
학사관리 시스템 DAO 명세서

조 번호 : 08조
201402412 임형섭
201402453 황동준
201402368 심규영

클래스 명	User_login
클래스 설명	학생과 관리자의 목록을 가지고 있는 class, id와 password 받아 로그인하는 method를 가지고 있다..

1. 속성

속성 명	타입	논리모델 속성명	설명
std_struct	Struct(구조체)	학생 정보	학생의 계정정보를 담고 있다
adm_struct	Struct(구조체)	관리자 정보	관리자의 계정정보를 담고 있다
adm_id	String	관리자 아이디	관리자의 아이디를 나타내는 속성
std_id	String	학생 학번	학생의 학번을 나타내는 속성
std_pwd	String	학생 비밀번호	학생의 비밀번호를 나타내는 속성
adm_pwd	String	관리자 비밀번호	관리자의 비밀번호를 나타내는 속성
std_list	Arraylist	학생 목록	학생들을 담고 있는 목록
adm_list	Arraylist	관리자 목록	관리자를 담고 있는 목록

2. 오퍼레이션 목록

이름	관리자정보 불러오기
영문명	Adm_load
설명	관리자의 정보를 외부 파일로부터 가져와 파싱한 후 처리해 준다.
파라미터	
반환값 및 타입	관리자목록(Arraylist) <ul style="list-style-type: none"> Adm_list : 정상적으로 목록을 불러온 경우 값이 들어간 Adm_list를 반환 Adm_list : 정상적이지 못한 경우 비어있는 Adm_list를 반환
개요	외부파일로부터 이미 저장되어 있는 관리자나 학생의 정보를 불러오는 함수.
알고리즘	<pre> try{ file filereader, bufferedReader 사용해서 파일을 불러오고 불러온 파일을 줄마다 파싱하고 파싱한정보를 구조체 담아 구조체를 list에 저장 한다 list를 반환 } catch{ 예외처리} </pre>

이름	학생정보 불러오기
영문명	Std_load
설명	학생의 정보를 외부 파일로부터 가져와 파싱한 후 처리해 준다.
파라미터	
반환값 및 타입	관리자목록(Arraylist) <ul style="list-style-type: none"> Std_list : 정상적으로 목록을 불러온 경우 값이 들어간 Std_list를 반환 Std_list : 정상적이지 못한 경우 비어있는 Std_list를 반환
개요	외부파일로부터 이미 저장되어 있는 관리자나 학생의 정보를 불러오는 함수.
알고리즘	<pre> try{ file filereader, bufferedReader 사용해서 파일을 불러오고 불러온 파일을 줄마다 파싱하고 파싱한정보를 구조체 담아 구조체를 list에 저장 한다 list를 반환 } catch{ 예외처리} </pre>

이름	관리자 로그인
영문명	Adm_login
설명	관리자가 로그인한다.
파라미터	1. p_id (아이디) 2. p_pwd (비밀번호)
반환값 및 타입	로그인결과(Boolean) <ul style="list-style-type: none"> "True" : 결과값이 관리자인 경우 True를 반환 "False" : 값이 존재하지 않을 경우 False를 반환
개요	등록되어 있는 arraylist에서 입력 받은 값들 중 일치하는 id를 찾고 id가 존재할 경우 password를 확인 후 일치 시 로그인 시킨다.
알고리즘	<pre> if(관리자 list에서 id 확인 && 관리자 list에서 pwd확인) { return True } return False </pre>

이름	학생 로그인
영문명	Std_login
설명	관리자와 학생이 로그인한다.
파라미터	3. p_id (아이디) 4. p_pwd (비밀번호)
반환값 및 타입	로그인결과(Boolean) <ul style="list-style-type: none"> • "True" : 결과값이 학생인 경우 True를 반환 • "False" : 값이 존재하지 않을 경우 False를 반환
개요	등록되어 있는 arraylist에서 입력 받은 값들 중 일치하는 id를 찾고 id가 존재할 경우 password를 확인 후 일치 시 로그인 시킨다.
알고리즘	<pre> for(학생 리스트 크기만큼) { if(학생 list에서 id 확인 && 학생 list에서 pwd확인) return True } return False </pre>

클래스 명	Scholarship
클래스 설명	장학생 관련 업무 처리

1. 속성

속성 명	타입	논리모델 속성명	설명
std_name	string	학생 이름	학생의 이름을 나타내는 속성
std_id	string	학생 학번	학생의 학번을 나타내는 속성
std_type	string	선발유형	학생의 장학 선발 유형을 나타내는 속성
std_money	string	금액	학생의 장학금액을 나타내는 속성
std_sch_list	Arraylist	장학생 목록	장학생들을 담고 있는 목록
sch_list	Arraylist	학생 장학 정보	학생 개인의 장학 속성들을 담고 있는 목록

2. 오퍼레이션 목록

이름	관리자_장학생목록 불러오기
영문명	Adm_Sch_load
설명	관리자가 장학생 목록을 불러온다.
파라미터	
반환값 및 타입	장학생목록(Arraylist) std_sch_list : 정상적으로 목록을 불러온 경우 값이 들어간 std_sch_list를 반환 std_sch_list : 정상적이지 못한 경우 비어있는 std_sch_list를 반환
개요	외부파일로부터 이미 저장되어 있는 관리자나 학생의 정보를 불러오는 함수.
알고리즘	<pre> try{ file filereader, bufferedReader 사용해서 파일을 불러오고 불러온 파일을 줄마다 파싱하고 파싱한 정보를 구조체 담아 구조체를 list에 저장 한다 list를 반환 } catch{ 예외처리} </pre>

이름	관리자_장학생조회
영문명	Adm_Sch_search
설명	관리자가 장학생 목록을 조회한다..
파라미터	x
반환값 및 타입	장학생목록(Arraylist) std_sch_list : 정상적으로 목록을 불러온 경우 값이 들어간 std_sch_list를 반환 std_sch_list : 정상적이지 못한 경우 비어있는 std_sch_list를 반환
개요	저장되어있는 장학생들의 목록을 보여준다.
알고리즘	Adm_Sch_load 호출 return Std_sch_list

이름	학생_장학생조회
영문명	Std_Sch_search
설명	학생이 자기 장학 목록을 조회한다..
파라미터	p_std_id
반환 값 및 타입	장학생목록(Arraylist) sch_list : 정상적으로 목록을 불러온 경우 값이 들어간 std_sch_list를 반환 sch_list : 정상적이지 못한 경우 비어있는 std_sch_list를 반환
개요	저장되어있는 자신의 장학 내역을 보여준다.
알고리즘	Adm_Sch_load 호출 for(장학생 리스트 크기만큼) if(일치하는 학번이 존재 시) 해당하는 정보를 sch_list에 전달 return sch_list else (아닐 경우는 비어있는 sch_list 반환) return sch_list

이름	장학생 등록
영문명	Sch_reg
설명	관리자가 장학생 목록에 장학생을 추가한다.
파라미터	p_std_name, p_std_id, p_std_type, p_std_money
반환 값 및 타입	등록확인(Boolean) "True" : 정상적으로 장학생 등록이 되었을 경우 "False" : 정상적으로 장학생 등록이 되지 않았을 경우
개요	장학생목록에 장학생을 등록한다.
알고리즘	<i>Adm_Sch_load 호출</i> <i>if(존재하지 않는 학번일 경우) {</i> <i>구조체에 학생 정보를 담고</i> <i>구조체를 std_sch_list에 추가해준다.</i> <i>}</i> <i>Sch_add 함수 호출.</i>

이름	장학생 삭제
영문명	Sch_del
설명	관리자가 장학생 목록에 장학생을 삭제한다.
파라미터	p_std_id
반환 값 및 타입	삭제확인(Boolean) "True" : 정상적으로 장학생 삭제가 되었을 경우 "False" : 정상적으로 장학생 삭제가 되지 않았을 경우
개요	장학생목록에 장학생을 삭제한다.
알고리즘	<i>Adm_Sch_load 호출</i> <i>for(장학생 리스트 크기만큼)</i> <i>if(일치하는 장학생 학번이 존재 시){</i> <i>장학생 정보 삭제</i> <i>Sch_add함수 호출</i> <i>}</i>

이름	장학생 수정
영문명	Sch_mod
설명	관리자가 장학생 목록에 장학생을 추가한다.
파라미터	p_std_name, p_std_id, p_std_type, p_std_money
반환 값 및 타입	수정확인(Boolean) "True" : 정상적으로 장학생 수정이 되었을 경우 "False" : 정상적으로 장학생 수정이 되지 않았을 경우
개요	장학생목록에 수정된 장학생 정보를 수정한다.
알고리즘	<i>Adm_Sch_load 호출</i> <i>for(장학생 리스트 크기만큼){</i> <i>if(일치하는 장학생 학번이 존재할 시)</i> <i>장학생 정보를 수정</i> <i>Sch_add함수 호출</i>

이름	장학생 추가
영문명	Sch_add
설명	장학생 목록을 외부파일에 써준다.
파라미터	p_std_sch_list
반환 값 및 타입	x
개요	외부파일에 장학생을 목록을 저장한다.
알고리즘	<i>try{</i> <i>writer를 사용하기 위한 값들 처리하고</i> <i>for(장학생 리스트 크기만큼)</i> <i>장학생 정보, 한줄 띄우기 write</i> <i>}catch{</i> <i>예외처리</i> <i>}</i>

클래스 명	Scheduler
클래스 설명	시간표 관리

1. 속성

속성 명	타입	논리모델 속성명	설명
subject_name	string	과목 이름	과목의 이름을 나타내는 속성
time	string	강의 시간	강의 시간을 나타내는 속성
day	string	요일	요일을 나타내는 속성
std_id	string	학생 학번	학번을 나타내는 속성
sch_map	map	시간표 맵	key값으로 강의 명을 받고 value로 sch_list를 가지는 map
sch_list	list	시간표 리스트	강의시간, 요일이 담겨있는 list

2. 오퍼레이션 목록

이름	시간표 생성
영문명	Scheduler_create
설명	학생의 시간표를 생성할 수 있게 한다.
파라미터	p_std_id
반환 값 및 타입	<p>학생들의 시간표 정보가 담겨있는 결과(map)</p> <ul style="list-style-type: none"> sch_map : 정상적인 경우 학생들의 시간표 정보가 담겨있는 map 반환 sch_map : 비정상 적인 경우 비어있는 map반환
개요	학생 정보로부터 값들을 불러와서 학생의 시간표를 생성한다.
알고리즘	<p>subject_list_load 호출</p> <p>for(subject_list_load 호출로 반환받은 리스트 크기만큼)</p> <p>if(학번 같은 값을 경우)</p> <p>호출 한 값들을 list(강의시간, 요일)와 map(key=과목이름, value= list)을 이용하여 처리 하여 준다.</p> <p>}</p> <p>사후 처리는 반환 된 맵을 사용하여 jsp와 javascript로 web에서 시간표 html를 생성 한다.</p>

이름	시간표 조회
영문명	Schedular_view
설명	학생의 시간표를 조회 할 수 있게 한다.
파라미터	p_std_id
반환 값 및 타입	x
개요	학생 정보로부터 값들을 불러와서 학생의 시간표를 조회한다.
알고리즘	<i>if(학번.html 파일존재시)</i> <i>html 파일을 호출한다.</i>

이름	시간표 삭제
영문명	Schedular_del
설명	학생의 시간표를 삭제할 수 있게 한다.
파라미터	p_std_id
반환 값 및 타입	x
개요	학생 정보로부터 값들을 불러와서 학생의 시간표 html 파일을 삭제한다.
알고리즘	<i>if(학번.html 파일존재시)</i> <i>html 파일을 삭제한다.</i>

클래스 명	Student_information
클래스 설명	학생의 정보들을 관리하는 클래스

1. 속성

속성 명	타입	논리모델 속성명	설명
Std_name	String	학생 이름	학생의 이름을 나타내는 속성
Std_id	String	학생 학번	학생의 학번을 나타내는 속성
Std_major	String	학생 전공	학생의 전공학과를 나타내는 속성
Std_image	String	학생 이미지 주소	학생의 이미지 파일을 불러오는 주소를 담고 있는 속성
Std_phone_num	String	학생 휴대폰 번호	학생의 휴대폰 번호를 나타내는 속성
std_private_num	String	학생 주민등록번호	학생의 주민등록번호를 나타내는 속성
info_list	ArrayList	학생 정보 목록	학생 개인의 정보를 담고 있는 list
std_address	String	학생 집 주소	학생의 집 주소를 나타내는 속성
entrance	String	학생 입학 년도	학생의 입학 년도를 나타내는 속성
Student	Struct(구조체)	학생 구조체	학생의 정보를 담은 구조체

2. 오퍼레이션 목록

이름	학생 목록 불러오기
영문명	Std_infolist_load
설명	학생 목록을 불러온다.
파라미터	
반환값 및 타입	학생 목록(ArrayList) std_info_list : 정상적으로 목록을 불러온 경우 값이 들어간 std_info_list를 반환 std_info_list : 정상적이지 못한 경우 비어있는 std_info_list를 반환
개요	외부파일로부터 이미 저장되어 있는 관리자나 학생의 정보를 불러오는 함수.
알고리즘	<pre> try{ file filereader, bufferedReader 사용해서 파일을 불러오고 불러온 파일을 줄마다 파싱하고 파싱한 정보를 구조체 담아 구조체를 list에 저장 한다 list를 반환 } catch{ 예외처리} </pre>

이름	학번 부여
영문명	std_id_assign
설명	학생의 학번을 부여한다.
파라미터	p_std_private_num, p_std_major, p_entrance
반환값 및 타입	학생 학번(String) "id" : 정상적으로 학번을 부여했을 경우 학번을 반환
개요	학생에게 학번을 부여하기 위한 함수
알고리즘	<pre> try{ 입학년도, std_major, Table Match값, 주민등록번호를 조합하여 id 값 생성 id 반환 } catch{ 예외처리 } </pre>
이름	학적 등록
영문명	std_info_reg
설명	학생의 학적을 등록한다.
파라미터	p_std_name, p_std_private_num, p_std_major, p_std_address, p_std_image, p_std_phone_num, p_entrance
반환값 및 타입	등록 확인(Boolean) "True" : 정상적으로 학적 등록이 되었을 경우 "False" : 정상적으로 학적 등록이 되지 않았을 경우
개요	학생의 학적을 등록하기 위한 함수.
알고리즘	<pre> try{ std_infolist_load 호출 if(주민등록번호가 list에 존재하지 않을 경우를 확인) Student 구조체에 담긴 변수에 맞게 값 지정 std_info_load에 write } catch{ 예외처리} </pre>

이름	학생 학적 수정
영문명	std_info_modi
설명	학생이 학생의 학적을 수정한다.
파라미터	p_std_id, p_std_address, p_std_image, p_std_phone_num
반환값 및 타입	수정 확인(Boolean) "True" : 정상적으로 학적 수정이 되었을 경우 "False" : 정상적으로 학적 수정이 되지 않았을 경우
개요	학생이 학적을 수정하기 위한 함수.
알고리즘	<pre> try{ std_infolist_load 호출 if(학번이 존재할 경우를 확인) Student 구조체에 담긴 변수에 맞게 값 수정 std_info_load에 write } catch{ 예외처리} </pre>

이름	관리자 학적 수정
영문명	Adm_info_modi
설명	관리자가 학생의 학적을 수정한다.
파라미터	p_std_id, p_std_name, p_std_private_num, p_std_major, p_std_address, p_std_image, p_std_phone_num
반환값 및 타입	수정 확인(Boolean) "True" : 정상적으로 학적 수정이 되었을 경우 "False" : 정상적으로 학적 수정이 되지 않았을 경우
개요	관리자가 학적을 수정하기 위한 함수.
알고리즘	<pre> try{ std_infolist_load 호출 if(학번이 존재할 경우를 확인) Student 구조체에 담긴 변수에 맞게 값 수정 std_info_load에 write } catch{ 예외처리} </pre>

이름	학적 삭제
영문명	info_del
설명	관리자가 학생의 학적을 삭제한다.
파라미터	p_std_id
반환값 및 타입	삭제 확인(Boolean) "True" : 정상적으로 학적 삭제가 되었을 경우 "False" : 정상적으로 학적 삭제가 되지 않았을 경우
개요	관리자가 학적을 삭제하기 위한 함수.
알고리즘	<pre> try{ std_infolist_load 호출 if(학번이 존재할 경우를 확인) info_list에서 성적정보를 제거한다. std_info_load에 write } catch{ 예외처리} </pre>
이름	학적 조회
영문명	info_search
설명	학생의 학적을 조회한다.
파라미터	p_std_id
반환값 및 타입	학생정보 목록(Arraylist) info_list : 정상적으로 목록을 불러온 경우 값이 들어간 info_list를 반환 info_list : 정상적이지 못한 경우 비어있는 info_list를 반환
개요	학생의 학적을 조회하기 위한 함수.
알고리즘	<pre> try{ std_info_load 호출 for(list의 크기만큼) if(입력받은 학번과 일치하는 학번이 존재 하는 경우) std_info_list에서 맞는값을 info_list에 전달 return info_list else return info_list(비어있는 리스트임) } catch{ 예외처리} </pre>

이름	학적 추가
영문명	Std_info_add
설명	학적 목록을 외부파일에 써준다.
파라미터	p_std_info_list
반환 값 및 타입	x
개요	외부파일에 학적정보를 담은 목록을 저장한다.
알고리즘	<pre> try{ writer를 사용하기 위한 값들 처리하고 for(학적목록 리스트 크기만큼) 학적 정보, 한줄 띄우기 write }catch{ 예외처리 } </pre>

클래스 명	Grade
클래스 설명	학생의 성적을 관리하는 클래스

1. 속성

속성 명	타입	논리모델 속성명	설명
Std_point	int	학점	학생의 이수학점을 나타내는 속성 ex)3 2
Std_score	String	등급 / 평점	학생의 성적정보를 나타내는 속성
Std_grade	String	성적평가	성적이 평점인지 p/f인지 나타내는 속성
Std_Cur	String	이수 구분	과목의 유형을 나타내는 속성
Std_subject	String	교과목 명	과목의 이름을 나타내는 속성
Std_id	String	학번	학생의 학번을 나타내는 속성
Std_subnum	String	학수 번호	과목의 학수번호를 나타내는 속성
Std_date	String	년도/학기	개설학과와 년도를 나타내는 속성
std_grade_list	Arraylist	학생성적정보목록	학생들의 성적 정보를 담고 있는 list
grade_list	Arraylist	학생성적정보목록	학생 개인의 성적 정보를 담고 있는 list
Student	Struct(구조체)	학생 구조체	학생의 정보를 담은 구조체

2. 오퍼레이션 목록

이름	학생 성적 정보목록 불러오기
영문명	grade_list_load
설명	학생들의 성적 정보를 담긴 목록을 불러온다.
파라미터	
반환값 및 타입	학생 성적 정보목록(Arraylist) std_grade_list : 정상적으로 목록을 불러온 경우 값이 들어간 std_grade_list를 반환 std_grade_list : 정상적이지 못한 경우 비어있는 std_grade_list를 반환
개요	외부파일로부터 이미 저장되어 있는 관리자나 학생의 정보를 불러오는 함수.
알고리즘	<pre> try{ file filereader, bufferedReader 사용해서 파일을 불러오고 불러온 파일을 줄마다 파싱하고 파싱한 정보를 구조체 담아 구조체를 list에 저장 한다 list를 반환 } catch{ 예외처리 } </pre>

이름	학생 성적 등록
영문명	gradelist_reg
설명	학생들의 성적 정보를 담긴 목록을 등록한다.
파라미터	p_std_id, p_std_point, p_std_score, p_std_grade, p_std_subnum, p_std_date, p_std_Cur, p_std_subject
반환값 및 타입	등록 확인(Boolean) "True" : 정상적으로 성적 등록이 되었을 경우 "False" : 정상적으로 성적 등록이 되지 않았을 경우
개요	외부파일에서 불러온 목록에 새로운 학생들의 성적 정보를 등록하는 함수
알고리즘	Std_gradelist_load 호출 for(std_grade_list 크기만큼) if(일치하는 학번 존재 시) if(일치하는 학수번호 존재 하지 않을 시) 구조체에 값을 맞게 넣고 Std_grade_list에 추가 td_grade_add 호출 else if(일치하는 학번이 존재하지 않을 시) 조건이 맞지 않으므로 마무리

이름	관리자 성적정보 조회
영문명	grade_search
설명	학생들의 성적 정보를 담긴 목록을 조회한다.
파라미터	p_std_id, p_std_subnum
반환값 및 타입	학생 성적 정보목록(Arraylist) grade_list : 정상적으로 목록을 불러온 경우 값이 들어간 std_grade_list를 반환 grade_list : 정상적이지 못한 경우 비어있는 std_grade_list를 반환
개요	학생의 성적을 조회하기 위한 함수.
알고리즘	Std_gradelist_load 호출 for(Std_grade_list의 크기만큼) if(일치하는 학번을 찾는다) Std_grade_list에서 학번이 일치하는 정보를 grade_list에 전달 return grade_list else return grade_list(empty list)

이름	학생 성적정보 조회
영문명	std_grade_search
설명	학생들의 성적 정보를 담긴 목록을 조회한다.
파라미터	p_std_id, p_std_date
반환값 및 타입	학생 성적 정보목록(Arraylist) grade_list : 정상적으로 목록을 불러온 경우 값이 들어간 std_grade_list를 반환 grade_list : 정상적이지 못한 경우 비어있는 std_grade_list를 반환
개요	학생의 성적을 조회하기 위한 함수.
알고리즘	<i>Std_gradelist_load 호출 for(Std_grade_list의 크기만큼) if(일치하는 학번을 찾는다.&& 일치하는 년도/학기를 찾는다.) Std_grade_list에서 학번이 일치하는 정보를 grade_list에 전달 return grade_list else return grade_list(empty list)</i>

이름	학생 성적 정보 삭제
영문명	Std_gradelist_del
설명	관리자가 학생들의 성적 정보를 담긴 목록을 삭제한다.
파라미터	p_std_subnum, p_std_id
반환값 및 타입	삭제 확인(Boolean) "True" : 정상적으로 성적 삭제가 되었을 경우 "False" : 정상적으로 성적 삭제가 되지 않았을 경우
개요	관리자가 학생의 성적을 삭제하기 위한 함수.
알고리즘	<i>Std_gradelist_load 호출 for(Std_grade_list의 크기만큼) if(일치하는 학번을 찾는다 && 일치하는 학수번호를 찾는다) Std_grade_list에서 성적정보를 제거한다. Std_grade_add 호출</i>

이름	학생 성적정보 수정
영문명	Std_grade_mod
설명	관리자가 학생들의 성적 정보를 담긴 목록을 수정한다.
파라미터	p_std_point, p_std_score, p_std_grade, p_std_Cur, p_std_subject, p_std_subnum, p_std_date, p_std_id
반환값 및 타입	수정 확인(Boolean) "True" : 정상적으로 성적 수정이 되었을 경우 "False" : 정상적으로 성적 수정이 되지 않았을 경우
개요	관리자가 학생의 성적을 수정하기 위한 함수
알고리즘	<i>Std_gradelist_load 호출</i> <i>for(Std_grade_list의 크기만큼)</i> <i>if(일치하는 학번을 찾는다. && 일치하는 학수번호를 찾는다)</i> <i>Std_grade_list에서 성적정보를 수정한다.</i> <i>Std_grade_add 호출</i>

이름	성적정보 추가
영문명	std_grade_add
설명	성적정보 목록을 외부파일에 써준다.
파라미터	p_std_grade_list
반환 값 및 타입	x
개요	외부파일에 장학생을 목록을 저장한다.
알고리즘	<i>try{</i> <i>writer를 사용하기 위한 값들 처리하고</i> <i>for(성적정보 리스트 크기만큼)</i> <i>성적 정보, 한줄 띄우기 write</i> <i>}catch{</i> <i>예외처리</i> <i>}</i>

클래스 명	Subject
클래스 설명	교과목 정보를 관리하는 클래스

1. 속성

속성 명	타입	논리모델 속성명	설명
Point	int	학점	교과목의 학점을 나타내는 속성
Sub_name	String	교과목 명	교과목의 이름을 나타내는 속성
Major	String	개설학과	교과목을 개설한 학과를 나타내는 속성
Date	String	개설학기	교과목이 개설된 학기를 나타내는 속성
Prof_name	String	담당교수	교과목을 담당한 교수의 이름을 나타내는 속성
Std_size	String	수강정원	해당 교과목의 수강정원을 나타내는 속성
Sub_num	String	학수번호	교과목의 학수번호를 나타내는 속성
Subject_list	ArrayList	교과목 정보 목록	교과목들의 정보를 담고 있는 목록
Std_subject_list	ArrayList	교과목 정보 목록	교과목의 정보를 담고 있는 목록

2. 오퍼레이션 목록

이름	교과목 정보목록 불러오기
영문명	subject_list_load
설명	교과목 정보가 담긴 목록을 불러온다.
파라미터	
반환값 및 타입	교과목 정보목록(ArrayList) Subject_list_load : 정상적으로 목록을 불러온 경우 값이 들어간 Subject_list_load를 반환 Subject_list_load : 정상적이지 못한 경우 비어있는 Subject_list_load를 반환
개요	외부파일로부터 이미 저장되어 있는 교과목 정보를 불러오는 함수.
알고리즘	<pre> try{ file filereader, bufferedReader 사용해서 파일을 불러오고 불러온 파일을 줄마다 파싱하고 파싱한 정보를 구조체 담아 구조체를 list에 저장 한다 list를 반환 } catch{ 예외처리 } </pre>

이름	교과목 정보 등록
영문명	subject_list_reg
설명	교과목 정보가 담긴 목록을 등록한다.
파라미터	p_Point, p_Sub_name, p_Major, p_Date, p_Prof_name, p_Std_size, p_Sub_num
반환값 및 타입	등록 확인(Boolean) "True" : 정상적으로 교과목 등록이 되었을 경우 "False" : 정상적으로 교과목 등록이 되지 않았을 경우
개요	외부파일에서 불러온 목록에 새로운 교과목 정보를 등록하는 함수
알고리즘	<i>Subject_list_load 호출 for(Subject_list 크기만큼) if(일치하는 학수번호 존재하지 않을 시) 구조체에 값을 맞게 넣고 Sujcet_list에 추가 Subject_list_add 호출</i>

이름	교과목 정보 삭제
영문명	subject_list_del
설명	교과목 정보가 담긴 목록을 삭제한다.
파라미터	p_Sub_num
반환값 및 타입	삭제 확인(Boolean) "True" : 정상적으로 교과목 삭제가 되었을 경우 "False" : 정상적으로 교과목 삭제가 되지 않았을 경우
개요	교과목을 삭제하기 위한 함수.
알고리즘	<i>Subject_list_load 호출 for(Subject_list의 크기만큼) if(일치하는 학수번호를 찾는다) Subject_list에서 교과목정보를 제거한다. Subject_list_add 호출</i>

이름	교과목 정보 수정
영문명	subject_list_modi
설명	교과목 정보가 담긴 목록을 수정한다.
파라미터	p_Point, p_Sub_name, p_Major, p_Date, p_Prof_name, p_Std_size, p_Sub_num
반환값 및 타입	수정 확인(Boolean) "True" : 정상적으로 교과목 수정이 되었을 경우 "False" : 정상적으로 교과목 수정이 되지 않았을 경우
개요	교과목 정보를 수정하기 위한 함수.
알고리즘	<i>Subject_list_load 호출</i> <i>for(Subject_list의 크기만큼)</i> <i>if(일치하는 학수번호를 찾는다)</i> <i>Subject_list에서 교과목정보를 수정한다.</i> <i>Subject_list_add 호출</i>

이름	관리자 교과목 정보 조회
영문명	Adm_subject_search
설명	교과목 정보를 담긴 목록을 조회한다.
파라미터	
반환값 및 타입	교과목 정보목록(Arraylist) Subject_list : 정상적으로 목록을 불러온 경우 값이 들어간 Subject_list를 반환
개요	교과목 정보를 조회하기 위한 함수.
알고리즘	<i>Subject_list_load 호출</i> <i>return Subject_list</i>

이름	학생 교과목 정보 조회
영문명	Std_subject_search
설명	교과목 정보가 담긴 목록을 조회한다.
파라미터	p_Sub_num
반환값 및 타입	교과목 정보목록(Arraylist) Std_subject_list : 정상적으로 목록을 불러온 경우 값이 들어간 Std_subject_list를 반환 Std_subject_list : 정상적이지 못한 경우 비어있는 Std_subject_list를 반환
개요	교과목을 조회하기 위한 함수.
알고리즘	<i>Subject_list_load 호출</i> <i>for(Subject_list의 크기만큼)</i> <i>if(일치하는 학수번호를 찾는다.)</i> <i>Subejct_list에서 학수번호가 일치하는 정보를 Std_subejct_list에 전달</i> <i>return Std_subject_list</i> <i>else</i> <i>return Std_subject_list(empty list)</i>

이름	교과목 정보 추가
영문명	Subejct_add
설명	교과목 정보 목록을 외부파일에 써준다.
파라미터	p_Subject_list
반환 값 및 타입	x
개요	외부파일에 교과목 목록을 저장한다.
알고리즘	<i>try{</i> <i>writer를 사용하기 위한 값들 처리하고</i> <i>for(교과목정보 리스트 크기만큼)</i> <i>교과목 정보, 한줄 띄우기 write</i> <i>}catch{</i> <i>예외처리</i> <i>}</i>

클래스 명	Subject_plan
클래스 설명	강의계획서 정보를 관리하는 클래스

1. 속성

속성 명	타입	논리모델 속성명	설명
sub_name	String	교과목 명	교과목 이름을 나타내는 속성
major	String	개설학과	개설학과를 나타내는 속성
date	String	설강학기	설강학기를 나타내는 속성
sub_num	String	학수번호	학수번호를 나타내는 속성
p_h_e	String	학점시수 및 평가	(point-hour-evaluation) 학점시수 및 평가를 나타내는 속성
prof_name	String	담당교수	담당교수를 나타내는 속성
prof_major	String	교수소속	교수소속을 나타내는 속성
major_tel	String	학과전화	학과전화 번호를 나타내는 속성
prof_tel	String	담당교수전화	담당교수전화 번호를 나타내는 속성
prof_email	String	담당교수E-MAIL	담당교수 E-mail을 나타내는 속성
anticipate	String	기대역량	강의의 기대역량을 나타내는 속성
class_info	String	수업개요	수업의 개요를 나타내는 속성
pre_study	String	선수학습내용	선수 학습내용을 나타내는 속성
method	String	수업방법(%)	수업방법을 나타내는 속성
evaluation	String	평가방법(%)	평가방법을 나타내는 속성
object	String	교과목표	교과목표를 나타내는 속성
operate	String	수업운영	수업운영 방법을 나타내는 속성
std_eva	String	학습 및 평가활동	학습 및 평가활동을 나타내는 속성
rule	String	수업 규정	수업 규정을 나타내는 속성
reference	String	교재 및 참고문헌	교재 및 참고문헌을 나타내는 속성
week_sche	String	주차별 강의 계획	주차별 강의 계획을 나타내는 속성
ps	String	참고사항	참고사항을 나타내는 속성
disabil_info	String	장애학습 학습편의 제공안내	장애학습 편의 제공 안내에 대한 정보를 나타내는 속성
subject_plan_map	hash map	강의계획서 정보 목록	강의 계획서들의 정보를 가지고 있는 목록 (key : 학수번호, value : arraylist)
subject_plan_list	arraylist	강의계획서 정보 목록	강의 계획서의(학수번호를 제외한) 정보를 가지고 있는 목록

2. 오퍼레이션 목록

이름	강의계획서 정보목록 불러오기
영문명	Subject_plan_list_load
설명	강의계획서 정보가 담긴 목록을 불러온다.
파라미터	
반환값 및 타입	강의계획서 정보 목록(Hash map) Subject_planlist_load : 정상적으로 목록을 불러온 경우 값이 들어간 Subject_planlist_load를 반환 Subject_planlist_load : 정상적이지 못한 경우 비어있는 Subject_planlist_load를 반환
개요	외부파일로부터 이미 저장되어 있는 교과목 정보를 불러오는 함수.
알고리즘	<pre> try{ file filereader, bufferedReader 사용해서 파일을 불러오고 불러온 파일을 줄마다 파싱하고 hash map에 저장 한다 hash map을 반환 } catch{ 예외처리 } </pre>

이름	강의계획서 정보 삭제
영문명	subject_plan_list_del
설명	관리자가 강의계획서를 삭제한다.
파라미터	p_sub_num
반환값 및 타입	삭제 확인(Boolean) "True" : 정상적으로 강의계획서 삭제가 되었을 경우 "False" : 정상적으로 강의계획서 삭제가 되지 않았을 경우
개요	관리자가 강의계획서를 삭제하기 위한 함수
알고리즘	<p>Subject_planlist_load 호출</p> <p>for(subject_plan_map 크기만큼)</p> <p>if(일치하는 학수번호가 있는 경우)</p> <p>subject_plan_map에서 정보를 삭제한다.</p> <p>subject_planlist_add 호출</p>

이름	관리자 강의계획서 정보 등록
영문명	Adm_subject_plan_list_reg
설명	강의계획서를 등록한다.
파라미터	p_sub_name, p_major, p_date, p_sub_num, p_p_h_e, p_prof_name, p_prof_major, p_major_tel, p_prof_tel, p_prof_email, p_anticipate, p_class_info, p_pre_study, p_method, p_evaluation, p_object, p_operate, p_std_eva, p_rule, p_reference, p_week_sche, p_ps, p_disabil_info
반환값 및 타입	등록 확인(Boolean) "True" : 정상적으로 강의계획서 등록이 되었을 경우 "False" : 정상적으로 강의계획서 등록이 되지 않았을 경우
개요	강의 계획서를 등록하기 위한 함수
알고리즘	<i>Subject_planlist_load 호출</i> <i>Subject_list_load 호출</i> try{ for(subject_plan_map 크기 만큼){ if(subject_plan_map에 학수번호가 존재하는지 확인) for(subject_list 크기 만큼) if(subject_list에 학수번호가 일치하는 과목이 있을 경우) 입력받은 값을 subject_plan_list에 담는다 subject_plan_list와 입력받은 학수번호 값을 subject_plan_map에 추가한다. subject_planlist_add 호출 } } catch(예외처리)

이름	관리자 강의계획서 정보 수정
영문명	Adm_subject_plan_list_mod
설명	강의계획서를 수정한다.
파라미터	p_sub_name, p_major, p_date, p_sub_num, p_p_h_e, p_prof_name, p_prof_major, p_major_tel, p_prof_tel, p_prof_email, p_anticipate, p_class_info, p_pre_study, p_method, p_evaluation, p_object, p_operate, p_std_eva, p_rule, p_reference, p_week_sche, p_ps, p_disabil_info
반환 값 및 타입	수정 확인(Boolean) "True" : 정상적으로 강의계획서 수정이 되었을 경우 "False" : 정상적으로 강의계획서 수정이 되지 않았을 경우
개요	강의 계획서를 수정하기 위한 함수
알고리즘	Subject_planlist_load 호출 try{ for(subject_plan_map 크기 만큼){ if(subject_plan_map에 학수번호가 존재하는지 확인) 입력받은 값을 subject_plan_list에 담는다 subject_plan_list를 subject_plan_map에 수정한다. subject_planlist_add 호출 }} catch{ 예외처리}

이름	강의계획서 조회
영문명	subject_plan_search
설명	강의계획서를 조회한다.
파라미터	p_sub_num
반환값 및 타입	강의계획서 반환(Arraylist) subject_plan_list : 정상적으로 값을 불러온 경우 subject_plan_list를 반환 subject_plan_list : 정상적이지 못한 경우 비어있는 subject_plan_list를 반환
개요	강의계획서를 조회하기 위한 함수.
알고리즘	Subject_plan_list_load 호출 for(Subject_plan_map의 크기만큼) if(일치하는 학수번호를 찾는다.) return subject_plan_map의 key값의 value값 else return subject_plan_map의 key값의 value값(empty list)

이름	강의계획서 정보 추가
영문명	Subejct_planlist_add
설명	강의계획서 정보 목록을 외부파일에 써준다.
파라미터	p_Subject_plan_map
반환 값 및 타입	x
개요	외부파일에 교과목 목록을 저장한다.
알고리즘	<i>try{ writer를 사용하기 위한 값들 처리하고 for(subject_plan_map 크기만큼) 강의계획서 정보, 한줄 띄우기 write }catch{ 예외처리 }</i>

클래스 명	Enrolment
클래스 설명	수강신청을 관리하는 클래스

1. 속성

속성 명	타입	논리모델 속성명	설명
enrol_std	Arraylist	현재 신청인원 목록	현재 신청인원들의 정보를 담고 있는 목록
Std_id	String	학번	학생의 학번을 나타내는 속성
enrol_count	int	현재 신청인원	현재 신청인원들을 나타내는 속성
subject_name	String	교과목 명	교과목의 이름을 나타내는 속성
date	String	강의 시간	강의 시간을 나타내는 속성
sub_lect_room	String	강의실	강의실을 나타내는 속성
sub_prof	String	담당교수	교과목을 담당한 교수의 이름을 나타내는 속성
max_num	int	수강정원	해당 교과목의 수강정원을 나타내는 속성
sub_num	String	학수번호	교과목의 학수번호를 나타내는 속성
Enrolment_list	Arraylist	신청 정보 목록	수강신청 정보를 담고 있는 목록
Enrolment_map	hash map	신청 정보 맵	수강 신청 정보를 담고 있는 맵

2. 오퍼레이션 목록

이름	수강신청 정보목록 불러오기
영문명	Enrolment_list_load
설명	교과목 정보가 담긴 목록을 불러온다.
파라미터	
반환값 및 타입	수강신청 정보(hash map) Enrolment_list_load :정상적으로 목록을 불러온 경우 값이 들어간 Enrolment_list_load를 반환 Enrolment_list_load : 정상적이지 못한 경우 비어있는 Enrolment_list_load를 반환
개요	외부파일로부터 이미 저장되어 있는 수강신청 정보를 불러오는 함수.
알고리즘	<pre> try{ file filereader, bufferedReader 사용해서 파일을 불러오고 불러온 파일을 줄마다 파싱하고 파싱한 정보를 hash map에 저장 한다 hash map을 반환 } catch{ 예외처리 } </pre>

이름	수강신청
영문명	Enrol
설명	수강 신청한다.
파라미터	p_sub_num, p_std_id
반환값 및 타입	신청 확인(Boolean) "True" : 정상적으로 수강 신청이 되었을 경우 "False" : 정상적으로 수강 신청이 되지 않았을 경우
개요	수강 신청을 하기 위한 함수
알고리즘	<i>enrolment_list_load 호출</i> <i>for(enrolment_list_map 크기만큼)</i> <i>if(학수번호가 map에 존재하는 경우 && 현재수강신청인원의 크기가 수강정원보다 작을 경우 && 현재신청인원에 학번이 존재하지 않을 경우)</i> <i>학수번호에 맞는 현재신청인원에 학번을 추가</i> <i>enrolment_list_add 호출</i>

이름	수강 취소
영문명	Enrol_cancel
설명	수강 취소한다.
파라미터	p_sub_num, p_std_id
반환값 및 타입	취소 확인(Boolean) "True" : 정상적으로 수강 취소가 되었을 경우 "False" : 정상적으로 수강 취소가 되지 않았을 경우
개요	수강 신청을 취소하기 위한 함수
알고리즘	<i>enrolment_list_load 호출</i> <i>for(enrolment_list_map 크기만큼)</i> <i>if(학수번호가 map에 존재하는 경우 && 현재신청인원에 학번이 존재하는 경우)</i> <i>학수번호에 맞는 현재신청인원에 학번을 삭제</i> <i>enrolment_list_add 호출</i>

이름	학생_수강 과목 조회
영문명	Std_Enrol_view
설명	학생이 수강 신청한 과목을 조회한다.
파라미터	p_sub_num, p_std_id, p_std_cur, p_major
반환값 및 타입	수강과목목록(hash_map) enrolment_map :정상적으로 값이 존재하는 경우 그에 해당하는 enrolment_map 반환 enrolment_map :정상적이지 못한 경우 비어있는 enrolment_map반환
개요	수강 신청을 조회하기 위한 함수
알고리즘	enrolment_list_load 호출 subject_list_load 호출 html 파일 호출 html 조회 항목을 선택 선택 받은 값에 맞는 형태로 값을 입력 for(enrolment_list_map 크기만큼) if(선택 받은 값과 일치하는 항목 존재 시) 그에 대한 enrolment_map 값 반환 else 비어있는 enrolment_map 반환

이름	수강과목 삭제
영문명	Adm_Enrol_del
설명	수강 신청할 수 있는 과목을 삭제한다.
파라미터	p_sub_num
반환 값 및 타입	x
개요	수강 신청 과목을 등록하기 삭제 위한 함수
알고리즘	enrolment_list_load 호출 for(enrolment_list_map 크기만큼) if(학수번호 존재할 경우) enrolmet_map에서 정보를 삭제한다. enrolment_list_add 호출

이름	수강과목 등록
영문명	Std_Enrol_reg
설명	수강 신청할 수 있는 과목을 등록한다.
파라미터	p_sub_num, p_sub_name, p_sub_prof, p_date, p_max_num, p_enrol_count, p_lect_room
반환 값 및 타입	x
개요	수강 신청 과목을 등록하기 위한 함수
알고리즘	<i>enrolment_list_load 호출</i> <i>for(enrolment_list_map 크기만큼)</i> <i>if(학수번호 존재하지 않을 시 && 강의실이 비어있는지 && 강의시간이 비어있는지)</i> <i>enrolment_map에 정보를 추가한다.</i> <i>enrolment_list_add 호출</i>

이름	관리자_수강 과목 조회
영문명	Adm_Enrol_view
설명	관리자가 수강 신청한 과목을 조회한다.
파라미터	
반환값 및 타입	수강과목목록(hash_map) enrolment_map :정상적으로 값이 존재하는 경우 그에 해당하는 enrolment_map 반환
개요	수강 신청을 조회하기 위한 함수
알고리즘	<i>enrolment_list_load 호출</i> <i>enrolment_map 값 반환</i>

이름	수강신청 정보 추가
영문명	enrolment_list_add
설명	수강신청 정보 목록을 외부파일에 써준다.
파라미터	p_enrolment_map
반환 값 및 타입	x
개요	외부파일에 교과목 목록을 저장한다.
알고리즘	<i>try{</i> <i>writer를 사용하기 위한 값들 처리하고</i> <i>for(enrolment_map 크기만큼)</i> <i>수강신청 정보, 한줄 띄우기 write</i>

	<code>}catch{</code> <code>예외처리}</code>
--	--