcherServlet.doDispatch(DispatcherServlet.java:1106)
at org.springframework.web.servlet.DispatcherServlet.doService(DispatcherServlet.java:979)
at org.springframework.web.servlet.FrameworkServlet.processRequest(FrameworkServlet.java:1014)
at org.springframework.web.servlet.FrameworkServlet.doGet(FrameworkServlet.java:903)
at jakarta.servlet.http.HttpServlet.service(HttpServlet.java:564)
at org.springframework.web.servlet.FrameworkServlet.service(FrameworkServlet.java:885)
at jakarta.servlet.http.HttpServlet.service(HttpServlet.java:658)
at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:195)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:140)
at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:51)
at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:164)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:140)
at org.springframework.web.filter.CompositeFilter$VirtualFilterChain.doFilter(CompositeFilter.java:108)
at org.springframework.security.web.FilterChainProxy.lambda$doFilterInternal$3(FilterChainProxy.java:231)
at org.springframework.security.web.FilterChainProxy$VirtualFilterChain.doFilter(FilterChainProxy.java:365)
```


파일(F) 편집(E) 보기(V) 탐색(N) 코드(C) 리팩터링(R) 빌드(B) 실행(U) 도구(T) Git(G) 창(W) 도움말(H)

timetable - WizardController.java [timetable.main]

timetable > src > main > java > org > zeropage > project > timetable > controller > WizardController > getWizard

프로젝트 WizardController.java Wizard.java settings.html GroupOfGroup.java WebSecurityConfig.java LectureController.java search Gradle

bootstrap-grid.css bootstrap-grid.rtl.css bootstrap-reboot.css bootstrap-reboot.rtl.css bootstrap-utilities.css bootstrap-utilities.rtl.css js templates fragments bodyHeader.html header.html layout.html wizard addCustomLecture.html addgroup.html result.html search.html settings.html home.html login.html signup_form.html timetable.html application.properties application.vml

10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

```
import java.util.List;

@Controller
@RequiredArgsConstructor
@RequestMapping("/wizard")
public class WizardController {

    @GetMapping("/settings")
    @ResponseBody
    public String getWizard(Model model) {
        model.addAttribute("pageContent", "wizard/settings :: wizardSettings");
        model.addAttribute("wizard", new Wizard());
        return "fragments/layout";
    }

    @PostMapping("/settings")
    @ResponseBody
    public String getResult(@ModelAttribute Wizard wizard, Model model) {
        List<Timetable> result = wizard.result();
        model.addAttribute("pageContent", "timetable :: timetableView");
        model.addAttribute("result", result);
        return "fragments/layout";
    }
}
```

실행: TimetableApplication

콘솔 액추에이터

1] o.a.c.c.C.[...].dispatcherServlet : Servlet.service() for servlet [dispatcherServlet] in context with path [] threw exception [Request processing failed: org.thymeleaf...

: Error resolving template [wizard/settings :: wizardSettings], template might not exist or might not be accessible by any of the configured Template Resolvers (template: "fragme...

templateManager.java:869) ~[thymeleaf-3.1.2.RELEASE.jar:3.1.2.RELEASE]

templateManager.java:250) ~[thymeleaf-3.1.2.RELEASE.jar:3.1.2.RELEASE]

resolveExecutedFragmentExpression(FragmentExpression.java:567) ~[thymeleaf-3.1.2.RELEASE.jar:3.1.2.RELEASE]

ientInsertionTagProcessor.computeFragment(AbstractStandardFragmentInsertionTagProcessor.java:399) ~[thymeleaf-3.1.2.RELEASE.jar:3.1.2.RELEASE]

Git 찾기 실행 디버그 TODO 문제 터미널 서비스 Profiler 빌드 Dependencies

CRLF UTF-8 4개 공백 hsworbranch

시간표 마법사 설정

Groups of Groups

Group 1

Remove Group

Group Name

Number of Groups 1

Lecture Groups

Lecture Group \${lter.index + 1}

Remove Lecture Group

Lecture Group Name

Number of Lecture Picks 1

Add Lecture

Add Lecture Group

Add Group

Common Settings

Minimum Number of Empty Days: 0

Lecture Start Time: --:--

Lecture End Time: --:--

- ☐ Exclude Saturday from empty day calculation
- ☐ Exclude video lectures from all conditions
- ☐ Exclude live video lectures only from empty day conditions
- ☐ Exclude live video lectures from all conditions (including empty day conditions)

Save

DevTools is now available in Korean!

Always match Chrome's language

Switch DevTools to Korean

Don't show again

Elements Console Sources Performance insights Network >> 9

top Filter Default levels No Issues

- 17 messages
 - 8 user messages
 - 9 errors
 - No warnings
 - 8 info
 - No verbose
- ☐ Hide network
 - ☐ Preserve log
 - ☐ Selected context only
 - ☒ Group similar messages in console
 - ☒ Show CORS errors in console
 - ☐ Log XMLHttpRequests
 - ☒ Eager evaluation
 - ☒ Autocomplete from history
 - ☒ Treat code evaluation as user action

New group name: settings:116

Uncaught
ReferenceError: bootstrap is not defined
at addGroup (settings:119:23)
at HTMLButtonElement.onclick (settings:53:76)

New group name: settings:116

Uncaught
ReferenceError: bootstrap is not defined
at addGroup (settings:119:23)
at HTMLButtonElement.onclick (settings:53:76)

Uncaught (in promise)
Error: A listener indicated an asynchronous response by returning true, but the message channel closed before a response was received

Console What's new Issues

Highlights from the Chrome 127 update

Enhanced 'Never pause here'

The 'Never pause here' option can now "cancel" DOM, XHR/fetch, CSP violation breakpoints, exceptions and promise rejections from built-in functions, and more.

New scroll snap event listeners



Groups of Groups

Group 1

Remove Group

Group Name

Number of Groups

Lecture Groups

Lecture Group 1

Remove Lecture Group

Lecture Group Name

Number of Lecture Picks

Add Lecture

Add Lecture Group

Add Group

Common Settings

DevTools is now available in Korean!

Always match Chrome's language

Switch DevTools to Korean

Don't show again

PageWorkspace>>:settingsx

▼top

▼localhost:8080

▼wiz

settings

cdn.jsdelivr.net

171
172
173
174
175
176
177
178
179
180
181
182
183

groupsContainer.removeChild(groupsContainer

saveState();

}

function addLectureGroup(groupIndex, name = '',

const container = document.getElementById('

const newLectureGroupIndex = container.chil

// Create a new lecture group fieldset

const newLectureGroupFieldset = document.cr

newLectureGroupFieldset.classList.add('mb-3

newLectureGroupFieldset.innerHTML = '

<legend>Lecture Group \${newLectureGroupInde

15 characters selectedCoverage: n/a

ScopeWatch

▼Breakpoints

Pause on uncaught exceptions

Pause on caught exceptions

▼Call Stack

Not paused

ConsoleWhat's newIssues

Highlights from the Chrome 127 update

Enhanced 'Never pause here'

The 'Never pause here' option can now "cancel" DOM, XHR/fetch, CSP violation breakpoints, exceptions and promise rejections from built-in functions, and more.

New scroll snap event listeners

new

김동우 2

🔍 📞 📺 ☰

📅 2024년 7월 28일 일요일

김동우

<https://everytime.kr/398053/v/349809761>

함께하는 대학생
에브리타임
함께하는 대학생, 에
everytime.kr

ㅋㅋㅋㅋㅋㅋ

김동우

저기 한명은 누군지 아는데 ㅋㅋ

프론트 돌인팀 귀하다...

김동우

📞 보이스톡

김동우

📞 5:39

컴퓨터공학/IT

복수/융합/설계전공&IT정보공유

익명 07/28 18:42

캡스톤 백엔드 팀원 한 분 구

현재 프론트개발자 2명 있으며 이미있는 프로

캡스톤디자인 2

환영합니다. '캡스톤디자인' 오픈채팅방입니

쪽지 신고

협업경험에 있

해주시면 감사

== 다음학기 캡스톤 포함 7전공

👍 0 💬 1 ⭐ 0

👍 공감 ☆ 스크랩

익명(글쓴이)

구했습니다!

07/28 23:17

그중에게 합성
괜찮으면 이 카톡 아이디로 단독 만들어도
될까요?

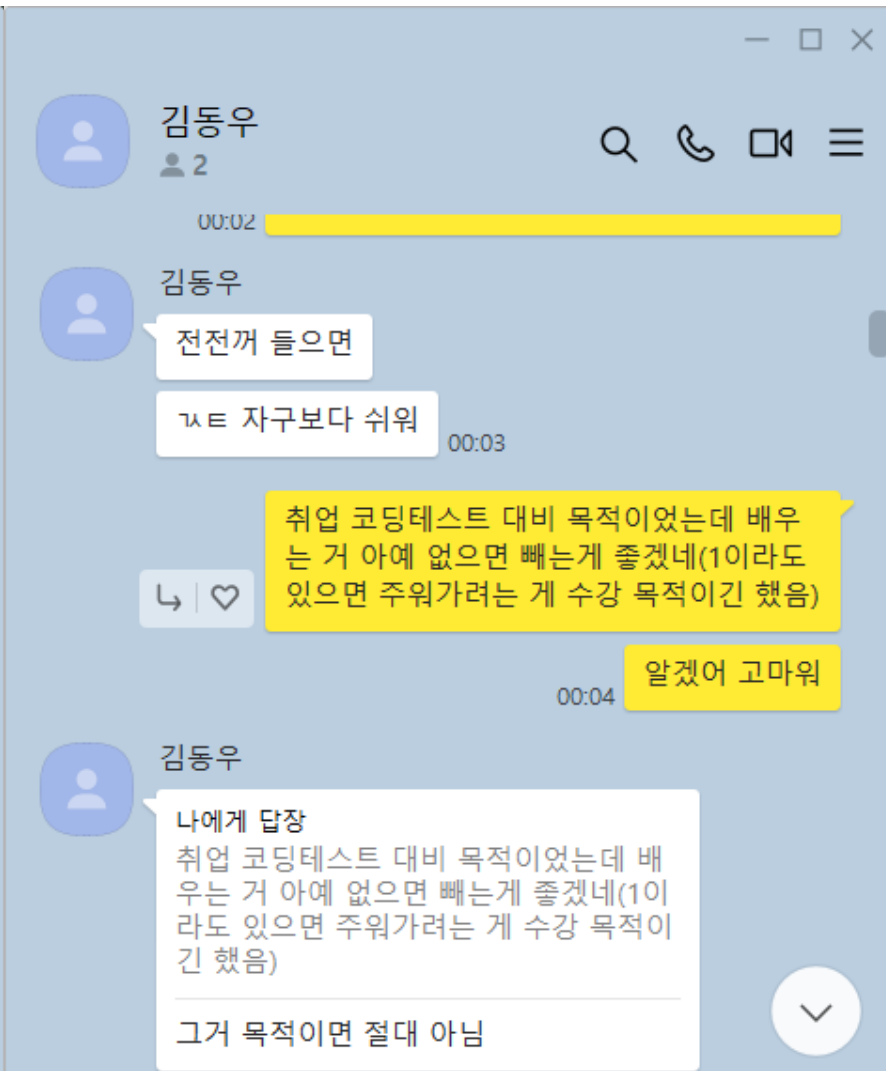
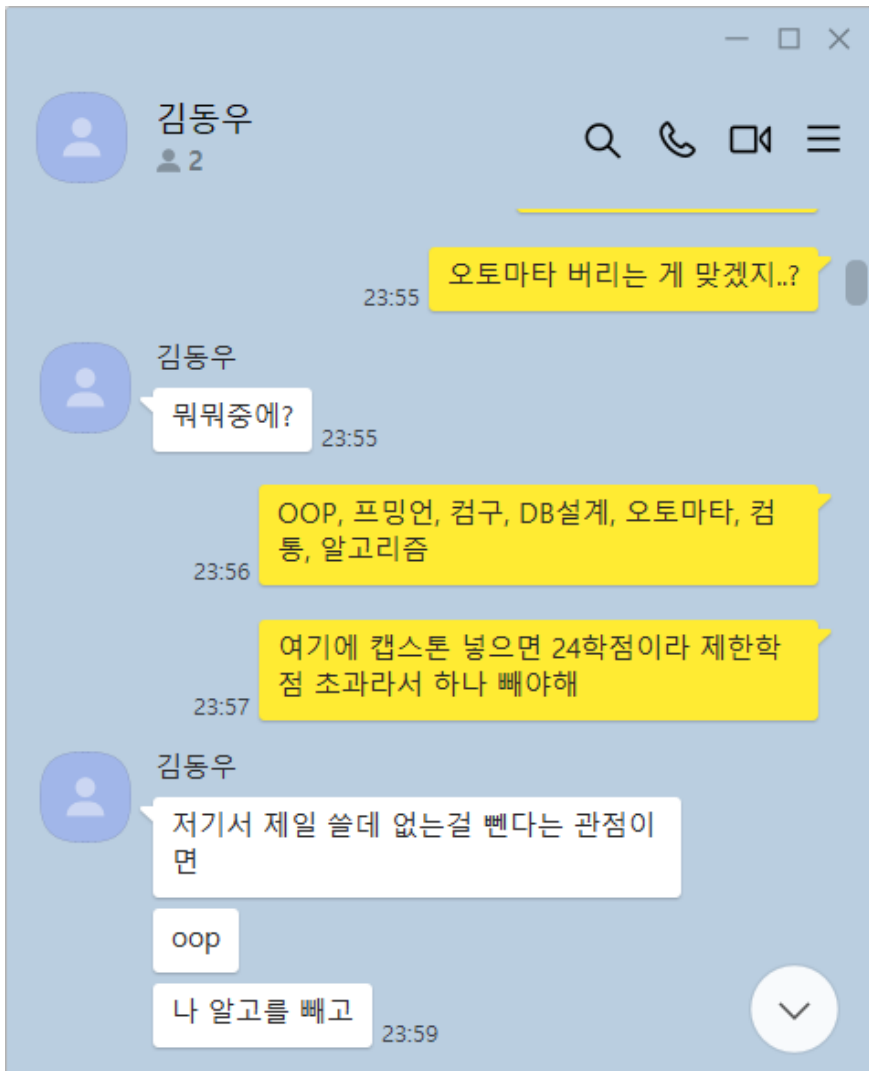
네!

감사합니다!

감 쪽지 신고

댓글을 입력하세요.

☐ 익명 ✎



남들은 주 3회 등교 가능한데...
참고) 이번에 1학년도 가능함

	월	화	수	목	금
9시			객체지향프로그래밍 (영어) 손봉수 310관 728호		
10시	객체지향프로그래밍 (영어) 손봉수 310관 728호				
11시	프로그래밍언어론 김진성 310관 729호		프로그래밍언어론 김진성 310관 729호		
12시		위험사회의웹빙사이언스 김양지 303관(법학관) 2			
13시			+ 컴퓨터통신 + 캡스톤디자인(1) ==공강불가		
14시					
15시	컴퓨터구조 (영어A강의) 310관 729호	ACT 최민지 102관(약학대학)	데이터베이스설계 (영어A) 이재용 310관 723호		
16시					
17시					
18시					
19시			다빈치스포츠 정우영 310관 721호		