|  |  |  |
| --- | --- | --- |
| 기능명세서 | **문서번호** | **AD-003** |
| **작성자** | **3조** |
| **작성일자** | **18.10.04** |

|  |  |  |
| --- | --- | --- |
| **구분** | **이름** | **역할** |
| **패키지** | **com.gm.academy** | **메인 패키지** |
| **클래스** | **MainClass** | **프로그램의 메인 클래스** |
| **메서드** | Search() | ID/PW 찾기 메소드 |
| Login() | 사용자에게 ID와 PW를 받아 학생,교사,관리자로 분기 |
| academyStatus() | 학원현황의 메인 분기문 |
| completeRate() | 수료율을 조회할 수 있는 메소드 |
| showEmploymentRate() | 취업률 조회 메소드 |
| subjectInformation() | 과목정보를 확인할 수 있는 메소드 |
| showRecommendedCompany() | 추천회사를 조회할 수 있는 메소드 |
| showCourseStatus | 수강현황을 조회할 수 있는 메소드 |

|  |  |  |
| --- | --- | --- |
| **구분** | **이름** | **역할** |
| **패키지** | **com.gm.academy.admin** | **관리자 부분 패키지** |
| **클래스** | **AdminController** | **관리자의 전체적인 기능을 담당** |
| **메소드** | main(id,pw) | Id와 pw를 받아 check 메서드에 넘김 |
| Admin\_LectureSubjectResigst() | 과정과 과목을 엮어주는 과정을 진행하는 메소드 |
| totalLog() | 전체로그의 메인 분기문 |
| Logout() | 로그인 이후 담겼던 변수들의 값들을 모두 null로 초기화해주는 과정 |
| textBookManagement() | 교재관리 기능의 메인 분기문 |
| Check(id,pw) | 관리자의 아이디와 비밀번호의 검사 메소드 |
| TeacherManagement() | 관리자의 교사계정을 관리하는 메인 분기문 |
| TeacherUpdateMenu() | 관리자의 교사계정의 수정문 |
| TeacherUpdatetel() | 관리자의 교사의 전화번호를 수정하는 메소드 |
| TeacherUpdateName() | 관리자의 교사의 이름을 수정하는 메소드 |
| TeacherRemove() | 관리자의 교사를 삭제하는 메소드 |
| TeacherRegister() | 관리자의 교사를 등록하는 메소드 |
| TeacherList() | 관리자의 교사정보를 출력하는 메소드 |
| TeacherServed() | 관리자 교사의 재직현황을 출력하는 메소드 |
| Updatesubject() | 관리자가 과목의 정보를 수정하는 메서드 |
| deleteSubject() | 관리자가 과목의 정보를 삭제하는 메소드 |
| addSubject() | 관리자가 과목을 추가하는 메소드 |
| subjectList() | 관리자가 과목현황을 조회하는 메소드 |
| updateAttendance() | 출결정보를 수정하는 메소드 |
| showAttendanceByStudent() | 학생별로 출결정보를 조회할 수 있게 하는 메소드 |
| showAttendanceByDay() | 기간별 출결정보를 조회할수 있게하는 메소드 |
| showAttendanceByLecture() | 과정별로 출결정보를 조회할 수 있게 하는 메소드 |
| Admin\_LectureUpdate() | 관리자파트로써 과정 정보를 수정할 수 있게하는 메소드 |
| LectureTeacherUpdate() | 과정관리에서 교사코드를 수정하는 메소드 |
|  | LectureClassroomUpdate() | 과정관리에서 강의실을 수정하는 메소드 |
| LectureStudentUpdate() | 과정관리에서 학원인원을 수정하는 메소드 |
| LectureProgressUpdate() | 과정관리에서 강의진행여부를 수정하는 메소드 |
| LectureDateUpdate() | 과정관리에서 과정 날짜를 수정할 수 있는 메소드 |
| LectureNameUpdate() | 과정관리에서 과정 이름을 수정할 수 있는 메소드 |
| Admin\_LectureRemove() | 관리자파트에서 과정정보를 삭제하는 메소드 |
| Admin\_LectureRegist() | 관리자파트에서 과정정보를 등록하느 메소드 |
| Admin\_LectureCheck() | 관리자 파트에서 과정정보의 현황을 조회하는 메소드 |
| RecommendedCompany() | 추천회사를 관리하는 메인 분기문 |
| updateRecommendedCompany() | 추천회사 목록에서 추천회사의 내용을 수정하는 메소드 |
| removeRecommendedCompany() | 추천회사 목록에서 추천회사의 내용을 삭제하는 메소드 |
| addRecommendedCompany() | 추천회사 목록에서 추천회사를 추가하는 메소드 |
| studentMange() | 관리자 교육생 관리 파트의 메인 분기문 |
| Oversight() | 관리자 파트의 사후관리의 메인 분기문 |
| updateOversight() | 관리자 파트의 사후관리 정보를 수정하는 메소드 |
| selectOversight() | 사후관리 정보를 조회할 수 있게 출력해주는 메소드 |
| insertOversight() | 사후관리 데이터를 입력하는 메소드 |
| deleteOversight() | 사후관리 정보를 삭제하는 메소드 |
| Counseling() | 상담일지 관리의 메인 분기문 |
| updateCounseling() | 상담일지 정보를 수정하는 메소드 |
| deleteCounseling() | 상담일지 정보를 삭제하는 메소드 |
| InsertCounseling() | 상담일지 정보를 추가하는 메소드 |
| stdUpdate() | 교육생정보를 수정하는 메소드 |
| stdDelete(seq ,stdName) | 교육생 정보를 삭제하는 메소드 |
| stdInsert(stdSeq) | 교육생 정보를 등록하는 메소드 |
| Llist() | 과정코드와 과정명을 출력해주는 메소드 |
| stdList(stdSeq) | 교육생의 일련번호와 이름을 조회할 수 있게 해주는 메소드 |
| CourseRecord() | 학생코드를 입력받아 상담일지에 내용을 추가하는 메소드 |
| insertList() | 교육생이 들어갈 수 있는 과정정보를 보여주는 메소드 |
| selectCounseling() | 상담일지를 조회하게 해주는 메소드 |
| counselingStList(stdseq) | 교육생 일련번호를 입력받아 해당 교육생의 상담일지 내용을 조회할 수 있게 해주는 메소드 |
| stdFind(LectureSeq,stdName) | 해당 과정을 듣는 교육생 정보를 출력하는 메소드 |
| **메소드** | updateGrade() | 성적 수정하기 메인 분기문 |
| studentTestInfo() | 시험관리 및 성적조회, 학생별 성적조회 |
| updateTest() | 시험 관리 및 성적 조회, 시험정보 수정하기 메인 분기문 |
| Admin\_ScoreList() | 과정 별 시험 성적 조회 |
| **클래스** | **AdminDAO** | **관리자 파트에서 연동된 DB에 쿼리를 전달하는 클래스** |
| **메소드** | systemError(String logCode,String part) | erorrLog코드 발생을 등록 |
| auth(AdminLogInDTO alDTO) | 관리자 > 로그인 인증 메소드 |
| textBookManagement() | 과목명,교재명,저자,가격 출력 교재관리 전체조회 메소드 |
| textBookLecture(String lec) | 과목명,교재명,저자,가격 출력 교재관리 과정조회 메소드 |
| lectureNumber() | 과정명 과정번호 출력, 교재관리 과정조회 메소드 |
| **메소드** | Subject() | 과목명, 과목코드 출력 교재관리\_교재등록 |
| Publisher() | 출판사명, 출판사코드 출력 교재관리\_교재등록 |
| textBookApplicationWrite(TextBookDTO tbDTO) | 교재관리\_교재신청하기 메소드 |
| List() | 교사 계정 관리 > 교사 재직 현황 메소드 |
| Insert(String tchname, String tchssn, String tchtel) | 교사 계정 관리 > 교사 등록 메소드 |
| Remove(String tchId, String tchName) | 교사 계정 관리 > 교사 삭제 메소드 |
| updateName(String tchNameBefor, String tchNameAfter) | 교사 계정 관리 > 교사 정보 수정 메소드 |
| updateTel(String tchTelBefor, String tchTelAfter) | 교사 수정 전화번호 메소드 |
| checkSsn(String tchssn) | 교사 계정 관리 > 교사 등록 (주민번호 유효성 검사) |
| Subjectlist() | - 개설 과목 관리 > 과목 현황 조회 메소드  - 과목 목록(강의중인 과정에 속한) |
| addsubject(SubjectDTO subject) | 개설 과목 관리 > 과목 등록 메소드 |
| deletesubject(SubjectDTO subject) | 개설과목 관리 > 과목 삭제 메소드 |
| updatesubject(SubjectDTO subject) | 개설 과목 관리 > 과목 수정 메소드 |
| studentSubjectList(String studentSeq, String studentName) | 성적 조회(필기, 실기, 출결) 메소드 |
| studentGradeInfo(String studentSeq, String studentName, String input) | 학생 과목별 성적 출력 정보 메소드 |
| getLecture() | 과정별 출결조회시 진행중인 과정 정보 메소드 |
| getAttendanceByDay(String year, String month, String day) | 기간별 출결 조회 메소드 |
| getStudentDTO(String stdName) | 출결 수정시 학생명을 정보 확인 메소드 |
| getAttendanceByStudent(String stdSeq) | 특정 교육생 출결 정보 데이터 반환 메소드 |
| updateAttendance(String stdSeq, String date, String situation) | 출결 정보 수정 메소드 |
| LectureList() | 과정 목록 반환 메소드 |
| LectureDetail(String lecSeq) | 과정 내용 상세 리턴 메소드 |
| lectureRegister(LectureDTO lecDTO) | 과정 등록 메소드 |
| LectureRemove(LectureDTO lecDTO) | 과정 삭제 메소드 |
| updateLectureName(LectureDTO lecDTO) | 과정 명 수정 메소드 |
| updateLectureDate(LectureDTO lecDTO) | 과정 일자 수정 메소드 |
| updateLectureProgress(LectureDTO lecDTO) | 과정 진행여부 수정 메소드 |
| updateClassRoom(LectureDTO lecDTO) | 과정의 강의실 정보 수정 메소드 |
| updateTeacher(LectureDTO lecDTO) | 과정의 담당 교사 수정 메소드 |
| updateTchId(String tchNameAfter) | 교사 수정 메소드 |
| getLecture(String str) | 과정 정보를 반환하는 메소드 |
| getRecommendedCompany() | 추천회사 정보를 반환하는 메소드 |
| **메소드** | addRecommendCompany(String name, String location, String payment) | 추천회사의 정보를 받아 DB에 저장 |
| updateRecommendedCompany(String name, String location, String payment, int seq) | 추천회사의 일련번호를 받아 삭제하는 메소드 |
| getLectureAndTeacher() | 추천회사의 정보를 받아 정보를 수정 |
| getSubject(String seq) | 과목 seq를 받아 과목정보를 반환해주는 메소드 |
| getAllTeacher() | 모든 교사의 정보를 반환해주는 메소드 |
| getAllLecture() | 쿼리를 날려 모든 과정의 정보를 반환해주는 메소드 |
| getEmploymentRate(String sel) | 과정번호를 받아 해당 과정의 취업률을 반환해주는 메소드 |
| getCompletionRate(String sel) | 과정번호를 받아 해당 과정의 수료율을 반환하는 메소드 |
| get(String seq) | 과정번호를 받아 해당 과정의 객체를 반환 |
| vwfind(String seq, String name) | 교육생 정보 반환 메소드 |
| insert(StudentDTO std) | 교육생 등록 메소드 |
| add(StudentDTO std, String seq) | 수강내역 교육생 등록 메소드 |
| stdCheck(String name) | 교육생 존재 여부 메소드 |
| delete(String name) | 교육생 삭제 메소드 |
| deleteCourse(String seq, String name) | 상담일지 삭제 메소드 |
| lList(boolean isAuth) | 전체 수강(과정) 목록 출력 메소드 |
| insertList(boolean b) | 교육생 등록 과정 선택 목록 메소드 |
| cousnselingList(int seq) | 교육생 상담일지 리스트 반환 메소드 |
| stdList(String seq) | 교육생 해당 과정 교육생 출력 메소드 |
| courseRecordStd(String stdseq) | 교육생 상담일지 정보 출력 메소드 |
| find(String seq, String name) | 교육생 과정 수강정보 존재 여부 메소드 |
| stdName(String name) | 교육생 이름 존재 여부 메소드 |
| nameUpdate(String name,String nameUpdate) | 교육생 이름 수정 메소드 |
| stdSsn(String ssn) | 교육생 주민번호 존재 여부 |
| ssnUpdate(String ssn, String ssnUpdate) | 교육생 주민번호 존재 수정 메소드 |
| stdTel(String tel) | 교육생 전화번호 존재 여부 메소드 |
| telUpdate(String tel, String telUpdate) | 교육생 전화번호 정보 수정 메서드 |
| vwStd(String input) | 교육생 과정 정보 반환 메서드 |
| insetCourseRecord(CourseRecordDTO courseRecord) | 상담일지 등록 메서드 |
| deleteCourseRecord(String data, String input) | 상담일지 삭제 메서드 |
| updateCourseRecord(String input, String data, String contents) | 상담일지 수정하기 메서드 |
| OversightCnt(String lectureseq, String stdseq) | 사후처리 조회 메서드 |
| selectStdManage(String lectureseq, String stdseq) | 사후 관리 조회 메서드 |
| vwfind2(String input, String seq) | 교육생 정보 및 과정 정보 메서드 |
| **메소드** | insertOversight(StudentManageDTO dto) | 사후 관리 등록 메서드 |
| deleteOversight(String stdseq) | 사후관리 삭제 메서드 |
| updateOversight(String stdseq, String company) | 사후관리 등록 메서드 |
| updateNoteGrade(StudentGradeInfoDTO studentgradeinfo) | 필기 점수 수정 메서드 |
| updateSkillGrade(StudentGradeInfoDTO studentgradeinfo) | 실기 점수 수정 메서드 |
| updateAttendanceGrade(StudentGradeInfoDTO studentgradeinfo) | 출석 점수 수정 메소드 |
| noteTestList(String subjectname) | 필기 문제 리스트 조회 메서드 |
| updateNoteTestQuest(String questnum, String quest, String distribution) | 필기 문제 수정 메서드 |
| skillTestList(String subjectname) | 실기문제 리스트 조회 메서드 |
| updateSkillTestQuest(String questnum, String quest, String distribution) | 실기 문제 수정 메소드 |
| updateDistribution(DistributionDTO distribution) | 배점 수정 메소드 |
| ScoreList(String stdSeq) | 과정별 시험 성적 조회 메서드 |
| LectureListDTO() | 과정, 학생번호, 학생 명 리스트 출력 점수가 0인 학생에 대한 출력 제외 메서드 |
| ErorrLog() | 에러 로그 목록 메소드 |
| Log() | 로그 목록을 반환하는 메소드 |
| showCurrentLecture() | 현재 강의 예정인 과정목록을 출력함. |
| LectureSubjectList<String lectureSeq) | 전체 과목 리스트를 반환 |
| textBookList(String subjectSeq) | 선택 과정 과목에 대한 교재 리스트를 반환하는 메소드 |
| registLectureSubject(LectureSubjectDTO lecSubReg) | LectureSubject 테이블로 insert |
| LoginLog() | 관리자 로그인 기록 저장 |
| LogoutLog() | 관리자 로그아웃 기록 저장 |
| **메소드** | subjectList(String lectureSeq) | 선택 과정의 현재 과목 리스트를 반환 |
| AdminLoginLog() | 관리자 로그인 기록 리스트 메소드 |
| TeacherLoginLog() | 교사 로그인 로그 리스트 메소드 |
| StudentLoginLog() | 교육생 로그인 로그 리스트 메소드 |
| **클래스** | **AdminLoginDTO** | **관리자 로그인 정보를 담고있는 클래스** |
| **CourseRecordListDTO** | **상담일지 정보를 담고 있는 클래스** |
| **DistributionDTO** | **배점 정보를 담고 있는 클래스** |
| **ErrorLogDTO** | **에러정보를 담고 있는 클래스** |
| **LogDTO** | **로그정보를 담고 있는 클래스** |
| **LoginDTO** | **로그인 정보를 담고 있는 클래스** |
| **RecommendCompanyDTO** | **추천회사 정보를 담고있는 클래스** |
| **TextBookManagementDTO** | **교재관리 정보를 담고있는 클래스** |
| **VwStudentDTO** | **학생객체 정보를 담고있는 클래스** |
| **DetailScoreListDTO** | **학생의 상세 점수를 담고있는 클래스** |
| **구분** | **이름** | **역할** |
| **패키지** | **com.gm.academy.Util** | **프로그램의 유틸관련 클래스 모음** |
| **클래스** | **BreadCrumb** | **사용자의 프로그램 사용과정을 출력해주는** |
| **메서드** | In(title) | 제목을 스택에 삽입하는 메소드 |
| Out() | 스택에서 pop해주는 메소드 |
| Now() | 현재 스택의 상태를 출력해주는 메소드 |
| Size() | 현재 스택의 길이를 반환하는 메소드 |
| **클래스** | **DBUtil** | **DataBase와의 연동을 담당하는 클래스** |
| **메소드** | getConnection(server,id,pw) | IP주소와 DB유저, pw를 받아 DB에 연동시켜주는 메소드 |
| **클래스** | **TeacherUtil** | **교사부분의 유틸 데이터 모음 클래스** |
| **메소드** | getLectureDTO() | 현재 강의중인 과정의 이름을 가진 객체배열을 반환 |
| setDistriBution() | 배점정보를 메모리상에 저장시켜주는 메소드 |
| myPage() | 교사의 정보를 제목과 함께 출력시켜줄 메소드 |
| **클래스** | **UtilPrint** | **출력을 담당하는 클래스** |
| **메소드** | Menu() | 메뉴 출력 메소드 |
| Bar() | 선 긋기 메소드 |
| Bar(int size) | 길이만큼 선 긋기 메소드 |
| Title(String label) | 제목을 출력시켜주는 메소드 |
| Pause(String label) | Scanner 객체의 next()메소드를 실행하여 일시정지 효과를 부여하는 메소드(멘트설정가능) |
| Pause() | Scanner 객체의 next()메소드를 실행하여 일시정지 효과를 부여하는 메소드(멘트고정) |
| Header(String[] labels) | 데이터의 헤더를 출력하는 메소드 |
| Data(Object[] datas) | 데이터를 출력하는 메소드 |
| Result(String label) | 단순 결과를 출력하는 메소드 |
| Result(int result, String label) | 조건에 따른 결과를 출력해주는 메소드 |
| **클래스** | **UtilScanner** | **입력을 담당하는 클래스** |
| **메소드** | Next(String label) | 입력제목을 받아 스캐너를 실행하는 메소드 |
| Nextln(String label) | 입력제목을 받아 라인을 띄우고 스캐너를 실행하는 메소드 |
| nextInt(String label) | 인트 값을 입력받는 메소드 |
| nextlnInt(String label) | 인트 값을 한칸 띄우고 입력받는 메소드 |

|  |  |  |
| --- | --- | --- |
| **구분** | **이름** | **역할** |
| **패키지** | **com.gm.academy.exam** | **시험 관련 정보를 포함하는 패키지** |
| **클래스** | **GradeDTO** | **성적정보를 담고있는 클래스** |
| **클래스** | **NoteTestDTO** | **필기문제 정보를 담고있는 클래스** |
| **클래스** | **NoteTestGradingDTO** | **필기테스트 정보를 담고있는 클래스** |
| **클래스** | **SkillTestDTO** | **실기 문제 정보를 담고있는 클래스** |
| **클래스** | **SkillTestGradingDTO** | **실기 테스트 정보를 담고있는 클래스** |

|  |  |  |
| --- | --- | --- |
| **구분** | **이름** | **역할** |
| **패키지** | **com.gm.academy.lecture** | **과정 정보를 포함하는 패키지** |
| **클래스** | **ClassroomDTO** | **강의실 정보를 담고 있는 클래스** |
| **클래스** | **CourseDTO** | **수강내역 정보를 담고 있는 클래스** |
| **클래스** | **LectureDTO** | **과정 정보를 담고 있는 클래스** |
| **클래스** | **LectureSubjectDTO** | **과정과목 정보를 담고 있는 클래스** |
| **클래스** | **PublisherDTO** | **출판사 정보를 담고 있는 클래스** |
| **클래스** | **SubjectDTO** | **과목 정보를 담고 있는 클래스** |
| **클래스** | **TextBookDTO** | **교재 정보를 담고 있는 클래스** |
| **클래스** | **LectureDAO** | **과정관련 쿼리를 DB에 전달하는 기능을 담고있는 클래스** |
| **메서드** | getLectureNameList() | 현재 강의중인 과정의 이름을 가진 ArrayList<String>을 반환해주는 메소드 |
| getLectureDTO() | 현재강의중인 과정객체를 담은 ArrayList<LectureDTO>을 반환해주는 메소드 |

|  |  |  |
| --- | --- | --- |
| **구분** | **이름** | **역할** |
| **패키지** | **com.gm.academy.student** | **학생관련 기능을 포함하는 패키지** |
| **클래스** | **AbsenceRecordDTO** | **근태정보를 담고 있는 클래스** |
| **AttendanceDTO** | **출석 정보를 담고 있는 클래스** |
| **CourseRecordDTO** | **상담일지 정보를 담고 있는 클래스** |
| **DropOutDTO** | **교육생의 중도탈락 정보를 담고 있는 클래스** |
| **LectureCompleteDTO** | **교육생의 과정수료 정보를 담고 있는 클래스** |
| **StudentLectureDTO** | **학생의 과정 정보를 담고 있는 클래스** |
| **StudentAttendanceDTO** | **학생의 출결 정보를 담고 있는 클래스** |
| **StudentDTO** | **학생 정보를 담고 있는 클래스** |
| **StudentGradeDTO** | **학생의 성적 정보를 담고 있는 클래스** |
| **StudentLoginDTO** | **학생의 아이디,비밀번호 정보를 담고 있는 클래스** |
| **StudentManageDTO** | **학생관리 정보를 담고 있는 클래스** |
| **AbsenceRecordTypeDTO** | **근태정보의 타입 정보를 담고 있는 클래스** |
| **StudentDAO** | **학생관련 기능에서 DB에 쿼리를 전달하는 기능 모음 클래스** |
| **메서드** | Auth(StudentLoginDTO slDTO) | 교육생의 로그인을 인증하는 메소드 |
| Grade() | 교육생의 성적을 조회하는 메소드 |
| inOutRegister(String today,int cate) | 교육생의 입실과 퇴실을 등록할수 있는 메소드 |
| lectureEvaluation(String score, String command) | 교사의 과정을 평가하는 메소드 |
| subjectEvaluation(String name, String score,String command) | 과목을 평가할수 있는 메소드 |
| Consulting() | 교육생의 상담일지를 조회하는 메소드 |
| attendanceStatus() | 현재 교육생의 출결정보를 반환하는 메소드 |
| Search(StudentDTO sDTO) | ID와 PW를 찾아내어 반환하는 메소드 |
| LoginLog() | 학생 로그인 기록 저장 |
| LogoutLog() | 학생 로그아웃 기록 저장 |
| **클래스** | **StudentController** | **학생파트의 전체적인 기능을 담당하는 클래스** |
| **메소드** | main(String id, String pw) | 아이디와 비밀번호를 받아 교육생측의 주메서드 |
| Logout() | 로그아웃을 담당하는 메소드 |
| Consulting() | 상담일지의 메인 분기문 |
| teacherEvaluation() | 교사평가의 메인 분기문 |
| teRegister() | 교사평가를 등록하는 매소드 |
| subjectEvaluation() | 과목평가를 진행하는 메소드 |
| lectureEvaluation() | 과정평가를 진행하는 메소드 |
| Attendance() | 출결(입실,퇴실,출결현황)을 관리하는 메인 분기문 |
| attendanceStatus() | 출결현황을 확인하는 메소드 |
| inOutRegister(int cate) | 입실등록을 하는 메소드(cate 입실,퇴실 분기메소드) |
| gradeSearch() | 실기/필기를 분기하여 성적을 확인하는 메소드 |
| check(String id, String pw) | ID/PW를 인증하는 메서드 |
| myPage() | 항시출력하는 마이페이지 메소드 |
| search(String name, String telCode, int cate) | 교육생 ID/PW찾기 |

|  |  |  |
| --- | --- | --- |
| **구분** | **이름** | **역할** |
| **패키지** | **com.gm.academy.teacher** | **교사 관련 기능정보를 포함하는 패키지** |
| **클래스** | **LectureEvaluationDTO** | **과정 평가 정보를 담고 있는 클래스** |
| **클래스** | **PossibleLecDTO** | **교사의 강의 가능 과목 정보를 담고있는 클래스** |
| **클래스** | **SubjectEvaluationDTO** | **과목 평가 정보를 담고있는 클래스** |
| **클래스** | **TeacherDTO** | **교사 정보를 담고있는 클래스** |
| **클래스** | **TeacherLoginDTO** | **교사 로그인 정보를 담고있는 클래스** |
| **클래스** | **TeacherSelectDTO** | **출력할 교사정보를 담고 있는 클래스** |
| **클래스** | **TeacherController** | **교사 파트의 메인 기능을 담고 있는 클래스** |
| **메서드** | Main(id,pw) | 아이디와 비밀번호를 받는 교사측의 주 메소드 |
| Logout() | 로그아웃을 하는 메소드 |
| Check() | 교사의 아이디와 비밀번호의 검사 메소드 |
| showSchedule() | 교사의 강의스케줄 정보를 출력해주는 메소드 |
| examManagement() | 교사의 시험관리파트의 메인 분기문 |
| updateExam() | 시험문제를 수정하는 메소드 |
| removeExam() | 시험문제를 삭제하는 메소드 |
| addExam() | 시험문제를 등록하는 메소드 |
| showExam() | 시험문제를 조회하는 메소드 |
| gradeManagement() | 교사의 성적관리 파트의 메인 분기문 |
| showGrade() | 학생성적을 조회하는 메소드 |
| addGrade() | 교사가 학생성적을 등록하는 메소드 |
| distributionManagement() | 배점관리를 진행하는 메소드 |
| Attendance() | 출결현황조회의 메인 분기문(기간/과정) |
| showAttendanceByLecture() | 과정별 출결조회를 담당하는 메소드 |
| showAttendanceByDay() | 기간별 출결조회를 담당하는 메소드 |
| teacherEvaluation() | 교사평가를 조회할 수 있게 해주는 메소드 |
| Consulting() | 상담일지의 작성, 조회를 담당하는 메소드 |
| Search(String name, String telCode, int cate) | 교사의 아이디/비밀번호를 찾을 수 있게 하는 메소드 |
| **클래스** | **TeacherDAO** | **교사와 관련된 기능에서 DB에 쿼리를 전달하는 기능 모음 클래스** |
| **메소드** | Login(String id, String pw) | 아이디와 비밀번호를 받아 값이 존재하면 1을 반환하는 메소드 |
| showAttendanceByDay(String year, String month, String day) | 기간별 학생 출결사항을 조회하는 쿼리를 DB에 전달하여 ArrayList<Object[]>로 반환해주는 메소드 |
| showAttendanceByLecture(LectureDTO selectedLecture) | 과정별 학생 출결사항을 조회하는 쿼리를 DB에 전달하여 ArrayList<Object[]>로 반환해주는 메소드 |
| addCourseRecord(String stdSeq, String lectureSeq, String content) | 상담일지를 등록하는 쿼리를 DB에 전달하여 삽입된 행의 개수를 반환. |
| showCourseRecord(String stdSeq, String lectureSeq) | 상담일지를 조회하는 쿼리를 DB에 전달하여 ArrayList<Object[]>로 반환해주는 메소드 |
| showTeacherEvaluation() | 교사평가를 조회하는 쿼리를 DB에 전달하고 반환값을 ArrayList<Object[]>로 반환해주는 메소드 |
| getDistribution() | 배점의 조회문을 DB에 전달하고 값을 가진 객체를 반환하는 메소드 |
| setDistribution(int note, int skill, int attendance) | 배점정보를 수정하는 쿼리를 DB에 전달해주는 메소드 |
| addGrade(String stdSeq, int note, int skill, int attendance, String subjectSeq) | 성적정보를 DB상에 등록하는 메소드, 행의 개수 반환. |
| getGradeDTO(String selectedLectureSeq, String selectedStdSeq) | 과정코드,학생코드를 받아 해당 학생(과정)의 성적정보 객체를 반환하는 메소드 |
| getGradeDTO(String stdSeq) | 학생코드를 받아 성적을 입력하는 메소드 |
| getLectureSchedule() | 과정스케줄 정보를 반환해주는 메소드 |
| getSubjectList(String tchSeq) | 해당 교사의 과목리스트를 반환해주는 메소드 |
| getQuestionList(String subjectSeq , int k ) | K는 실기와 필기를 나누는 분기변수이고, 이를 통해 해당 과목의 시험문제 정보를 반환해준다. |
| addExam(String question, String distribution, String subjectSeq, int k) | K는 실기와 필기를 나누는 분기변수이고, 이를 통해 시험문제를 등록하는 메소드 |
| getExamList(String subjectSeq , int k) | K는 필/실 분기변수이고, 이를 통해 해당 시험정보를 반환해준다. |
| removeExam(String examSeq, int k) | 분기변수 k , 시험정보를 삭제하는 메소드 |
| updateExam(String examSeq, String examContent, int k ) | 분기변수 k, 시험문제를 수정하는 메소드 |
| getSubjectAndSeqList(String tchSeq) | 해당 교사의 과목명과 과목번호를 반환해주는 메소드 |
| getStudentDTO(String tchSeq) | 해당 교사가 가르치는 학생명과 학생번호를 반환해주는 메소드 |
| Search(TeacherDTO tDTO) | 해당 교사객체의 아이디와 비밀번호를 반환해주는 메소드 |
| **메소드** | lectureList(String seq) | 강의중인 과정의 목록을 반환 |
| LoginLog() | 교사 로그인 기록 저장 |
| logoutLog() | 교사 로그아웃 기록 저장 |