전국민독서량증진과도서관방문율을향상시키는

**도서관통합관리시스템( Lifrary )**

|  |  |
| --- | --- |
| Hosting | [http://tkdguq93.cafe24.com/](http://shohye.cafe24.com/)  (Chrome 최적화) |
| Git hub | <https://github.com/tkdguq9369/Lifrary> |

|  |  |
| --- | --- |
| 목차 | |
| 1.[프로젝트 기본 정보](#프로젝트기본정보) | |
| 2. 요구사항 정의서 | |
|  | [2.1 기능적](#구현기능) 요구사항 |
| 3. [업무분담 (담당업무)](#업무분담담당업무) | |
| 4. 설계 사양서 | |
|  | 4.1 [ERD 논리모드](#ERD논리모드) |
|  | 4.2 [ERD 물리모드](#ERD물리모드) |
|  | 4.3 [테이블 구조](#테이블구조) |
|  | 4.4[네이밍 규칙](#네이밍규칙) |
|  | 4.5[디렉토리 구조](#디렉토리구조) |
|  | 4.6[패키지 구조](#패키지구조) |
| 5. [개발 환경](#개발환경) | |
| 6. 어플리케이션 | |
|  | 6.1 [어플리케이션 프로시저](#어플리케이션프로시져) |
|  | 6.2 [서버 및 기타 설정](#서버및기타설정) |
|  | 6.3 [주요 기능별 정리](#주요기능별정리) |
|  | 6.3.1 도서관 사이트 |
|  | 6.3.2 관리자 사이트 |
| 7. [프로젝트 후기](#프로젝트후기) | |

**1. 프로젝트 기본 정보**

|  |  |  |
| --- | --- | --- |
| 프로젝트명 | 전국민독서량증진과도서관방문율을향상시키는전국도서관통합관리시스템 | |
| 개발기간 | **■2019-10-24~ 2020-01-13** | |
| **2019-10-24 ~ 2019-11-01** | ⚫프로젝트 주제 선정 및 사전조사 |
| **2019-11-02 ~ 2019-12-01** | ⚫프로젝트 설계(기능정의, 테이블 구상)  ⚫설계자료 취합, 수정 및 보완  ⚫ERD 작업 및 메서드, 패키지 정리 |
| **2019-12-02 ~ 2019-12-03** | ⚫서버 설정  ⚫개발환경 설정  ⚫Hosting 설정 |
| **2019-12-03 ~ 2020-01-09** | ⚫기능 구현(개인 분담 프로그래밍) |
| **2019-12-30 ~ 2020-01-09** | ⚫테스트, 오류 수정 및 보완 |
| **2020-01-10 ~ 2020-01-13** | ⚫결과 보고서 작성 |
| 개발목표 | 1. 개인 독서 공간 제공과 목표량 및 포인트 제도를 도입하여 지속적인 독서량 증진과 도서관 방문을 늘리게 한다.  2. 도서관리 단순화와 일괄된 도서관페이지 관리 시스템으로 효율적이고 통일된 관리를 목표로 한다.  3. 전국 도서관간의 데이터를 공유하여 다양한 통계와 자료를 제공할 수 있다. | |
| 기대효과 | 1. 나만의 독서공간을 마련하여 독서 상황을 기록할 수 있으며, 포인트 제도를 도입하여  도서관사이트를 지속적으로 이용하게 할 수 있다.  2. 도서정보의 반입하고 도서관에 따른 청구기호 생성 등 도서관리 사용법이 손쉬우므로 빠른 작업이 가능하고, 사서수가 부족한 경우에도 불편함 없이 사용할 수 있다.  3. 다양한 문화행사 및 강좌를 제공 및 편리한 공공시설 예약시스템으로 인한 사용자 편의성 증가로 이용자들이 증가하는것을 기대할 수 있다. | |
| 개발방식 |  | |

**2.1 기능적 요구사항**

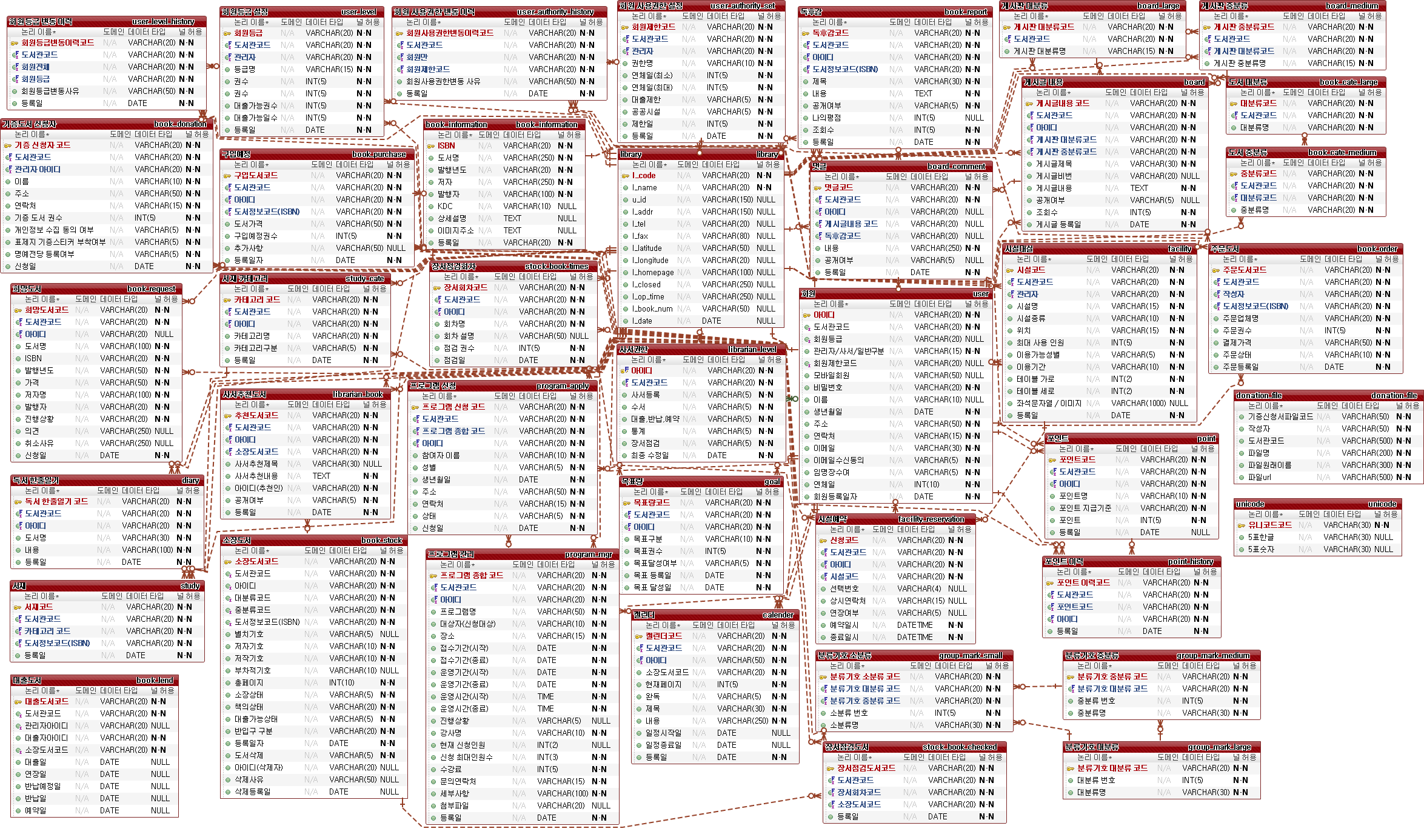
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **기능적 요구사항** | **관리자** | **사서** | **회원** |
| 1 | 로그인 | | | |
| 1.1 | 로그인 | 로그인 | 로그인 |  |
| 1.2 | 로그아웃 | 로그아웃 | 로그아웃 |  |
| 1.3 | 아이디 찾기 | 아이디 찾기 | 아이디 찾기 | 아이디 찾기 |
| 1.4 | 비밀번호 찾기 | 비밀번호 찾기 | 비밀번호 찾기 | 비밀번호 찾기 |
| 2 | 이용자 관리 | | | |
| 2.1 | 아이디 중복검사 | | | |
| 2.1.1 | 아이디 중복검사 | 아이디 중복검사 | 아이디 중복검사 | 아이디 중복검사 |
| 2.2 | 회원 관리 | | | |
| 2.2.1 | 회원 등록 | 회원 등록 | 회원 등록 | 회원 등록 |
| 2.2.2 | 회원 검색(리스트) | 회원 검색(리스트) | 회원 검색(리스트) |  |
| 2.2.3 | 회원 상세정보 | 회원 상세정보 | 회원 상세정보 |  |
| 2.2.4 | 회원 정보 수정 | 회원 정보 수정 | 회원 정보 수정 | 회원 정보 수정 |
| 2.2.5 | 회원 탈퇴 | 회원탈퇴 | 회원탈퇴 | 회원탈퇴 |
| 2.3 | 사서 관리 | | | |
| 2.3.1 | 사서 등록 | 사서 등록 | 사서 등록 |  |
| 2.3.2 | 사서 권한 부여 | 사서 권한 부여 |  |  |
| 2.3.3 | 사서 검색(리스트) | 사서 검색(리스트) |  |  |
| 2.3.4 | 사서 상세정보 | 사서 상세정보 |  |  |
| 2.3.5 | 사서 정보 수정 | 사서 정보 수정 | 사서 정보 수정 |  |
| 2.3.6 | 사서 삭제 | 사서 삭제 | 사서 삭제 |  |
| 2.4 | 목표량 관리 | | | |
| 2.4.1 | 목표량 등록 | 목표량 등록 | 목표량 등록 | 목표량 등록 |
| 2.4.2 | 목표량 삭제 | 목표량 삭제 | 목표량 삭제 | 목표량 삭제 |
| 2.5 | 포인트 관리 | | | |
| 2.5.1 | 포인트 등록 | 포인트 등록 | 포인트 등록 |  |
| 2.5.2 | 포인트 수정 | 포인트 수정 | 포인트 수정 |  |
| 2.5.3 | 포인트 삭제 | 포인트 삭제 | 포인트 삭제 |  |
| 2.5.4 | 포인트 검색(리스트) | 포인트 검색(리스트) | 포인트 검색(리스트) |  |
| 2.5.5 | 포인트 이력 등록 | 포인트 이력 등록 | 포인트 이력 등록 | 포인트 이력 등록 |
| 2.5.6 | 포인트 이력 삭제 | 포인트 이력 삭제 | 포인트 이력 삭제 |  |
| 2.5.7 | 포인트 사용 내역검색(리스트) | 포인트 사용 내역 검색(리스트) | 포인트 사용 내역 검색(리스트) | 포인트 사용 내역 검색(리스트) |
| 3 | 도서 관리 | | | |
| 3.1 | 희망 도서 신청 관리 | | | |
| 3.1.1 | 희망 도서 신청 등록 | 희망 도서 신청 등록 | 희망 도서 신청 등록 | 희망 도서 신청 등록 |
| 3.1.2 | 희망 도서 신청 삭제 | 희망 도서 신청 삭제 | 희망 도서 신청 삭제 | 희망 도서 신청 삭제 |
| 3.1.3 | 희망 도서 신청 수정 | 희망 도서 신청 수정 | 희망 도서 신청 수정 |  |
| 3.1.4 | 희망 도서 신청 리스트 | 희망 도서 신청 리스트 | 희망 도서 신청 리스트 | 희망 도서 신청 리스트 |
| 3.1.5 | 희망 도서 신청 검색(리스트) | 희망 도서 신청 검색(리스트) | 희망 도서 신청 검색(리스트) |  |
| 3.1.6 | 희망 도서 구입대상 등록 | 희망 도서 구입대상 등록 | 희망 도서 구입대상 등록 |  |
| 3.2 | 기증도서 관리 | | | |
| 3.2.1 | 기증 도서 신청서 내용 등록(신청) | 기증 도서 신청서 내용 등록(신청) | 기증 도서 신청서 내용 등록(신청) | 기증 도서 신청서 내용 등록(신청) |
| 3.2.2 | 기증 도서 신청서 내용 수정 | 기증 도서 신청서 내용 수정 | 기증 도서 신청서 내용 수정 | 기증 도서 신청서 내용 수정 |
| 3.2.3 | 기증 도서 신청서 내용 삭제 | 기증 도서 신청서 내용 삭제 | 기증 도서 신청서 내용 삭제 | 기증 도서 신청서 내용 삭제 |
| 3.3 | 구입 도서 관리 | | | |
| 3.3.1 | 구입 도서 등록 | 구입 도서 등록 | 구입 도서 등록 |  |
| 3.3.2 | 구입 도서 수정 | 구입 도서 수정 | 구입 도서 수정 |  |
| 3.3.3 | 구입 도서 삭제 | 구입 도서 삭제 | 구입 도서 삭제 |  |
| 3.3.4 | 구입 도서 검색(리스트) | 구입 도서 검색(리스트) | 구입 도서 검색(리스트) |  |
| 3.4 | 주문 도서 관리 | | | |
| 3.4.1 | 주문 도서 등록 | 주문 도서 등록 | 주문 도서 등록 |  |
| 3.4.2 | 주문 도서 수정 | 주문 도서 수정 | 주문 도서 수정 |  |
| 3.4.3 | 주문 도서 삭제 | 주문 도서 삭제 | 주문 도서 삭제 |  |
| 3.4.4 | 주문 도서 검색(리스트) | 주문 도서 검색(리스트) | 주문 도서 검색(리스트) |  |
| 3.5 | 소장 도서 관리 | | | |
| 3.5.1 | 소장 도서 등록 | 소장 도서 등록 | 소장 도서 등록 |  |
| 3.5.2 | 소장 도서 수정 | 소장 도서 수정 | 소장 도서 수정 |  |
| 3.5.3 | 소장 도서 삭제(리스트) | 소장 도서 삭제(리스트) | 소장 도서 삭제(리스트) |  |
| 3.5.4 | 소장 도서 삭제(DB) | 소장 도서 삭제(DB) | 소장 도서 삭제 |  |
| 3.5.5 | 소장 도서 검색(리스트) | 소장 도서 검색(리스트) | 소장 도서 검색(리스트) | 소장 도서 검색(리스트) |
| 3.5.6 | 도서 상세 검색(리스트) | 도서 상세 검색(리스트) | 도서 상세 검색(리스트) | 도서 상세 검색(리스트) |
| 3.5.7 | 도서 상세 보기 | 도서 상세 보기 | 도서 상세 보기 | 도서 상세 보기 |
| 3.5.8 | 도서 미리 보기 | 도서 미리 보기 | 도서 미리 보기 | 도서 미리 보기 |
| 3.5.9 | 장서 점검 회차 등록 | 장서점검 회차 등록 | 장서점검 회차 등록 |  |
| 3.5.10 | 장서 점검 회차 수정 | 장서 점검 회차 수정 | 장서 점검 회차 수정 |  |
| 3.5.11 | 장서 점검 회차 삭제 | 장서 점검 회차 삭제 | 장서 점검 회차 삭제 |  |
| 3.5.12 | 장서 점검 회차 리스트 | 장서 점검 회차 리스트 | 장서 점검 회차 리스트 |  |
| 3.5.13 | 장서 점검 도서 등록 | 장서 점검 도서 등록 | 장서 점검 도서 등록 |  |
| 3.5.14 | 장서 점검 도서 리스트 | 장서 점검 도서 리스트 | 장서 점검 도서 리스트 |  |
| 3.5.15 | 장서 점검 도서 삭제 | 장서 점검 도서 삭제 | 장서 점검 도서 삭제 |  |
| 3.5.16 | 장서 점검 완료 | 장서 점검 완료 | 장서 점검 완료 |  |
| 3.5.17 | 장서 점검 누락 도서 상태 변경 | 장서 점검 누락 도서 상태 변경 | 장서 점검 누락 도서 상태 변경 |  |
| 3.6 | 대출 관리 | | | |
| 3.6.1 | 도서 대출 등록 | 도서 대출 등록 | 도서 대출 등록 |  |
| 3.6.2 | 대출 도서 검색(리스트) | 대출 도서 검색(리스트) | 대출 도서 검색(리스트) | 대출 도서 검색(리스트) |
| 3.6.3 | 도서 연장 신청 | 도서 연장 신청 | 도서 연장 신청 | 도서 연장 신청 |
| 3.7 | 예약 관리 | | | |
| 3.7.1 | 도서 예약 등록 | 도서 예약 등록 | 도서 예약 등록 | 도서 예약 등록 |
| 3.7.2 | 도서 예약 취소 | 도서 예약 취소 | 도서 예약 취소 | 도서 예약 취소 |
| 3.7.3 | 예약 도서 검색(리스트) | 예약 도서 검색(리스트) | 예약 도서 검색(리스트) | 예약 도서 검색(리스트) |
| 3.8 | 반납 도서 | | | |
| 3.8.1 | 도서 반납 등록 | 도서 반납 등록 | 도서 반납 등록 |  |
| 3.8.2 | 반납 도서 검색(리스트) | 반납 도서 검색(리스트) | 반납 도서 검색(리스트) | 반납 도서 검색(리스트) |
| 3.9 | 서재관리 | | | |
| 3.9.1 | 서재 카테고리 등록 | 서재 카테고리 등록 | 서재 카테고리 등록 | 서재 카테고리 등록 |
| 3.9.2 | 서재 카테고리 삭제 | 서재 카테고리 삭제 | 서재 카테고리 삭제 | 서재 카테고리 삭제 |
| 3.9.3 | 서재 카테고리 수정 | 서재 카테고리 수정 | 서재 카테고리 수정 | 서재 카테고리 수정 |
| 3.9.4 | 서재 카테고리 리스트 | 서재 카테고리 리스트 | 서재 카테고리 리스트 | 서재 카테고리 리스트 |
| 3.9.5 | 서재 도서 등록 | 서재 도서 등록 | 서재 도서 등록 | 서재 도서 등록 |
| 3.9.6 | 서재 도서 삭제 | 서재 도서 삭제 | 서재 도서 삭제 | 서재 도서 삭제 |
| 3.9.7 | 서재 도서(리스트) | 서재 도서(리스트) | 서재 도서(리스트) | 서재 도서(리스트) |
| 3.10 | 출력 | | | |
| 3.10.1 | 바코드 출력 | 바코드 출력 | 바코드 출력 |  |
| 3.10.2 | 청구기호 출력 | 청구기호 출력 | 청구기호 출력 |  |
| 4 | 도서관 사이트 관리 | | | |
| 4.1 | 도서 추천 관리 | | | |
| 4.1.2 | 인기 도서 리스트 | 인기 도서 리스트 | 인기 도서 리스트 | 인기 도서 리스트 |
| 4.1.3 | 사서 추천 도서 등록 | 사서 추천 도서 등록 | 사서 추천 도서 등록 |  |
| 4.1.4 | 사서 추천 도서 수정 | 사서 추천 도서 수정 | 사서 추천 도서 수정 |  |
| 4.1.5 | 사서 추천 도서 삭제 | 사서 추천 도서 삭제 | 사서 추천 도서 삭제 |  |
| 4.1.6 | 사서 추천 도서 리스트 | 사서 추천 도서 리스트 | 사서 추천 도서 리스트 |  |
| 4.1.7 | 사서 추천 도서 상세보기 | 사서 추천 도서 상세보기 | 사서 추천 도서 상세보기 | 사서 추천 도서 상세보기 |
| 4.1.8 | 신간 추천 도서 리스트 | 신간 추천 도서 리스트 | 신간 추천 도서 리스트 | 신간 추천 도서 리스트 |
| 4.2 | 공공시설 관리 | | | |
| 4.2.1 | 열람실 신청 | 열람실 신청 | 열람실 신청 | 열람실 신청 |
| 4.2.2 | 열람실 이용 연장 신청 | 열람실 이용 연장 신청 | 열람실 이용 연장 신청 | 열람실 이용 연장 신청 |
| 4.2.3 | 열람실 신청 취소 | 열람실 신청 취소 | 열람실 신청 취소 | 열람실 신청 취소 |
| 4.2.4 | 실시간 열람실 현황 | 실시간 열람실 현황 | 실시간 열람실 현황 | 실시간 열람실 현황 |
| 4.2.5 | 열람실 신청 내역 | 열람실 신청 내역 | 열람실 신청 내역 | 열람실 신청 내역 |
| 4.2.6 | 스터디룸 신청 | 스터디룸 신청 | 스터디룸 신청 | 스터디룸 신청 |
| 4.2.7 | 스터디룸 신청 취소 | 스터디룸 신청 취소 | 스터디룸 신청 취소 | 스터디룸 신청 취소 |
| 4.2.8 | 실시간 스터디룸 현황 | 실시간 스터디룸 현황 | 실시간 스터디룸 현황 | 실시간 스터디룸 현황 |
| 4.2.9 | 스터디룸 신청 내역 | 스터디룸 신청 내역 | 스터디룸 신청 내역 | 스터디룸 신청 내역 |
| 4.2.10 | 강연실 신청 | 강연실 신청 | 강연실 신청 | 강연실 신청 |
| 4.2.11 | 강연실 신청 취소 | 강연실 신청 취소 | 강연실 신청 취소 | 강연실 신청 취소 |
| 4.2.12 | 실시간 강연실 현황 | 실시간 강연실 현황 | 실시간 강연실 현황 | 실시간 강연실 현황 |
| 4.2.13 | 강연실 신청 내역 | 강연실 신청 내역 | 강연실 신청 내역 | 강연실 신청 내역 |
| 4.2.14 | 사물함 신청 | 사물함 신청 | 사물함 신청 | 사물함 신청 |
| 4.2.15 | 사물함 신청 취소 | 사물함 신청 취소 | 사물함 신청 취소 | 사물함 신청 취소 |
| 4.2.16 | 실시간 사물함 현황 | 실시간 사물함 현황 | 실시간 사물함 현황 | 실시간 사물함 현황 |
| 4.2.17 | 사물함 신청 내역 | 사물함 신청 내역 | 사물함 신청 내역 | 사물함 신청 내역 |
| 4.2.18 | 공공시설 신청 내역 검색(리스트) | 공공시설 신청 내역 검색(리스트) | 공공시설 신청 내역 검색(리스트) |  |
| 4.2.19 | 공공시설 내역 리스트 | 공공시설 내역 리스트 | 공공시설 내역 리스트 | 공공시설 내역 리스트 |
| 4.2.20 | 공공시설 등록 | 공공시설 등록 | 공공시설 등록 |  |
| 4.2.21 | 공공시설 수정 | 공공시설 수정 | 공공시설 수정 |  |
| 4.2.22 | 공공시설 삭제 | 공공시설 삭제 | 공공시설 삭제 |  |
| 4.3 | 북다이어리 | | | |
| 4.3.1 | 다이어리 일정 등록 | 다이어리 일정 등록 | 다이어리 일정 등록 | 다이어리 일정 등록 |
| 4.3.2 | 다이어리 일정 수정 | 다이어리 일정 수정 | 다이어리 일정 수정 | 다이어리 일정 수정 |
| 4.3.3 | 다이어리 일정 삭제 | 다이어리 일정 삭제 | 다이어리 일정 삭제 | 다이어리 일정 삭제 |
| 4.5 | 독후감 관리 | | | |
| 4.5.1 | 독후감 등록 | 독후감 등록 | 독후감 등록 | 독후감 등록 |
| 4.5.2 | 독후감 수정 | 독후감 수정 | 독후감 수정 | 독후감 수정 |
| 4.5.3 | 독후감 삭제 | 독후감 삭제 | 독후감 삭제 | 독후감 삭제 |
| 4.5.4 | 독후감 검색(리스트) | 독후감 검색(리스트) | 독후감 검색(리스트) | 독후감 검색(리스트) |
| 4.5.5 | 독후감 상세보기 | 독후감 상세보기 | 독후감 상세보기 | 독후감 상세보기 |
| 4.5.6 | 독후감 댓글 등록 | 독후감 댓글 등록 | 독후감 댓글 등록 | 독후감 댓글 등록 |
| 4.5.7 | 독후감 댓글 수정 | 독후감 댓글 수정 | 독후감 댓글 수정 | 독후감 댓글 수정 |
| 4.5.8 | 독후감 댓글 삭제 | 독후감 댓글 삭제 | 독후감 댓글 삭제 | 독후감 댓글 삭제 |
| 4.5.9 | 독후감 댓글 리스트 | 독후감 댓글 리스트 | 독후감 댓글 리스트 | 독후감 댓글 리스트 |
| 4.6 | 문의 관리 | | | |
| 4.6.1 | 문의 등록 | 문의 등록 | 문의 등록 | 문의 등록 |
| 4.6.2 | 문의 삭제 | 문의 삭제 | 문의 삭제 | 문의 삭제 |
| 4.6.3 | 문의 검색(리스트) | 문의 검색(리스트) | 문의 검색(리스트) | 문의 검색(리스트) |
| 4.6.4 | 문의 상세보기 | 문의 상세보기 | 문의 상세보기 | 문의 상세보기 |
| 4.6.5 | 답변 등록 | 답변 등록 | 답변 등록 |  |
| 4.6.6 | 답변 수정 | 답변 수정 | 답변 수정 |  |
| 4.6.7 | 답변 삭제 | 답변 삭제 | 답변 삭제 |  |
| 4.7 | 문화 행사 관리 | | | |
| 4.7.1 | 문화 행사 등록 | 문화 행사 등록 | 문화 행사 등록 |  |
| 4.7.2 | 문화 행사 수정 | 문화 행사 수정 | 문화 행사 수정 |  |
| 4.7.3 | 문화 행사 삭제 | 문화 행사 삭제 | 문화 행사 삭제 |  |
| 4.7.4 | 문화 행사 리스트 | 문화 행사 리스트 | 문화 행사 리스트 | 문화 행사 리스트 |
| 4.7.5 | 문화 행사 상세보기 | 문화 행사 상세보기 | 문화 행사 상세보기 | 문화 행사 상세보기 |
| 4.7.6 | 문화 행사 신청 | 문화 행사 신청 | 문화 행사 신청 | 문화 행사 신청 |
| 4.7.7 | 문화 행사 신청 취소 | 문화 행사 신청 취소 | 문화 행사 신청 취소 | 문화 행사 신청 취소 |
| 4.7.8 | 문화 행사 신청 검색(리스트) | 문화 행사 신청 검색(리스트) | 문화 행사 신청 검색(리스트) | 문화 행사 신청 검색(리스트) |
| 4.8 | 도서관 일정 관리 | | | |
| 4.8.1 | 도서관 일정 등록 | 도서관 일정 등록 | 도서관 일정 등록 |  |
| 4.8.2 | 도서관 일정 수정 | 도서관 일정 수정 | 도서관 일정 수정 |  |
| 4.8.3 | 도서관 일정 삭제 | 도서관 일정 삭제 | 도서관 일정 삭제 |  |
| 4.9 | 공지사항 관리 | | | |
| 4.9.1 | 공지사항 등록 | 공지사항 등록 | 공지사항 등록 |  |
| 4.9.2 | 공지사항 수정 | 공지사항 수정 | 공지사항 수정 |  |
| 4.9.3 | 공지사항 삭제 | 공지사항 삭제 | 공지사항 삭제 |  |
| 4.9.4 | 공지사항 상세보기 | 공지사항 상세보기 | 공지사항 상세보기 | 공지사항 상세보기 |
| 4.9.5 | 공지사항 검색(리스트) | 공지사항 검색(리스트) | 공지사항 검색(리스트) | 공지사항 검색(리스트) |
| 5 | 통계 관리 | | | |
| 5.1 | 통계 관리 | | | |
| 5.1.1 | 통계 출력 | 통계 출력 |  |  |
| 5.1.2 | 통계 조회 | 통계 조회 |  |  |

**3. 업무분담 (담당업무)**

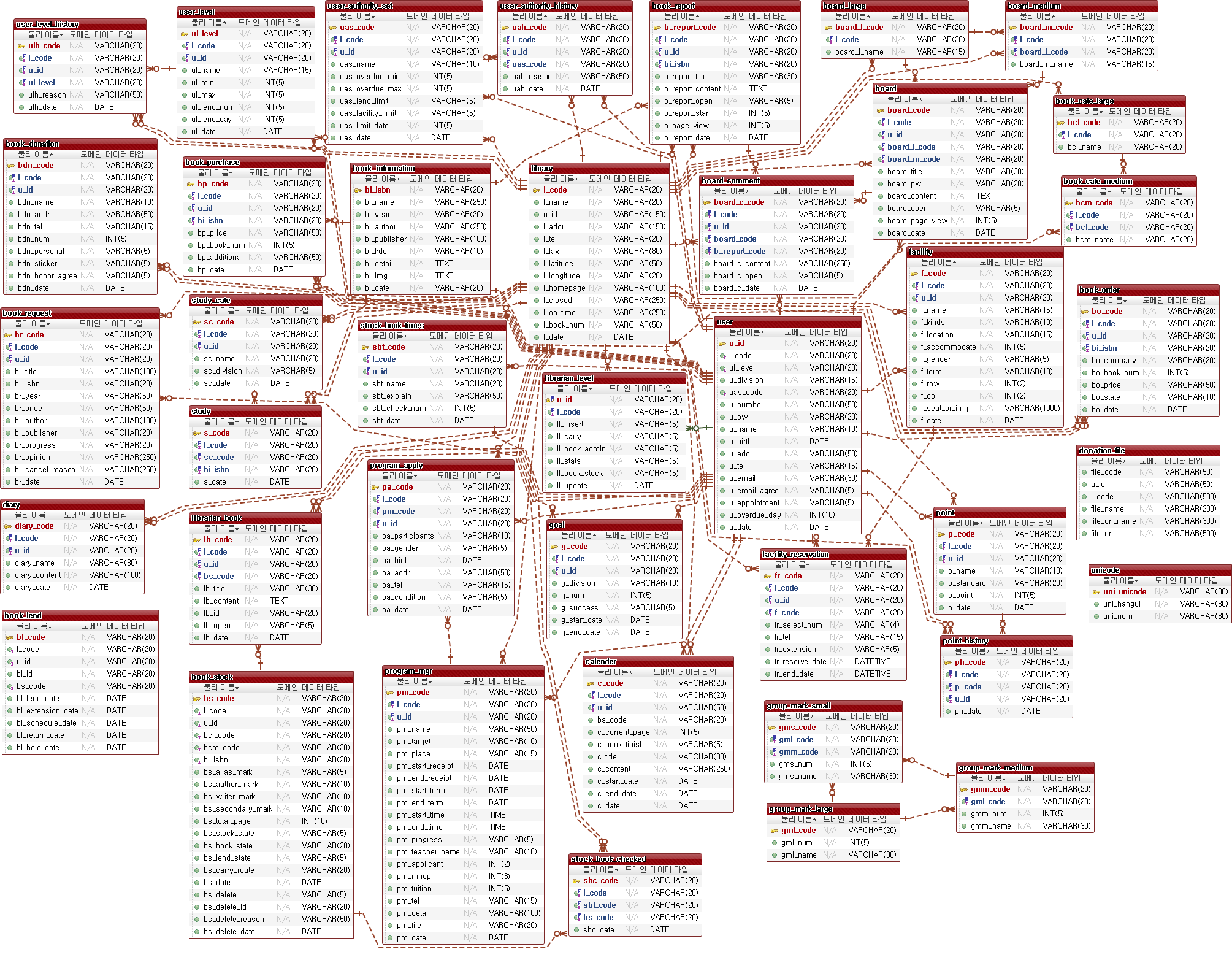
|  |  |  |
| --- | --- | --- |
| **이름** | **담당업무** | **업무내용** |
| **최지혜** | 팀장 | 회의 진행, 일정 관리, 업무 조율 |
| Hosting | 프로젝트 호스팅 계정관리 및 URL 연결 작업 및 프로젝트 배포 업무 수행 |
| 인터셉터 관리 | 도서관/관리자 페이지 인터셉터 처리 |
| 포인트 관리 | 포인트 등록, 수정, 삭제, 조회 |
| 대출/반납/예약 관리 | 대출/반납/예약 등록, 취소, 연장 |
| 북다이어리 관리 | 목표량/북캘린더/독후감 등록, 수정, 삭제, 조회 |
| 서재 관리 | 서재 카테고리 등록, 수정, 삭제, 조회 및 도서 등록, 수정, 삭제, 조회 |
| **정진영** | 공지사항 | 공지사항 등록, 수정, 삭제 |
| 문의 | 문의 등록, 수정, 삭제/ 답변등록, 삭제 |
| 추천도서 | 사서 추천도서 등록, 수정, 삭제 |
| 통계 | 메인 페이지 소장도서, 금일대출도서, 사서, 회원 Count / 지역, 연령, 나이 별 인기대출 도서 |
| **신다은** | 수서 관리 | 구매예정 / 주문 / 희망 도서(회원 신청,확인 가능)와 기부신청자의 등록, 수정, 삭제, 조회 |
| 소장도서 관리 | 소장도서 등록, 수정, 조회 |
| 삭제도서 관리 | 소장도서 중 삭제처리된 도서 수정, 삭제(DB삭제), 조회 |
| 도서관 소개 | 도서관 소개 정보를 조회한다. |
| 페이징 처리 | 도서 검색 결과에 대한 페이징 처리 |
| **김상협** | 로그인 관리 | 권한별 로그인 및 세션 등록 |
| 프로그램 관리 | 프로그램(행사 / 강좌) 등록, 수정, 삭제, 조회 |
| 공공시설 관리 | 공공 시설 등록, 수정, 삭제, 조회 , 시설 사용 연장 처리, 퇴실 처리 |
| 출력 관리 | 도서 바코드 / 회원증 출력 |
| **한우리** | 회원관리 | 회원가입, 회원/등급/권한 등록 , 수정 , 삭제, 조회, 탈퇴 처리 |
| 사서관리 | 사서 권한 설정 및 등록, 수정, 삭제, 조회 |
| 장서점검 | 장서를 점검 및 회차 관리 |
| **공동** | DB / ERD | DBMS(mySql) 관리 및 table 설계, ERD 작성 |
| 개발문서 관리 | 요구 분석, 기능 설계, ERD, UML을 포함한 전체 개발문서 분담 |

**4.** **설계사양서**

**4.1 ERD 논리모드**

****

**4.2 ERD 물리모드**

****

**4.3 테이블 구조**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| board 게시글 내용 | | | | |
| board\_code(PK) | 게시글내용 코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| board\_l\_code(FK) | 게시판 대분류코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| board\_m\_code(FK) | 게시판 중분류코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| board\_title | 게시글제목 | N/A | VARCHAR(30) | NOT NULL |
| board\_pw | 게시글비번 | N/A | VARCHAR(20) | NULL |
| board\_content | 게시글내용 | N/A | TEXT | NOT NULL |
| board\_open | 공개여부 | N/A | VARCHAR(5) | NULL |
| board\_page\_view | 조회수 | N/A | INT(5) | NOT NULL |
| board\_date | 게시글 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| board\_comment댓글 | | | | |
| board\_c\_code(PK) | 댓글코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NULL |
| board\_code(FK) | 게시글내용 코드(FK) | N/A | VARCHAR(20) | NULL |
| b\_report\_code(FK) | 독후감코드(FK) | N/A | VARCHAR(20) | NULL |
| board\_c\_content | 내용 | N/A | VARCHAR(250) | NOT NULL |
| board\_c\_open | 공개여부 | N/A | VARCHAR(5) | NULL |
| board\_c\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| board\_large 게시판 대분류 | | | | |
| board\_l\_code(PK) | 게시판 대분류코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| board\_l\_name | 게시판 대분류명 | N/A | VARCHAR(15) | NOT NULL |
|  |  |  |  |  |
| board\_medium 게시판 중분류 | | | | |
| board\_m\_code(PK) | 게시판 중분류코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| board\_l\_code(FK) | 게시판 대분류코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| board\_m\_name | 게시판 중분류명 | N/A | VARCHAR(15) | NOT NULL |
|  |  |  |  |  |
| book\_cate\_large 도서 대분류 | | | | |
| bcl\_code(PK) | 대분류코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| bcl\_name | 대분류명 | N/A | VARCHAR(20) | NOT NULL |
|  |  |  |  |  |
| book\_cate\_medium 도서 중분류 | | | | |
| bcm\_code(PK) | 중분류코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| bcl\_code(FK) | 대분류코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| bcm\_name | 중분류명 | N/A | VARCHAR(20) | NOT NULL |
|  |  |  |  |  |
| book\_donation 기증도서 신청자 | | | | |
| bdn\_code(PK) | 기증 신청자 코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 관리자 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| bdn\_name | 이름 | N/A | VARCHAR(10) | NOT NULL |
| bdn\_addr | 주소 | N/A | VARCHAR(50) | NOT NULL |
| bdn\_tel | 연락처 | N/A | VARCHAR(15) | NOT NULL |
| bdn\_num | 기증 도서 권수 | N/A | INT(5) | NOT NULL |
| bdn\_personal | 개인정보 수집 동의 여부 | N/A | VARCHAR(5) | NOT NULL |
| bdn\_sticker | 표제지 기증스티커 부착여부 | N/A | VARCHAR(5) | NOT NULL |
| bdn\_honor\_agree | 명예전당 등록여부 | N/A | VARCHAR(5) | NOT NULL |
| bdn\_date | 신청일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| book\_information 도서정보 | | | | |
| bi\_isbn(PK) | ISBN(PK) | N/A | VARCHAR(20) | NOT NULL |
| bi\_name | 도서명 | N/A | VARCHAR(250) | NOT NULL |
| bi\_year | 발행년도 | N/A | VARCHAR(20) | NOT NULL |
| bi\_author | 저자 | N/A | VARCHAR(250) | NOT NULL |
| bi\_publisher | 발행자 | N/A | VARCHAR(100) | NOT NULL |
| bi\_kdc | KDC | N/A | VARCHAR(10) | NULL |
| bi\_detail | 상세설명 | N/A | TEXT | NULL |
| bi\_img | 이미지주소 | N/A | TEXT | NULL |
| bi\_date | 등록일 | N/A | VARCHAR(20) | NOT NULL |
|  |  |  |  |  |
| book\_lend 대출도서 | | | | |
| bl\_code(PK) | 대출도서코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code | 도서관코드 | N/A | VARCHAR(20) | NOT NULL |
| u\_id | 관리자아이디 | N/A | VARCHAR(20) | NULL |
| bl\_id | 대출자아이디 | N/A | VARCHAR(20) | NOT NULL |
| bs\_code | 소장도서코드 | N/A | VARCHAR(20) | NOT NULL |
| bl\_lend\_date | 대출일 | N/A | DATE | NULL |
| bl\_extension\_date | 연장일 | N/A | DATE | NULL |
| bl\_schedule\_date | 반납예정일 | N/A | DATE | NULL |
| bl\_return\_date | 반납일 | N/A | DATE | NULL |
| bl\_hold\_date | 예약일 | N/A | DATE | NULL |
|  |  |  |  |  |
| book\_order 주문도서 | | | | |
| bo\_code(PK) | 주문도서코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 작성자(FK) | N/A | VARCHAR(20) | NOT NULL |
| bi\_isbn(FK) | 도서정보코드(ISBN)(FK) | N/A | VARCHAR(20) | NOT NULL |
| bo\_company | 주문업체명 | N/A | VARCHAR(20) | NOT NULL |
| bo\_book\_num | 주문권수 | N/A | INT(5) | NOT NULL |
| bo\_price | 결제가격 | N/A | VARCHAR(50) | NOT NULL |
| bo\_state | 주문상태 | N/A | VARCHAR(10) | NOT NULL |
| bo\_date | 주문등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| book\_purchase 구입예정 | | | | |
| bp\_code(PK) | 구입도서코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| bi\_isbn(FK) | 도서정보코드(ISBN)(FK) | N/A | VARCHAR(20) | NOT NULL |
| bp\_price | 도서가격 | N/A | VARCHAR(50) | NOT NULL |
| bp\_book\_num | 구입예정권수 | N/A | INT(5) | NOT NULL |
| bp\_additional | 추가사항 | N/A | VARCHAR(50) | NULL |
| bp\_date | 등록일자 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| book\_report 독후감 | | | | |
| b\_report\_code(PK) | 독후감코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| bi\_isbn(FK) | 도서정보코드(ISBN)(FK) | N/A | VARCHAR(20) | NOT NULL |
| b\_report\_title | 제목 | N/A | VARCHAR(30) | NOT NULL |
| b\_report\_content | 내용 | N/A | TEXT | NOT NULL |
| b\_report\_open | 공개여부 | N/A | VARCHAR(5) | NOT NULL |
| b\_report\_star | 나의평점 | N/A | INT(5) | NULL |
| b\_page\_view | 조회수 | N/A | INT(5) | NOT NULL |
| b\_report\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| book\_request 희망도서 | | | | |
| br\_code(PK) | 희망도서코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NULL |
| br\_title | 도서명 | N/A | VARCHAR(100) | NOT NULL |
| br\_isbn | ISBN | N/A | VARCHAR(20) | NOT NULL |
| br\_year | 발행년도 | N/A | VARCHAR(50) | NOT NULL |
| br\_price | 가격 | N/A | VARCHAR(50) | NOT NULL |
| br\_author | 저자명 | N/A | VARCHAR(100) | NOT NULL |
| br\_publisher | 발행자 | N/A | VARCHAR(20) | NOT NULL |
| br\_progress | 진행상황 | N/A | VARCHAR(20) | NOT NULL |
| br\_opinion | 의견 | N/A | VARCHAR(250) | NULL |
| br\_cancel\_reason | 취소사유 | N/A | VARCHAR(250) | NULL |
| br\_date | 신청일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| book\_stock 소장도서 | | | | |
| bs\_code(PK) | 소장도서코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code | 도서관코드 | N/A | VARCHAR(20) | NOT NULL |
| u\_id | 아이디 | N/A | VARCHAR(20) | NOT NULL |
| bcl\_code | 대분류코드 | N/A | VARCHAR(20) | NOT NULL |
| bcm\_code | 중분류코드 | N/A | VARCHAR(20) | NOT NULL |
| bi\_isbn | 도서정보코드(ISBN) | N/A | VARCHAR(20) | NOT NULL |
| bs\_alias\_mark | 별치기호 | N/A | VARCHAR(5) | NULL |
| bs\_author\_mark | 저자기호 | N/A | VARCHAR(10) | NOT NULL |
| bs\_writer\_mark | 저작기호 | N/A | VARCHAR(10) | NOT NULL |
| bs\_secondary\_mark | 부차적기호 | N/A | VARCHAR(10) | NULL |
| bs\_total\_page | 총페이지 | N/A | INT(10) | NOT NULL |
| bs\_stock\_state | 소장상태 | N/A | VARCHAR(5) | NOT NULL |
| bs\_book\_state | 책의상태 | N/A | VARCHAR(20) | NOT NULL |
| bs\_lend\_state | 대출가능상태 | N/A | VARCHAR(5) | NOT NULL |
| bs\_carry\_route | 반입구 구분 | N/A | VARCHAR(20) | NOT NULL |
| bs\_date | 등록일자 | N/A | DATE | NOT NULL |
| bs\_delete | 도서삭제 | N/A | VARCHAR(5) | NOT NULL |
| bs\_delete\_id | 아이디(삭제자) | N/A | VARCHAR(20) | NULL |
| bs\_delete\_reason | 삭제사유 | N/A | VARCHAR(50) | NULL |
| bs\_delete\_date | 삭제등록일 | N/A | DATE | NULL |
|  |  |  |  |  |
| calender 캘린더 | | | | |
| c\_code(PK) | 캘린더코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(50) | NOT NULL |
| bs\_code | 소장도서코드 | N/A | VARCHAR(20) | NOT NULL |
| c\_current\_page | 현재페이지 | N/A | INT(5) | NOT NULL |
| c\_book\_finish | 완독 | N/A | VARCHAR(5) | NOT NULL |
| c\_title | 제목 | N/A | VARCHAR(30) | NOT NULL |
| c\_content | 내용 | N/A | VARCHAR(250) | NOT NULL |
| c\_start\_date | 일정시작일 | N/A | DATE | NULL |
| c\_end\_date | 일정종료일 | N/A | DATE | NULL |
| c\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| diary 독서 한줄일기 | | | | |
| diary\_code(PK) | 독서 한줄일기 코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| diary\_name | 도서명 | N/A | VARCHAR(30) | NOT NULL |
| diary\_content | 내용 | N/A | VARCHAR(100) | NOT NULL |
| diary\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| donation\_file 기증도서 파일 | | | | |
| file\_code | 기증신청서파일코드 | N/A | VARCHAR(50) | NOT NULL |
| u\_id | 작성자 | N/A | VARCHAR(50) | NOT NULL |
| l\_code | 도서관코드 | N/A | VARCHAR(500) | NOT NULL |
| file\_name | 파일명 | N/A | VARCHAR(200) | NOT NULL |
| file\_ori\_name | 파일원래이름 | N/A | VARCHAR(300) | NOT NULL |
| file\_url | 파일url | N/A | VARCHAR(500) | NOT NULL |
|  |  |  |  |  |
| facility 시설대실 | | | | |
| f\_code(PK) | 시설코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 관리자(FK) | N/A | VARCHAR(20) | NOT NULL |
| f\_name | 시설명 | N/A | VARCHAR(15) | NOT NULL |
| f\_kinds | 시설종류 | N/A | VARCHAR(10) | NOT NULL |
| f\_location | 위치 | N/A | VARCHAR(15) | NOT NULL |
| f\_accommodate | 최대 사용 인원 | N/A | INT(5) | NOT NULL |
| f\_gender | 이용가능성별 | N/A | VARCHAR(5) | NOT NULL |
| f\_term | 이용기간 | N/A | VARCHAR(10) | NOT NULL |
| f\_row | 테이블 가로 | N/A | INT(2) | NOT NULL |
| f\_col | 테이블 세로 | N/A | INT(2) | NOT NULL |
| f\_seat\_or\_img | 좌석문자열 / 이미지 | N/A | VARCHAR(1000) | NULL |
| f\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| facility\_reservation 시설예약 | | | | |
| fr\_code(PK) | 신청코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| f\_code(FK) | 시설코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| fr\_select\_num | 선택번호 | N/A | VARCHAR(4) | NULL |
| fr\_tel | 상시연락처 | N/A | VARCHAR(15) | NULL |
| fr\_extension | 연장여부 | N/A | VARCHAR(5) | NULL |
| fr\_reserve\_date | 예약일시 | N/A | DATETIME | NOT NULL |
| fr\_end\_date | 종료일시 | N/A | DATETIME | NOT NULL |
|  |  |  |  |  |
| goal 목표량 | | | | |
| g\_code(PK) | 목표량코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| g\_division | 목표구분 | N/A | VARCHAR(10) | NOT NULL |
| g\_num | 목표권수 | N/A | INT(5) | NOT NULL |
| g\_success | 목표달성여부 | N/A | VARCHAR(5) | NOT NULL |
| g\_start\_date | 목표 등록일 | N/A | DATE | NOT NULL |
| g\_end\_date | 목표 달성일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| group\_mark\_large 분류기호 대분류 | | | | |
| gml\_code(PK) | 분류기호 대분류 코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| gml\_num | 대분류 번호 | N/A | INT(5) | NOT NULL |
| gml\_name | 대분류명 | N/A | VARCHAR(30) | NOT NULL |
|  |  |  |  |  |
| group\_mark\_medium 분류기호 중분류 | | | | |
| gmm\_code(PK) | 분류기호 중분류 코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| gml\_code(FK) | 분류기호 대분류 코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| gmm\_num | 중분류 번호 | N/A | INT(5) | NOT NULL |
| gmm\_name | 중분류명 | N/A | VARCHAR(30) | NOT NULL |
|  |  |  |  |  |
| group\_mark\_small 분류기호 소분류 | | | | |
| gms\_code(PK) | 분류기호 소분류 코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| gml\_code(FK) | 분류기호 대분류 코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| gmm\_code(FK) | 분류기호 중분류 코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| gms\_num | 소분류 번호 | N/A | INT(5) | NOT NULL |
| gms\_name | 소분류명 | N/A | VARCHAR(30) | NOT NULL |
|  |  |  |  |  |
| librarian\_book 사서추천도서 | | | | |
| lb\_code(PK) | 추천도서코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| bs\_code(FK) | 소장도서코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| lb\_title | 사서추천제목 | N/A | VARCHAR(30) | NULL |
| lb\_content | 사서추천내용 | N/A | TEXT | NOT NULL |
| lb\_id | 아이디(추천인) | N/A | VARCHAR(20) | NOT NULL |
| lb\_open | 공개여부 | N/A | VARCHAR(5) | NOT NULL |
| lb\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| librarian\_level 사서권한 | | | | |
| u\_id(PK)(FK) | 아이디(PK)(FK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| ll\_insert | 사서등록 | N/A | VARCHAR(5) | NOT NULL |
| ll\_carry | 수서 | N/A | VARCHAR(5) | NOT NULL |
| ll\_book\_admin | 대출,반납,예약 | N/A | VARCHAR(5) | NOT NULL |
| ll\_stats | 통계 | N/A | VARCHAR(5) | NOT NULL |
| ll\_book\_stock | 장서점검 | N/A | VARCHAR(5) | NOT NULL |
| ll\_update | 최종 수정일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| library 도서관 | | | | |
| l\_code(PK) | 도서관 코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_name | 도서관명 | N/A | VARCHAR(20) | NOT NULL |
| u\_id | 등록자 id | N/A | VARCHAR(150) | NULL |
| l\_addr | 주소 | N/A | VARCHAR(150) | NULL |
| l\_tel | 전화번호 | N/A | VARCHAR(20) | NULL |
| l\_fax | 팩스 | N/A | VARCHAR(80) | NULL |
| l\_latitude | 위도 | N/A | VARCHAR(50) | NULL |
| l\_longitude | 경도 | N/A | VARCHAR(20) | NULL |
| l\_homepage | 홈페이지 주소 | N/A | VARCHAR(100) | NULL |
| l\_closed | 휴관일 | N/A | VARCHAR(250) | NULL |
| l\_op\_time | 운영시간 | N/A | VARCHAR(250) | NULL |
| l\_book\_num | 총 권수 | N/A | VARCHAR(50) | NULL |
| l\_date | 등록일 | N/A | DATE | NULL |
|  |  |  |  |  |
| point 포인트 | | | | |
| p\_code(PK) | 포인트코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| p\_name | 포인트명 | N/A | VARCHAR(10) | NOT NULL |
| p\_standard | 포인트 지급기준 | N/A | VARCHAR(20) | NOT NULL |
| p\_point | 포인트 | N/A | INT(5) | NOT NULL |
| p\_date | 등록일 | N/A | DATE | NULL |
|  |  |  |  |  |
| point\_history 포인트 이력 | | | | |
| ph\_code(PK) | 포인트 이력코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| p\_code(FK) | 포인트코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| ph\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| program\_apply 프로그램 신청 | | | | |
| pa\_code(PK) | 프로그램 신청 코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| pm\_code(FK) | 프로그램 종합 코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| pa\_participants | 참여자 이름 | N/A | VARCHAR(10) | NOT NULL |
| pa\_gender | 성별 | N/A | VARCHAR(5) | NOT NULL |
| pa\_birth | 생년월일 | N/A | DATE | NOT NULL |
| pa\_addr | 주소 | N/A | VARCHAR(50) | NOT NULL |
| pa\_tel | 연락처 | N/A | VARCHAR(15) | NOT NULL |
| pa\_condition | 상태 | N/A | VARCHAR(5) | NOT NULL |
| pa\_date | 신청일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| program\_mgr 프로그램 관리 | | | | |
| pm\_code(PK) | 프로그램 종합 코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| pm\_name | 프로그램명 | N/A | VARCHAR(50) | NOT NULL |
| pm\_target | 대상자(신청대상) | N/A | VARCHAR(10) | NOT NULL |
| pm\_place | 장소 | N/A | VARCHAR(15) | NOT NULL |
| pm\_start\_receipt | 접수기간(시작) | N/A | DATE | NOT NULL |
| pm\_end\_receipt | 접수기간(종료) | N/A | DATE | NOT NULL |
| pm\_start\_term | 운영기간(시작) | N/A | DATE | NOT NULL |
| pm\_end\_term | 운영기간(종료) | N/A | DATE | NOT NULL |
| pm\_start\_time | 운영시간(시작) | N/A | TIME | NOT NULL |
| pm\_end\_time | 운영시간(종료) | N/A | TIME | NOT NULL |
| pm\_progress | 진행상황 | N/A | VARCHAR(5) | NULL |
| pm\_teacher\_name | 강사명 | N/A | VARCHAR(10) | NOT NULL |
| pm\_applicant | 현재 신청인원 | N/A | INT(2) | NULL |
| pm\_mnop | 신청 최대인원수 | N/A | INT(3) | NOT NULL |
| pm\_tuition | 수강료 | N/A | INT(5) | NOT NULL |
| pm\_tel | 문의연락처 | N/A | VARCHAR(15) | NOT NULL |
| pm\_detail | 세부사항 | N/A | VARCHAR(100) | NOT NULL |
| pm\_file | 첨부파일 | N/A | VARCHAR(20) | NULL |
| pm\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| stock\_book\_checked 장서점검도서 | | | | |
| sbc\_code(PK) | 장서점검도서코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| sbt\_code(FK) | 장서회차코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| bs\_code(FK) | 소장도서코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| sbc\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| stock\_book\_times장서점검회차 | | | | |
| sbt\_code(PK) | 장서회차코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| sbt\_name | 회차명 | N/A | VARCHAR(20) | NOT NULL |
| sbt\_explain | 회차 설명 | N/A | VARCHAR(50) | NULL |
| sbt\_check\_num | 점검 권수 | N/A | INT(5) | NOT NULL |
| sbt\_date | 점검일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| study 서재 | | | | |
| s\_code(PK) | 서재코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| sc\_code(FK) | 카테고리 코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| bi\_isbn(FK) | 도서정보코드(ISBN)(FK) | N/A | VARCHAR(20) | NOT NULL |
| s\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| study\_cate 서재 카테고리 | | | | |
| sc\_code(PK) | 카테고리 코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 아이디(FK) | N/A | VARCHAR(20) | NOT NULL |
| sc\_name | 카테고리명 | N/A | VARCHAR(20) | NOT NULL |
| sc\_division | 카테고리구분 | N/A | VARCHAR(5) | NOT NULL |
| sc\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| unicode 유니코드 | | | | |
| uni\_unicode(PK) | 유니코드코드(PK) | N/A | VARCHAR(30) | NOT NULL |
| uni\_hangul | 5표한글 | N/A | VARCHAR(30) | NULL |
| uni\_num | 5표숫자 | N/A | VARCHAR(30) | NULL |
|  |  |  |  |  |
| user 회원 | | | | |
| u\_id(PK) | 아이디(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code | 도서관코드 | N/A | VARCHAR(20) | NOT NULL |
| ul\_level | 회원등급 | N/A | VARCHAR(20) | NULL |
| u\_division | 관리자/사서/일반구분 | N/A | VARCHAR(15) | NOT NULL |
| uas\_code | 회원제한코드 | N/A | VARCHAR(20) | NULL |
| u\_number | 모바일회원 | N/A | VARCHAR(50) | NULL |
| u\_pw | 비밀번호 | N/A | VARCHAR(20) | NOT NULL |
| u\_name | 이름 | N/A | VARCHAR(10) | NULL |
| u\_birth | 생년월일 | N/A | DATE | NOT NULL |
| u\_addr | 주소 | N/A | VARCHAR(50) | NOT NULL |
| u\_tel | 연락처 | N/A | VARCHAR(15) | NOT NULL |
| u\_email | 이메일 | N/A | VARCHAR(30) | NOT NULL |
| u\_email\_agree | 이메일수신동의 | N/A | VARCHAR(5) | NOT NULL |
| u\_appointment | 임명장수여 | N/A | VARCHAR(5) | NOT NULL |
| u\_overdue\_day | 연체일 | N/A | INT(10) | NOT NULL |
| u\_date | 회원등록일자 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| user\_authority\_history 회원 사용권한 변동 이력 | | | | |
| uah\_code(PK) | 회원사용권한변동이력코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 회원만(FK) | N/A | VARCHAR(20) | NOT NULL |
| uas\_code(FK) | 회원제한코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| uah\_reason | 회원사용권한변동 사유 | N/A | VARCHAR(50) | NOT NULL |
| uah\_date | 동록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| user\_authority\_set 회원 사용권한 설정 | | | | |
| uas\_code(PK) | 회원제한코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 관리자(FK) | N/A | VARCHAR(20) | NOT NULL |
| uas\_name | 권한명 | N/A | VARCHAR(10) | NOT NULL |
| uas\_overdue\_min | 연체일(최소) | N/A | INT(5) | NOT NULL |
| uas\_overdue\_max | 연체일(최대) | N/A | INT(5) | NOT NULL |
| uas\_lend\_limit | 대출제한 | N/A | VARCHAR(5) | NOT NULL |
| uas\_facility\_limit | 공공시설 | N/A | VARCHAR(5) | NOT NULL |
| uas\_limit\_date | 제한일 | N/A | INT(5) | NOT NULL |
| uas\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| user\_level 회원등급 설정 | | | | |
| ul\_level(PK) | 회원등급(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 관리자(FK) | N/A | VARCHAR(20) | NOT NULL |
| ul\_name | 등급명 | N/A | VARCHAR(15) | NOT NULL |
| ul\_min | 권수 | N/A | INT(5) | NOT NULL |
| ul\_max | 권수 | N/A | INT(5) | NOT NULL |
| ul\_lend\_num | 대출가능권수 | N/A | INT(5) | NOT NULL |
| ul\_lend\_day | 대출가능일수 | N/A | INT(5) | NOT NULL |
| ul\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |
| user\_level\_history 회원등급 변동 이력 | | | | |
| ulh\_code(PK) | 회원등급변동이력코드(PK) | N/A | VARCHAR(20) | NOT NULL |
| l\_code(FK) | 도서관코드(FK) | N/A | VARCHAR(20) | NOT NULL |
| u\_id(FK) | 회원전체(FK) | N/A | VARCHAR(20) | NOT NULL |
| ul\_level(FK) | 회원등급(FK) | N/A | VARCHAR(20) | NOT NULL |
| ulh\_reason | 회원등급변동사유 | N/A | VARCHAR(50) | NOT NULL |
| ulh\_date | 등록일 | N/A | DATE | NOT NULL |
|  |  |  |  |  |

**4.4 네이밍 규칙**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Directory** | 1. directory의 모든 명칭은 영문자(소문자)로 구성되면, 반드시 영문자로 시작한다.   2. <view>하위 메뉴 깊이를 기준으로 directory를 생성하며, 의미를 알 수 있는 약어만을 사용한다.  3. 공통 모듈은 공용 directory를 사용한다. 공용으로 사용이 불가 할 경우 자체 directory에서 module 임을 알 수 있도록 표현한다. | | | | |
| **html(view)** | 1. 해당 directory명칭을 말머리로 갖고 파일의 주요기능을 표현한다. 2. 모든 Section은 영문자(소문자)로 시작되며 카멜표현식을 사용한다. 3. 가급적 의미를 알 수 없는 약어는 사용하지 않는다. | | | | |
| 일반JSP 파일 | directory명칭 | 메뉴명에 대한 풀네임 또는 약어 | | ex>  구입 예정 도서 등록  bookPurchaseInsert.html  프로그램 신청 검색 리스트  programApplySearchList.html |
| 주요기능 | 리스트, 목록 | List.,SearchList |
| 입력 | Insert |
| 수정 | Update |
| 상세조회 | Detail |
| 삭제 | Delete |
| **package** | 1. 모든 package는 하단에 표기된 상위 package를 갖으며, 영문자(소문자)로만 표현한다. | | | | |
| 상위 package 명 | ksmart.pentagon | | | |
| **Class** | 1. 모든 class는 영문자로만 구성하되 반드시 첫글자는 대문자로 시작하며, 카멜표현식으로 표현한다. | | | | |
| DAO | 모든 DAO는 해당 테이블명을카멜식으로 표현, 마지막에 Dao를 붙인다. | | | |
| DTO | 모든 DTO는 해당 테이블명 뒤에 DTO를 붙이지 않는다. | | | |
| Controller | 모든 Controller는 해당 테이블명을카멜식으로 표현하고 뒤에 Controller를 붙인다. | | | |
| Service | 모든 Service는 해당 테이블명을카멜식으로 표현하고 뒤에 Service를 붙인다. | | | |
| Mapper | 모든 Mapper는 해당 테이블명을카멜식으로 표현하고 뒤에 Mapper를 붙인다. | | | |
| **Object/Variable** | 1. 객체 및 변수는 영문자만을 또는 영문자와 숫자로 구성된 카멜표현식을 사용하고, 첫 글자는 반드시 소문자로 시작하며 의미를 알 수 없는 약어는 사용하지 않는다. 2. 변수의 경우, 단순 증감이며 for문 등의 제어문 내에서 선언되는 경우 <i, j, k>등과 같은 간단한 소문자를 사용한다. ex> for(int I = 0; I < 10; I++) 3. 특정 정보를 담고 있는 변수의 경우, 해당 정보의 성격을 나타낼 수 있는 명칭을 반드시 사용하여야 한다. ex>String bsCode = “소장도서 코드”; 4. HTML내의 <input>태그로 사용되는 필드 값을 전달 받을 경우 동일한 이름의 변수에 할당한다. | | | | |
| **DB** | 1. DB의 테이블 및 칼럼의 모든 명칭은 underscore표현식을 사용하며 영문자(소문자)로 시작한다. 2. 테이블명칭은 해당 주요기능을 표현하며, 칼럼 또한 칼럼의 속성이 표현될 수 있는 명칭을 사용한다. 3. 의미를 알 수 있는 약어는 사용하지 않으며 되도록 풀네임을 사용한다. | | | | |

**4.5 디렉토리 구조**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **상위** | **중위** | **하위** | **파일명** | **설명** |
| **view** | **intro** |  | intro.html | 인트로 화면 |
| **adminpage** | **userManagement** | adminUserUpdate | 회원 수정 화면 |
| userSearchList | 회원 전체 리스트 |
| adminUserDetail | 회원 정보 상세보기 화면 |
| adUserAuthorityInsert | 회원사용권한 등록화면 |
| adUserAuthorityUpdate | 회원사용권한 수정화면 |
| adUserAuthorityList | 회원사용권한 리스트화면 |
| adUserAuthorityHistorySearchList | 회원 사용 권한이력 리스트화면 |
| adUserLevelInsert | 회원 등급 등록 화면 |
| adUserLevelUpdate | 회원 등급 수정 화면 |
| adUserLevelList | 회원 등급 검색 화면 |
| adUserLevelHistorySearchList | 회원 등급 이력 리스트 화면 |
| **librarian** | adminLogin | 관리자 로그인 화면 |
| librarianInsert | 사서 등록 화면(관리자) |
| librarianLevelUpdate | 사서 정보/권한 수정 화면(관리자) |
| librarianLevelList | 사서 정보/권한 전체 리스트(관리자) |
| librarianDetail | 사서 정보 상세보기 화면(관리자) |
| librarianMyUpdate | 사서 내정보 수정 화면(사서) |
| librarianMyDetail | 사서 내 정보/권한 상세보기(사서) |
| **point** | pointList | 포인트 리스트 화면 |
| pointHistorySearchList | 포인트 이력 리스트 화면 |
| **bookLend** | lendSearchList | 대출 도서 전체 리스트 |
| holdSearchList | 예약 전체 리스트 |
| **bookCarry** | requestSearchList | 희망도서신청 전체 리스트 |
| requestDetail | 희망도서신청 상세 내용 |
| bookPurchaseList | 구입예정도서 전체 리스트 |
| bookPurchaseInsert | 구입예정도서 등록 화면 |
| bookPurchaseUpdate | 구입예정도서 상세&수정 화면 |
| bookOrderList | 주문도서 전체 리스트 |
| bookOrderInsert | 주문도서 등록 화면 |
| bookOrderUpdate | 주문도서 상세&수정 화면 |
| bookDonationInsert | 기증도서 신청자 등록 화면 |
| bookDonationUpdate | 기증도서 신청자 상세&수정 화면 |
| bookDonationList | 기증도서 신청자 전체 리스트 |
| **bookStock** | stockSearchList | 소장도서 전체 리스트 |
| stockDetail | 소장도서 상세 화면 |
| stockDetailInsert | 소장도서 등록 화면 |
| stockDetailUpdate | 소장도서 수정 화면 |
| stockDeleteList | 삭제도서 전체 리스트 |
| stockDeleteDetail | 삭제도서 상세&복구 화면 |
| **stockCheck** | stockCheckInsert | 장서점검 회차등록 |
| stockCheckUpdate | 장서점검 회차 수정 화면 |
| stockCheckList | 장서점검 회차 전체 리스트 |
| stockCheckBookList | 장서점검 도서 전체 리스트 |
| stockLossBookList | 분실 도서 전체 리스트 |
| **static** | bookStatistic | 도서 통계 화면 |
| **board** | noticeInsert | 공지사항 등록 화면 |
| noticeUpdate | 공지사항 수정 화면 |
| noticeSearchList | 공지사항 전체 리스트 화면 |
| noticeDetail | 공지사항 상세보기 화면 |
| adminInquirySearchList | 문의 전체 리스트 화면 |
| adminInquiryDetail | 문의 상세 보기 화면 |
| **facility** | facilityInsert | 공공시설 등록 화면 |
| facilityUpdate | 공공시설 수정 화면 |
| facilityDetail | 공공시설 상세보기 화면 |
| facilityList | 공공시설 리스트 화면 |
| facilityReservationSearchList | 공공시설 신청 내역 전체 리스트 화면 |
| **program** | programInsert | 프로그램(행사,강좌) 등록 화면 |
| programDetail | 프로그램(행사, 강좌) 상세보기 |
| programUpdate | 프로그램(행사,강좌) 수정 화면 |
| programSearchList | 프로그램(행사,강좌) 전체 리스트 화면 |
| programApplySearchList | 프로그램(행사,강좌) 신청 전체 리스트화면 |
| **bookRecommend** | bookRecommendInsert | 도서 추천 등록 화면 |
| bookRecommendUpdate | 도서 추천 수정 화면 |
| bookRecommendDetail | 도서 추천 상세보기 |
| bookRecommendList | 도서 추천 리스트 |
| **calender** | adminCalender | 캘린더 메인 화면 |
| adminCalenderInsert | 캘린더 일정 등록 화면 |
| adminCalenderUpdate | 캘린더 일정 수정 화면 |
| **librarypage** | **user** | login | 로그인 화면 |
| userInsert | 회원 등록 화면 |
| myUserUpdate | 회원 수정 화면 |
| myUserDetail | 회원 정보 상세보기 화면 |
| userFindIdPw | 회원 ID/PW찾기 화면 |
| myUserDelete | 회원 탈퇴 페이지 |
| **facility** | lectureRoomReservation | 강연실 안내정보 + 강연실 실시간 현황 |
| studyRoomReservation | 스터디룸 안내정보 + 스터디룸 실시간 현황 |
| readingRoomReservation | 열람실 안내정보 + 열람실 실시간 현황 |
| lockerReservation | 사물함 안내정보 + 사물함 실시간 현황 |
| myLectureRoomReserveList | 마이페이지 - 강연실 신청내역 |
| myStudyRoomReserveList | 마이페이지 - 스터디룸 신청내역 |
| myReadingRoomReserveList | 마이페이지 - 열람실 신청내역 |
| myLockerReserveList | 마이페이지 - 사물함 신청내역 |
| reservationApply | 시설 예약 신청화면 |
| **program** | programSearchList | 프로그램(행사,강좌) 리스트 출력 |
| programDetail | 프로그램(행사,강좌) 상세보기 |
| programApply | 프로그램(행사,강좌) 신청 화면 |
| myProgramApplyList | 마이페이지 - 프로그램(행사,강좌) 신청 리스트 |
| myProgramApplyDetail | 마이페이지 - 프로그램 신청리스트 상세보기 |
|  | 마이페이지 - 프로그램 신청 취소 |
| **community** | noticeList | 공지사항 리스트 화면 |
| inquirySearchList | 문의 내역 리스트 화면 |
| myInquiryList | 마이페이지 - 내 문의내역 리스트 |
| inquiryInsert | 문의 등록 화면 |
| inquiryDetail | 문의 상세보기 화면 |
| **bookReport** | bookReportSearchList | 독후 전체리스트 화면 |
| myBookReportSearchList | 마이페이지 - 내 독후감 전체 리스트 |
| bookReportInsert | 독후감 작성 화면 |
| bookReportDetail | 독후감 상세보기 화면 |
| bookReportUpdate | 독후감 수정 화면 |
| **calender** | myCalender | 마이페이지 - 캘린더 메인 화면 |
| **bookData** | bookDataSearchList | 도서 자료 검색 리스트 |
| bookDataSearchGrid | 도서 자료 검색 리스트 |
| bookDataSearch | 도서 검색창 화면 |
| bookDataSearchDetail | 도서 상세 검색창 화면 |
| bookDetail | 도서 상세보기 화면 |
| **book** | bookRequestIntro | 희망 도서 신청 안내 화면 |
| bookRequestInsert | 희망 도서 신청 화면 |
| myBookRequestList | 마이페이지 - 희망 도서 신청 리스트 화면 |
| bookDonationGuide | 도서 기증 안내 화면 |
| myStudyList | 마이페이지 - 서재리스트 |
| myStudyCateInsert | 마이페이지 - 서재카테고리 등록 화면 |
| myStudyCateUpdate | 마이페이지 - 서재카테고리 수정 화면 |
| myStudyBookUpdate | 마이페이지 - 서재 도서 수정 화면 |
| myLendList | 마이페이지 - 대출리스트 |
| myHoldList | 마이페이지 - 예약리스트 |
| **point** | myPointList | 마이페이지-포인트사용리스트 |
| **libraryInfo** | libraryInformation | 도서관 소개 화면 |

**4.6 패키지구조**

|  |  |  |
| --- | --- | --- |
| **패키지명** | **클래스명** | **설명** |
| ksmart.pentagon.vo | User | 회원 vo |
| UserAuthoritySet | 회원사용권한설정 vo |
| UserAuthorityHistory | 회원사용권한변동이력 vo |
| UserLevel | 회원등급설정 vo |
| UserLevelHistory | 회원등급변동이력 vo |
| LibrarianLevel | 사서권한 vo |
| Calendar | 캘린더 vo |
| BookLend | 대출도서 vo |
| BookReport | 독후감 vo |
| Study | 서재 vo |
| StockBookTimes | 장서점검회차vo |
| Goal | 목표량 vo |
| Point | 포인트 vo |
| BookCarry | 수서 vo |
| BookStock | 소정도서 vo |
| Board | 게시판 vo |
| CallNumber | 청구기호 vo |
| BookRequest | 희망도서 vo |
| Paging | 페이징vo |
| ProgramManager | 프로그램 vo |
| ProgramApply | 프로그램 신청 vo |
| Facility | 시설 대실 vo |
| FacilityReservation | 시설 예약 vo |
| ksmart.pentagon.user | AdminController.java | 사서/관리자 관리 controller |
| AdminService.java | 사서/관리자 관리 service |
| AdminMapper.java | 사서/관리자 관리 interface |
| AdminMapper.xml | 사서/관리자 관리 mapper |
| LibraryController.java | 회원 관리 controller |
| LibraryService.java | 회원 관리 service |
| LibraryMapper.java | 회원 관리 interface |
| LibraryMapper.xml | 회원관리 mapper |
| ksmart.pentagon.point | PointController.java | 포인트 관리 controller |
| PointService.java | 포인트 관리 service |
| PointMapper.java | 포인트 관리 interface |
| PointMapper.xml | 포인트 관리 mapper |
| ksmart.pentagon.booklend | BookLendController.java | 대출/반납/예약 관리 controller |
| BookLendService.java | 대출/반납/예약 관리 service |
| BookLendMapper.java | 대출/반납/예약 관리 interface |
| BookLendMapper.xml | 대출/반납/예약 관리 mapper |
| ksmart.pentagon.bookcarry | BookCarryController.java | 수서 관리 controller |
| BookCarryService.java | 수서 관리 service |
| BookCarryMapper.java | 수서 관리 interface |
| BookCarryMapper.xml | 수서 관리 mapper |
| ksmart.pentagon.stock | BookStockController.java | 소장도서 관리 controller |
| BookStockService.java | 소장도서 관리 service |
| BookStockMapper.java | 소장도서 관리 interface |
| BookStockMapper.xml | 소장도서 관리 mapper |
| CallNumberController.java | 청구기호 생성 controller |
| CallNumberService.java | 청구기호 생성 service |
| CallNumberMapper.java | 청구기호 생성 interface |
| CallNumberMapper.xml | 청구기호 생성 mapper |
| MakeBsmarkService.java | 도서기호 생성 service |
| StaticCodeClass.java | 한글자모음 유니코드 array static class |
| ksmart.pentagon.stockcheck | StockBookCheckController.java | 장서점검 controller |
| StockBookCheckService.java | 장서점검 service |
| StockBookCheckMapper.java | 장서점검 interface |
| StockBookCheckMapper.xml | 장서점검 mapper |
| ksmart.pentagon.board | BoardController.java | 게시판 controller |
| BoardMapper.java | 게시판 service |
| BoardService.java | 게시판 interface |
| BoardMapper.xml | 게시판 mapper |
| ksmart.pentagon.program | ProgramController.java | 프로그램 관리 controller |
| ProgramService.java | 프로그램 관리 service |
| ProgramMapper.java | 프로그램 관리 interface |
| ProgramMapper.xml | 프로그램 관리 mapper |
| ksmart.pentagon.facility | FacilityController.java | 공공시설 관리 controller |
| FacilityService.java | 공공시설 관리 service |
| FacilityMapper.java | 공공시설 관리 interface |
| FacilityMapper.xml | 공공시설 관리 mapper |
| ksmart.pentagon.calendar | LibraryCalenderController.java | 관리자캘린더 controller |
| LibraryCalenderService.java | 관리자캘린더 service |
| LibraryCalenderMapper.java | 관리자캘린더 interface |
| LibraryCalenderMapper.xml | 관리자캘린더 mapper |
| BookCalenderController.java | 북캘린더 controller |
| BookCalenderService.java | 북캘린더 service |
| BookCalenderMapper.java | 북캘린더 interface |
| BookCalenderMapper.xml | 북캘린더 mapper |
| GoalController.java | 목표량 controller |
| GoalService.java | 목표량 service |
| GoalMapper.java | 목표량 interface |
| GoalMapper.xml | 목표량 mapper |
| ksmart.pentagon.mypage | BookReportController.java | 독후감 controller |
| BookReportService.java | 독후감 service |
| BookReportMapper.java | 독후감 interface |
| BookReportMapper.xml | 독후감 mapper |
| StudyController.java | 서재 controller |
| StudyService.java | 서재 service |
| StudyMapper.java | 서재 interface |
| StudyMapper.xml | 서재 mapper |
| ksmart.pentagon.print | PrintController.java | 출력 관리 controller |
| PrintService.java | 출력 관리 service |
| PrintMapper.java | 출력 관리 interface |
| PrintMapper.xml | 출력 관리 mapper |
| ksmart.pentagon.codeup | CodeUp.java | 코드증가 |
| ksmart.pentagon.libraryinfo | LibraryInfoController.java | 도서관 안내 controller |
| ksmart.pentagon.interceptor | AdminInterceptor.java | 관리자 페이지 interceptor |
| LibraryInterceptor.java | 도서관 페이지 interceptor |
| ksmart.pentagon.config | MybatisConfig.java | 마이바티스config |
| WebConfig.java | 인터셉터config |
| ksmart.pentagon.controller | LayoutController.java | 인트로,인덱스 controller |
| LayoutService.java | 인트로,인덱스 service |
| LayoutMapper.java | 인트로,인덱스 interface |
| LayoutMapper.xml | 인트로,인덱스 mapper |
| AgeOpenApi.java | 나이검색 api |
| AreaOpenApi.java | 지역검색 api |
| ksmart.pentagon.librarianbook | LibrarianBookController.java | 사서추천도서 controller |
| LibrarianBookMapper.java | 사서추천도서 service |
| LibrarianBookMapper.java | 사서추천도서 interface |
| LibrarianBookMapper.xml | 사서추천도서 mapper |

**5.개발환경**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **\*** | **구 분** | **개발 환경** | | **서비스 환경** |
| **S / W** | **OS** | Windows 7 | | Windows 7 |
| **웹브라우저** | Chrome | | Chrome 최적화 |
| **WAS** | apache-tomcat-9.0.27 | | apache-tomcat-9.0.X |
| **개발언어** | 서버 | Java 1.8.0\_231 | Servlet 3.1  JDK 1.8.X |
| 클라이언트 | java  JavaScript  HTML  CSS  jQuery |
| **Framework** | STS 4.2.1,  spring framework 4.2.1,  spring-web mvc 4.3.9,  Maven,  mybatis 3.5.3,  mybatis-spring-boot 2.1.1,  bootstrap | | 클라이언트 |
| **DBMS** | MySQL 5.5.32 | | MySQL 5.5.X UTF-8 |
| **Tools** | Spring Tool Suite 4.2.1  HeidiSQL 10.1.0  GitHub(https://github.com/shohye/Lifrary)  Maven | |  |

**6. 어플리케이션**

**6.1 어플리케이션 프로시저**

|  |
| --- |
| **기본어플리케이션프로시저** |
| EMB00000a6c0562 |

**6.2 서버 및 기타 설정**

|  |
| --- |
| **pom.xml** |
| <?xmlversion=*"1.0"*encoding=*"UTF-8"*?>  <projectxmlns=*"http://maven.apache.org/POM/4.0.0"*  xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*  xsi:schemaLocation=*"http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd"*>  <modelVersion>4.0.0</modelVersion>  <parent>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-parent</artifactId>  <version>2.2.1.RELEASE</version>  <relativePath/><!-- lookup parent from repository -->  </parent>  <groupId>ksmart.pentagon</groupId>  <artifactId>lifrary</artifactId>  <version>0.0.1-SNAPSHOT</version>  <packaging>war</packaging>  <name>lifrary</name>  <description>2team project</description>  <properties>  <java.version>1.8</java.version>  </properties>  <dependencies>  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-thymeleaf</artifactId>  </dependency>  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-web</artifactId>  </dependency>  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-devtools</artifactId>  <scope>runtime</scope>  <optional>true</optional>  </dependency>  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-tomcat</artifactId>  <scope>provided</scope>  </dependency>  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-test</artifactId>  <scope>test</scope>  <exclusions>  <exclusion>  <groupId>org.junit.vintage</groupId>  <artifactId>junit-vintage-engine</artifactId>  </exclusion>  </exclusions>  </dependency>  <!-- https://mvnrepository.com/artifact/nz.net.ultraq.thymeleaf/thymeleaf-layout-dialect -->  <dependency>  <groupId>nz.net.ultraq.thymeleaf</groupId>  <artifactId>thymeleaf-layout-dialect</artifactId>  <version>2.3.0</version>  </dependency>  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-jdbc</artifactId>  </dependency>  <!-- https://mvnrepository.com/artifact/org.mybatis.spring.boot/mybatis-spring-boot-starter -->  <dependency>  <groupId>org.mybatis.spring.boot</groupId>  <artifactId>mybatis-spring-boot-starter</artifactId>  <version>2.1.1</version>  </dependency>  <dependency>  <groupId>mysql</groupId>  <artifactId>mysql-connector-java</artifactId>  <scope>runtime</scope>  </dependency>  <!-- https://mvnrepository.com/artifact/org.bgee.log4jdbc-log4j2/log4jdbc-log4j2-jdbc4.1 -->  <dependency>  <groupId>org.bgee.log4jdbc-log4j2</groupId>  <artifactId>log4jdbc-log4j2-jdbc4.1</artifactId>  <version>1.16</version>  </dependency>  <!-- https://mvnrepository.com/artifact/org.jsoup/jsoup -->  <dependency>  <groupId>org.jsoup</groupId>  <artifactId>jsoup</artifactId>  <version>1.12.1</version>  </dependency>  </dependencies>  <build>  <plugins>  <plugin>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-maven-plugin</artifactId>  </plugin>  </plugins>  <resources>  <resource>  <filtering>false</filtering>  <directory>src/main/java</directory>  <includes>  <include>\*\*/\*.xml</include>  </includes>  </resource>  <resource>  <directory>src/main/resources</directory>  </resource>  </resources>  </build>  </project> |

|  |
| --- |
| **application.properties** |
| server.port=80  spring.thymeleaf.cache=false  resources.notload.list = /adminStatic/\*\*, /libraryStatic/\*\*, /admin/login, /lifrary/introSearchList  spring.datasource.driver-class-name=net.sf.log4jdbc.sql.jdbcapi.DriverSpy  spring.datasource.url = jdbc:log4jdbc:mysql://shohye.cafe24.com/shohye?serverTimezone=UTC&characterEncoding=UTF8  #spring.datasource.url = jdbc:log4jdbc:mysql://localhost:3306/shohye?serverTimezone=UTC&characterEncoding=UTF8  spring.datasource.username=아이디  spring.datasource.password=비밀번호  mybatis.mapper-locations=classpath:mapper/\*.xml  mybatis.type-aliases-package=ksmart.pentagon  logging.config=classpath:logback-spring.xml |

**6.3** [**주요 기능별 정리**](#주요기능별정리)

**6.3.1 도서관 사이트**

**(1)인트로**

|  |
| --- |
|  |

|  |
| --- |
| 도서 검색 페이지 |
| 도서관 선택 후 도서 검색이 가능하다.  하단의 아이콘을 클릭하면 해당 페이지로 이동한다. (펜타콘도서관 , 스퀘어 도서관 , 관리자 페이지) |

**(2) 로그인**

|  |
| --- |
|  |

■SQL : 로그인 처리시 입력한 아이디의 정보를 가져오는 쿼리문 실행

|  |
| --- |
| <selectid=*"loginCheck"*parameterType=*"String"*resultType=*"ksmart.pentagon.vo.User"*>  SELECT  u\_id as uId,  u\_pw as uPw,  u\_name as uName,  u\_division as uDivision  FROM  user  WHERE u\_id = #{uId} and l\_code = #{libNum}  </select> |

|  |
| --- |
| 로그인 페이지 |
| 접속한 도서관 페이지의 도서관 코드와 회원 코드가 일치하는 경우,  해당 도서관의 회원이므로 로그인이 된다.  회원가입 버튼 클릭시 회원가입 페이지로 이동하게 된다.  로그인시 인덱스 페이지로 이동한다. |

**(3) 회원가입 페이지**

|  |
| --- |
|  |

■controller :회원가입을 하기위해 4개의vo를 호출한다.

|  |
| --- |
| //회원가입등록  //4개테이블동시등록  @PostMapping("/lifrary/userInsert")  public String userInsert(HttpSessionsession, User user, UserLevelHistoryuserLevelHistory  , UserAuthorityHistoryuserAuthorityHistory, StudyCatestudyCate) {  System.*out*.println("userInsert @PostMapping회원가입폼 ");  System.*out*.println("user 확인 ==>> " + user);  System.*out*.println("userLevelHistory확인 ==>> " + userLevelHistory);  System.*out*.println("userAuthorityHistory확인 ==>> " + userAuthorityHistory);  System.*out*.println("studyCate확인 ==>> " + studyCate);      libraryService.userInsertUser(user);  libraryService.userInsertUserLevelHistory(userLevelHistory);  libraryService.userInsertUserAuthorityHistory(userAuthorityHistory);  libraryService.userInsertStudyCate(studyCate);    return"redirect:/";  } |

|  |
| --- |
| 도서관 회원가입 페이지 |
| 모든 정보 입력 후 회원가입 버튼을 누르면 회원가입 됩니다 |

**(4) 도서관 ID/PW 찾기 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 도서관 ID/PW 찾기 페이지 |
| 아이디 찾기 : 이름과 이메일, 핸드폰번호 확인 후 맞으면 아이디가 찾아집니다.  비밀번호 찾기 :ID , 이름, 핸드폰 번호, 이메일 확인 후 맞으면 비밀번호가 나타납니다. |

|  |
| --- |
|  |

**(5) 도서관 메인**

|  |
| --- |
| 도서관 메인 페이지 |
| 슬라이드 세션을 통해 인긴도서/사서 추천도서/신착 도서를 확인 가능하다.  도서관/자료실 휴관일과 일정을 한 눈에 확인 가능하다.  최근 등록된 3개의 프로그램&행사를 볼 수 있다.자세히보기 버튼을 클릭하여, 해당 프로그램의 상세보기 페이지로 이동할 수 있다. |

**(7) 도서검색**

|  |
| --- |
| 도서 검색 페이지 |
| 대분류( 단행본, 연소간행물, 비도서) 와 도서명으로 검색이 가능하다 |

**(8) 도서상세검색**

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
| 도서 상세 검색 페이지 |
| 대분류( 단행본, 연소간행물, 비도서) 와 도서명, ISBN, 출판사, 저자명, 키워드, 발행년도로 검색이 가능하다 |

**(9) 도서상세결과 리스트**

|  |
| --- |
|  |

■책 상태가 정상인 모든 소장도서 중 상세 검색 조건에 맞는 도서 리스트 가져오기( 페이징 포함 )

|  |
| --- |
| <selectid=*"getDetailSearchStockList"*resultMap=*"bookStockResultMap"*  parameterType=*"hashMap"*>  SELECT  bs.bs\_code  , bs.u\_id  , bi.bi\_img  , bi.bi\_name  , bi.bi\_isbn  , bi.bi\_year  , bi.bi\_author  , bi.bi\_publisher  , bi.bi\_kdc  , bi.bi\_detail  , bcl.bcl\_name  , bcm.bcm\_name  , bs\_alias\_mark  , bs\_writer\_mark  , bs\_secondary\_mark  , bs.bs\_total\_page  , bs.bs\_book\_state  , bs.bs\_lend\_state  , bs.bs\_carry\_route  , bs.bs\_date  From book\_stockbs  left JOIN book\_informationbi ON bs.bi\_isbn = bi.bi\_isbn  left JOIN book\_cate\_largebcl ON bs.bcl\_code = bcl.bcl\_code  left JOIN book\_cate\_mediumbcm ON bs.bcm\_code = bcm.bcm\_code  <trimprefix=*"where"*prefixOverrides=*"AND|OR"*>  <iftest=*"bclCode != null and bclCodeneq ''.toString()"*>  AND bs.bcl\_code=#{bclCode}  </if>    <iftest=*"lCode != null and lCodeneq ''.toString()"*>  AND bs.l\_code=#{lCode}  </if>  <iftest=*"biName != null and biNameneq ''.toString()"*>  AND bi.bi\_name Like CONCAT ('%',#{biName},'%')  </if>    <iftest=*"biPublisher != null and biPublisherneq ''.toString()"*>  AND bi.bi\_publisher Like CONCAT ('%',#{biPublisher},'%')  </if>    <iftest=*"biIsbn != null and biIsbnneq ''.toString()"*>  AND bi.bi\_isbn=#{biIsbn}  </if>    <iftest=*"biAuthor != null and biAuthorneq ''.toString()"*>  AND bi.bi\_author Like CONCAT ('%',#{biAuthor},'%')  </if>    <iftest=*"biDtail != null and biDtailneq ''.toString()"*>  AND bi.bi\_detail Like CONCAT ('%',#{biDtail},'%')  </if>    <iftest=*"biYearStartneq null and biYearStartneq ''*  *and biYearEndneq null and biYearEndneq ''.toString()"*>  AND <![CDATA[  bi.bi\_year>= CAST( #{biYearStart} AS DECIMAL)  and bi.bi\_year<= CAST( #{biYearEnd} AS DECIMAL)  ]]>  </if>  <iftest=*"startRowneq null and rowPerPageneq null"*>  ORDER BY bi.bi\_name  LIMIT #{startRow},#{rowPerPage}  </if>  </trim>  </select> |

|  |
| --- |
| 도서 상세 검색 결과 리스트 |
| 검색 결과를 리스트 형태로 보여준다 |

**(10) 도서상세결과 리스트 (그리드ver)**

|  |
| --- |
|  |

|  |
| --- |
| 도서 상세 검색 결과 리스트(그리드ver) |
| 검색 결과를 그리드 리스트 형태로 보여준다 |

**(11) 도서 상세 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 도서 상세 페이지 |
| 리스트에서 선택한 도서의 상세 정보와 대출가능 상태를 보여준다.  알라딘의 도서미리보기 페이지와 연결하여 도서를 미리 볼 수 있다  당일도서예약 기능으로 당일 대출할 도서를 미리 예약할 수 있다( 당일 대출을 안하는 경우 예약은 취소된다) |

**(12) 프로그램 리스트**

|  |
| --- |
|  |

■SQL : 키워드 혹은 옵션으로 검색하여 프로그램은 select하는 쿼리 실행

|  |
| --- |
| <!-- 프로그램검색결과리스트츨력 -->  <selectid=*"getSearchProgramList"*parameterType=*"String"*resultMap=*"PMResultMap"*>  SELECT  \*  FROM  program\_mgr  WHERE  l\_code = #{libNum}  <iftest=*"keywords == null and target == '전체'"*>  and pm\_name LIKE CONCAT('%',#{keywords},'%')  </if>  <iftest=*"keywords == null and target != '전체'"*>  and pm\_target = #{target}  and pm\_name LIKE CONCAT('%',#{keywords},'%')  </if>  <iftest=*"keywords != null and target != '전체'"*>  and pm\_target = #{target}  </if>  </select> |

|  |
| --- |
| 등록된 프로그램 리스트 페이지 |
| 키워드와 옵션으로 검색이 가능하다.  좌측 메뉴에서 최근 등록된 프로그램을 확인할 수 있다.  사서가 등록한 프로그램 및 행사 리스트를 볼 수 있다. |

**(13) 프로그램 상세보기**

|  |
| --- |
|  |

|  |
| --- |
| 등록 프로그램 상세보기 페이지 |
| 사서가 등록한 프로그램 및 행사의 상세 정보를 확인할 수 있다.  신청 가능 인원이 max에 도달하면 신청을 할 수 없다.  로그인을 해야 하며, 회원만 신청하기가 가능하다. |

**(14) 프로그램 신청하기**

|  |
| --- |
|  |

■SQL : 프로그램 상세보기를 위해 pmCode / paCode로 DB에서 select하는 쿼리 실행

|  |
| --- |
| <!-- 도서관 / 프로그램신청하기 -->  <insertid=*"applyProgram"*parameterType=*"ksmart.pentagon.vo.ProgramApply"*>  <selectKeykeyProperty=*"paCode"*resultType=*"String"*order=*"BEFORE"*>  SELECT  CONCAT('pa-',DATE\_FORMAT(CURDATE(), '%y%m%d'), (IF(DATE\_FORMAT(CURDATE(),'%y%m%d') > LEFT(SUBSTRING\_INDEX(MAX(pa.pa\_code), '-', -1),6), '00001',LPAD(RIGHT(MAX(pa.pa\_code),5)+1,5,'0'))))  FROM  program\_apply AS pa  </selectKey>  INSERT INTO  program\_apply  (pa\_code, l\_code, pm\_code, u\_id, pa\_participants, pa\_gender, pa\_birth, pa\_addr, pa\_tel, pa\_date)  VALUES (#{paCode}, #{lCode}, #{pmCode}, #{uId}, #{paParticipants}, #{paGender}, #{paBirth}, #{paAddr}, #{paTel}, NOW())  </insert> |

|  |
| --- |
| 프로그램 신청 페이지 |
| 키워드와 옵션으로 검색이 가능하다.  좌측 메뉴에서 최근 등록된 프로그램을 확인할 수 있다.  사서가 등록한 프로그램 및 행사 리스트를 볼 수 있다. |

**(15) 독후감**

|  |
| --- |
|  |

|  |
| --- |
| 독후감 페이지 |
| 독후감 리스트를 확인하고 검색할 수 있다. 제목을 클릭하면 독후감 상세보기 페이지로 이동할 수 있다.작성하기 버튼을 클릭하면 독후감 작성 페이지로 이동할 수 있다. |

**(16) 독후감 상세**

|  |
| --- |
|  |
| [그림 1]  작성자인 경우 우측에 수정하기 버튼이 표시되며, 클릭 시 독후감 수정 페이지로 이동한다.    [그림 2]  댓글 작성자인 경우 우측에 수정, 삭제 버튼이 표시된다. 새 댓글 등록 시 공개여부를 설정할 수 있다. |

|  |
| --- |
| 독후감 상세 페이지 |
| 독후감의 상세내용을 확인할 수 있다. 수정하기를 클릭하여 독후감 수정페이지로 이동할 수 있다. 댓글을 등록, 수정, 삭제할 수 있다. |

**(17) 독후감 작성**

|  |
| --- |
|  |

|  |
| --- |
| 독후감 작성 페이지 |
| 도서정보를 검색하여 도서정보를 표시하고 독후감을 등록할 수 있다. 에디터를 사용하여 원하는 대로 내용을 입력할 수 있다. 나만의 별점과 공개여부를 설정할 수 있다. |

**(18) 희망도서 신청 안내 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 희망도서 신청 안내 페이지 |
| 희망도서 신청 안내내용을 확인 할 수 있으며. 회원인 경우만 신청,신청확인이 가능하다. |

**(19) 희망도서 신청 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 희망도서 신청 페이지 |
| 권한이 회원인 경우만 접근이 가능하며 도서관에 반입을 희망하는 도서 상세정보를 작성한 후, 신청이 가능하다. |

**(20)도서 기부신청 안내페이지**

|  |
| --- |
| C:\Users\Administrator\Desktop\완료보고서_캡쳐\lifrary_기부 신청 안내.png |

|  |
| --- |
| 도서 기부신청 안내페이지 |
| 도서 기부신청 안내를 보여주며, 기부신청서의 다운로드가 가능하다. |

**(21) 공공시설 신청 페이지 / 열람실, 사물함**

|  |  |
| --- | --- |
| [그림 1]  자리 배치도에서 자리를 선택합니다. | |
| [그림 2]  자리를 누르게 되면 confirm창이 뜨게됩니다. |
| [그림 3]  처음 선택한 자리 외에 다른곳을 선택하게 되면 경고창이 뜨게 됩니다. |

■javaScript :시설 리스트중 하나를 선택하면 aJax를 사용하여 동적으로 있는 가로 세로값으로 테이블을 동적으로 생성해준다. 선택한 자리의 색을 노란색으로 바꿔준다. 사용하고 있는 자리는 주황색으로 표시된다. 현재 시설을 신청하여 사용 중 이거나, 하루 1회 이상 신청 한경우 신청을 못하도록 막는다.

|  |
| --- |
| <scriptth:inline=*"javascript"*>  $(document).ready(function(){    varselectNum =''; // 누른자리저장 / 비교에사용  varselectSeat = true; // 자리이눌러져있으면 true  varselectFcode =''; // 누른테이블의공공시설코드  varselectfKinds = ''; // 누른테이블의공공시설종류  varuId = [[${session.SID}]]  varlCode = [[${session.LIBNUM}]]      /\* 예약기능. 당일예약이있으면예약불가., \*/  $(document).on('click','.tdClass',function(){ // 자리선택시이벤트  varfCode = $('#fCode')  varfrSelectNum = $('#frSelectNum')  varfKinds = $('#fKinds')    if($(this).attr("status") == 'true'&&selectSeat == true){ // 자리선택시해당자리의 status가 true인경우 (예약가능인경우)  selectNum = $(this).attr("seat"); // 누른것저장. (하나만선택가능하게할것)  varconfirmd = confirm($(this).text()+'번을선택하셨습니다. \n선택한자리를예약하시겠습니까?')  if(confirmd){ // 확인버튼누르면  $(this).css("background-color",'yellow') // 선택된자리의색이변경  $(this).css("color",'black') // 선택된자리의글자색이변경  $(this).attr("status",'false'); //status를 false로변경  selectSeat = false;    fCode.val(selectFcode) //선택된것이어떤시설의코드인지.  frSelectNum.val(selectNum) //선택된자리의번호  fKinds.val(selectfKinds)  }  } else if($(this).attr("status") == 'true'&&selectSeat == false){  alert('자리가이미선택되어있습니다. \n취소하려면선택된좌석을클릭해주세요.')  } else{ // 자리선택시해당자리의 status가 false인경우  if($(this).attr("seat") == selectNum){  varconfirmd = confirm('이미선택한자리입니다. \n취소하시겠습니까?')  if(confirmd){ // 확인버튼누르면  $(this).css("background-color",'black') // 선택된자리의색이변경  $(this).css("color",'white') // 선택된자리의글자색이변경  $(this).attr("status",'true'); // status를 false로변경  selectSeat = true;  fCode.val('') //선택된것이어떤시설의코드인지.  frSelectNum.val('') //선택된자리의번호  fKinds.val('') // 시설종류  }  }  }  })      $(document).on('click', '.fTable', function(){ // 공공시설목록을눌렀을때.  varfCode = $(this).find('input[name=fCode]').val();  varfKinds = $(this).find('input[name=fKinds]').val();  varfSeatOrImg = $(this).find('input[name=fSeatOrImg]').val();  varfName = $(this).find('input[name=fName]').val();  vartdTable = $('#tdTable')  var contents = '';  selectFcode = fCode; // 어떤공공시설눌렀는지fCode를저장.  selectfKinds = fKinds;  console.log('열람실 /사물함임')  varfRow = $(this).find('input[name=fRow]').val();  varfCol = $(this).find('input[name=fCol]').val();  varreserveArray = new Array(); // ajax로가져온결과를담는배열. 예약된자리를담는배열    /\* 테이블그려주기 \*/  varseatArray = fSeatOrImg.split(',');  selectSeat = true; // 선택초기화    contents += '<br><h3>자리배치도&emsp;&emsp;'+fName+'</h3><br>'  contents += '<table class="makeTbLine" border="0px;" style="display: inline-block;">';  for(vari = 1; i<= fCol; i++){  contents += '<tr>';  for(var j = 1; j <= fRow; j++){  contents += '<td class="tdClass" style="color:black; text-align: center; font-size:12px; width: 40px; height: 35px;"'+  'seat="'+i+'-'+j+'" status="true">'+i+'-'+j+'</td>';  }  contents += '</tr>';  }  contents += '</table>';  contents += '<br><br><h3>신청하기</h3><br><br>'  contents += '<form id="reserveForm" action="/lifrary/reservation" method="post">'  contents += '<table class="reserveForm" border="1";>'  contents += '<input id="uId" type="hidden" name = "uId">'  contents += '<input id="lCode" type="hidden" name = "lCode">'  contents += '<input id="fCode" type="hidden" name = "fCode">'  contents += '<input id="fKinds" type="hidden" name = "fKinds">'  contents += '<input id="frSelectNum" type="hidden" name = "frSelectNum" readonly="readonly">'  contents += '<tr><td>연락처</td><td><input id="frTel" type="text" name = "frTel"></td></tr>'    contents += '<tr><td>신청하기</td><td><button id="reserveBtn" type="button">신청</button></td></tr></table>'  contents += '</form>'    tdTable.html(contents)  vartdClass = $('.tdClass');    $('#uId').val(uId) //세션에저장된아이디값넣기  $('#lCode').val(lCode) //세션에저장된도서관코드넣기  /\* DB에서해당시설코드에대한예약자를문자열(배열로만들수있게)가져온다. \*/  var request = $.ajax({  url: "/lifrary/reservationAjax",  method: "POST",  data: { 'fCode' : fCode },  dataType: "json"  });    request.done(function(data) {  reserveArray = data.seatNum; // 결과물을reserveArray에대입한다.  /\* 뿌려준테이블에가져온배열(선택된자리)색칠해주기 \*/  for(vari = 0; i<tdClass.length; i++){  if(seatArray.indexOf(tdClass.eq(i).attr('seat')) >= 0){ // 배열에서tdClass의 seat값과같은게있는가.  tdClass.eq(i).text(seatArray.indexOf(tdClass.eq(i).attr('seat'))+1); // seat배열과같은 index로 text변경    if(reserveArray.indexOf(tdClass.eq(i).attr('seat'))+1 > 0){ // ajax로가져와담은배열에테이블의 seat속성과같은게있으면, true  tdClass.eq(i).css("background-color",'orange');  tdClass.eq(i).attr("status",'false');  }else { tdClass.eq(i).css("background-color",'black'); tdClass.eq(i).css("color",'white') tdClass.eq(i).attr("status",'true');  }    }else{ tdClass.eq(i).css("background-color",'white') // seat값과다르면글자색을흰색으로변경 tdClass.eq(i).text(''); // 글자없애기 tdClass.eq(i).attr("status",'false') // 속성을 false로바꿔누를수도없게만든다.  }  }  });    request.fail(function( jqXHR, textStatus ) {  alert( "Request failed: " + textStatus );  });  }) |

■SQL : 시설이 예약되어있는지 확인 후 신청되어있는 자리는 색을 변경해 주기 위해 쿼리를 실행

|  |
| --- |
| <!--예약일자<= 현재날짜<= 예약종료일자현재날짜가사이에있는좌석만출력.-->  <selectid=*"getReservationSeat"*parameterType=*"String"*resultMap=*"reserveResultMap"*>  SELECT  fr\_select\_num  FROM  facility\_reservation  WHERE  f\_code = #{fCode}  <![CDATA[ AND fr\_reserve\_date<= NOW() AND fr\_end\_date>= NOW() ]]>  </select> |

■SQL : 시설을 예약하는 쿼리 실행

|  |
| --- |
| <!--공공시설예약쿼리 -->  <insertid=*"reserveFacility"*parameterType=*"ksmart.pentagon.vo.FacilityReservation"*>  <selectKeykeyProperty=*"frCode"*resultType=*"String"*order=*"BEFORE"*>  SELECT  CONCAT('fr-',DATE\_FORMAT(CURDATE(), '%y%m%d'),  (IF(DATE\_FORMAT(CURDATE(),'%y%m%d') > LEFT(SUBSTRING\_INDEX(MAX(fr.fr\_code), '-', -1),6)  , '00001'  ,LPAD(RIGHT(MAX(fr.fr\_code),5)+1,5,'0'))))  FROM  facility\_reservation AS fr  </selectKey>  INSERT INTO facility\_reservation  (fr\_code, l\_code, u\_id, f\_code, fr\_select\_num, fr\_tel, fr\_reserve\_date, fr\_end\_date)    VALUES (#{frCode}, #{lCode}, #{uId}, #{fCode}, #{frSelectNum}, #{frTel}, NOW(),  <iftest=*'fKinds.equals("열람실")'*>  DATE\_ADD(NOW(), INTERVAL 3 HOUR)  </if>  <iftest=*'fKinds.equals("스터디룸")'*>  DATE\_ADD(NOW(), INTERVAL 5 HOUR)  </if>  <iftest=*'fKinds.equals("강연실")'*>  DATE\_ADD(NOW(), INTERVAL 1 DAY)  </if>  <iftest=*'fKinds.equals("사물함")'*>  DATE\_ADD(NOW(), INTERVAL 1 MONTH)  </if>  )  </insert> |

■SQL : 접속한 아이디로 당일 예약을 했는지 체크하는 쿼리문 실행

|  |
| --- |
| <!--당일예약을 1회라도했는지찾는쿼리문. 1회라도있으면당일예약은더이상불가능. -->  <selectid=*"getRevserVation"*resultMap=*"reserveResultMap"*>  SELECT  fr.u\_id as uId  FROM  facility\_reservation AS fr INNER JOIN facility AS f  ON fr.f\_code = f.f\_code  WHERE  fr.u\_id = #{uId} AND  fr.l\_code = #{libNum} AND  f.f\_kinds = #{fKinds} AND  DATE\_FORMAT(fr.fr\_reserve\_date, "%Y-%m-%d") = CURDATE()  </select> |

|  |
| --- |
| 공공시설 신청 페이지 (열람실 / 사물함) |
| 자리를 선택 후 신청할 수 있다.  신청시 당일 1회이상 예약했는지 DB조회 후 기록이 있다면 예약을 못하도록 한다.  자리 선택시 노란색으로 변경되며, 사용중인 자리는 주황색으로 표시된다.  자리가 선택된 상태에서 다른 자리를 선택하게 되면, 이미 자리가 선택되었다는 경고창이 뜨게 된다.  예약 시간~종료 시간 사이에 있는것만 사용 처리를 하며, 시간이 지나면 자동으로 사용 종료처리된다. |

**(22) 공공시설 신청 페이지 / 스터디룸, 강연실**

|  |
| --- |
|  |

|  |
| --- |
| <scriptth:inline=*"javascript"*>  $(document).ready(**function**(){  **var** selectFcode =''; // 누른테이블의공공시설코드  **var** selectfKinds = ''; // 누른테이블의공공시설종류  **var** uId = [[${session.SID}]]  **var** lCode = [[${session.LIBNUM}]]  $(document).on('click', '.fTable', **function**(){ // 공공시설목록을눌렀을때.  **var** fCode = $(**this**).find('input[name=fCode]').val();  **var** fKinds = $(**this**).find('input[name=fKinds]').val();  **var** fSeatOrImg = $(**this**).find('input[name=fSeatOrImg]').val();  **var** fName = $(**this**).find('input[name=fName]').val();  **var** tdTable = $('#tdTable')  **var** contents = '';  selectFcode = fCode; // 어떤공공시설눌렀는지 fCode를저장.  selectfKinds = fKinds;  contents += '<br><h3>자리배치도&emsp;&emsp;'+fName+'</h3><br>'  //강연실 / 스터디룸이미지보여주기  contents += '<img alt="이미지를표시할수없습니다." src="/adminStatic/img/avatars/studyroom.jpg"><br>'    contents += '<br><br><h3>신청하기</h3><br><br>'  contents += '<form id="reserveForm" action="/lifrary/reservation" method="post">'  contents += '<table class="reserveForm" border="1";>'  contents += '<input id="uId" type="hidden" name = "uId">'  contents += '<input id="lCode" type="hidden" name = "lCode">'  contents += '<input id="fCode" type="hidden" name = "fCode">'  contents += '<input id="fKinds" type="hidden" name = "fKinds">'  contents += '<tr><td>연락처</td><td><input id="frTel" type="text" name = "frTel"></td></tr>'  contents += '<tr><td>신청하기</td><td><button id="reserveBtn" type="button">신청</button></td></tr></table>'  contents += '</form>'  tdTable.html(contents)    $('#uId').val(uId) //세션에저장된아이디값넣기  $('#lCode').val(lCode) //세션에저장된도서관코드넣기  $('#fCode').val(fCode)  $('#fKinds').val(fKinds)  })  })  </script> |

■javaScript : 스터디룸, 강연실에 대한 이미지 출력, 신청처리

|  |
| --- |
| 공공시설 신청 페이지 (스터디룸 /강연실) |
| 시설 클릭시 이미지와 신청 폼을 동적으로 생성해준다. |

**(23) 도서관 소개 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 도서관 소개 페이지 |
| 도서관의 기본정보와, 인사말, 도서관 서비스 헌장, 이용자 협조사항의 내용을 보여준다. |

**(24) 도서관 내 정보 상세보기 페이지**

|  |
| --- |
|  |

■LibraryMapper.xml : 3개 테이블 조인하기

|  |
| --- |
| <selectid=*"myUserDetail"*resultMap=*"userResultMap"*parameterType=*"String"*>  SELECT  u.u\_id  ,u.l\_code  ,u.u\_number  ,u.u\_division  ,u.u\_pw  ,u.u\_name  ,u.u\_birth  ,u.u\_addr  ,u.u\_tel  ,u.u\_email  ,u.u\_email\_agree  ,u.u\_date  ,u.u\_appointment  ,uas.uas\_name  ,ul.ul\_name  FROM  user AS u  INNER JOIN  user\_authority\_set AS uas  ON  u.uas\_code = uas.uas\_code  INNER JOIN  user\_level AS ul  ON  u.ul\_level = ul.ul\_level  where  u.u\_id = #{getSID}  AND u.l\_code = #{libNum}  </select> |

|  |
| --- |
| 도서관 내 정보 상세보기 페이지 |
| 가입한 내정보를 상세하게 볼 수 있는 페이지 |

**(25) 도서관 내 정보 수정 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 도서관 내 정보 수정 페이지 |
| 가입한 내 정보, 비밀번호 확인 후 수정하는 페이지 |

**(26) 도서관 회원탈퇴 페이지**

|  |
| --- |
|  |

■controller :

|  |
| --- |
|  |

|  |
| --- |
| 도서관 회원탈퇴 페이지 |
| 수정페이지에서 ‘회원탈퇴 바로가기’ 누르면 비밀번호 확인 후 회원탈퇴 하는 페이지 |

**(27)내 포인트**

|  |
| --- |
|  |
| [그림 1]  포인트 사용 버튼 클릭 시 포인트 사용 Modal이 나타난다.선택한 포인트의 차감 포인트를 표시하고,사용 후 남은 포인트를 계산하여 총 포인트에 출력한다. |

|  |
| --- |
| 내 포인트 페이지 |
| 내 포인트 리스트와 총 포인트를 확인할 수 있다.포인트를 사용할 수 있다. |

**(28) 도서 대출 현황**

|  |
| --- |
|  |
| [그림 1]  대출 도서는 연장하기 버튼을 클릭하여 1회 반납예정일(7일 연장)을 연장할 수 있다. 1회 연장된 도서와 반납된 도서는 연장불가가 표시된다. |

|  |
| --- |
| 대출 도서 현황 페이지 |
| 내 대출 도서 리스트를 확인하고 검색할 수 있다. 대출 도서의 경우 1회 반납예정일을 연장할 수 있다. |

**(29) 도서 예약 현황**

|  |
| --- |
|  |
| [그림 1]  예약도서는 예약취소 버튼을 클릭하여 예약을 취소할 수 있다.  취소 완료된 예약도서는 취소완료가 표시된다. |

|  |
| --- |
| 도서 예약 현황 페이지 |
| 예약 도서 리스트를 확인하고 검색할 수 있다. 예약도서를 취소할 수 있다. |

**(30) 북 캘린더**

|  |
| --- |
|  |
| [그림 1]  목표량 등록일로부터 반납된 도서 권수를 확인하여 목표량 진행률을 보여준다.    [그림 2]  목표량 설정을 클릭하면 목표량 등록 Modal이 나타난다. 목표 기간을 선택하면 자동으로 종료일이 표시된다.    [그림 3]  캘린더의 날짜를 클릭하면 일정등록 Modal이 나타난다. 도서명을 입력하고 도서검색 버튼을 누르면 검색결과가 리스트로 출력된다.    [그림 4]  검색한 도서 정보가 표시된다. 제목, 내용, 읽은 페이지를 입력할 수 있다. 읽은 페이지를 입력하면 자동으로 진행 상황이 표시된다. 선택일은 변경할 수 있다.    [그림 5]  캘린더 날짜 칸에 표시되는 도서 이미지를 클릭하면 일정 수정 Modal이 나타난다. 제목, 내용, 읽은 페이지를 수정할 수 있다. 삭제 버튼을 클릭하면 일정을 삭제할 수 있다. |

■ 목표량, 캘린더 Javascript

|  |
| --- |
| <scripttype=*"text/javascript"*>  $(document).ready(**function**() {  **var** today =moment().format('YYYY-MM-DD');//오늘 날짜  **var** month =parseInt(moment().format('M'));//이번 달  **var** year =moment().format('YYYY');//이번 년도  **var** gEndDate = **null**; //분기별 마지막일    $('.goal').eq(0).find('label').text(year+'년 목표량');  $('.goal').eq(2).find('label').text(month+'월 목표량');  **if**(month < 4){//1분기  gEndDate = year+'-03-31';  $('.goal').eq(1).find('label').text('1분기 목표량');  }**elseif**(3 < month && month < 7){//2분기  gEndDate = year+'-06-30';  $('.goal').eq(1).find('label').text('2분기 목표량');  }**elseif**(6 < month && month < 10){//3분기  gEndDate = year+'-09-30';  $('.goal').eq(1).find('label').text('3분기 목표량');  }**elseif**(9 < month){//4분기  gEndDate = year+'-12-31';  $('.goal').eq(1).find('label').text('4분기 목표량');  }  //목표량 출력  **var** request = $.ajax({  url: "/lifrary/getGoalList",  data: { gEndDate : gEndDate },  method: "POST",  dataType: "json"  });  request.done(**function**(data) {  **for**(**var** i = 0; i<data.length; i++){  goal = data[i];  **if**(goal.gDivision == 'month'){  $('.goal').eq(2).find('strong').text('('+goal.lendCount+'/'+goal.gNum+')');  $('.progress-bar').eq(2).attr('aria-valuenow',goal.percent).css('width',goal.percent+'%').text(goal.percent+'%');  }**elseif**(goal.gDivision == 'quarterly'){  $('.goal').eq(1).find('strong').text('('+goal.lendCount+'/'+goal.gNum+')');  $('.progress-bar').eq(1).attr('aria-valuenow',goal.percent).css('width',goal.percent+'%').text(goal.percent+'%');  }**elseif**(goal.gDivision == 'year'){  $('.goal').eq(0).find('strong').text('('+goal.lendCount+'/'+goal.gNum+')');  $('.progress-bar').eq(0).attr('aria-valuenow',goal.percent).css('width',goal.percent+'%').text(goal.percent+'%');  }  }  });//request.done  request.fail(**function**( jqXHR, textStatus ) {  alert('목표량을 불러오는데 실패하였습니다');  });//request.fail  //기간에 따른 종료일 구하기  $('#goalInsert li').click(**function**(){  **var** endDay = **null**;//종료일  //선택한 옵션 값  **var** optValue = $(**this**).attr('rel');  **if**(optValue == 'month'){//월별인 경우 종료일에 이번달 마지막일 셋팅  endDay = moment().endOf('month').format('YYYY-MM-DD');  }**elseif**(optValue == 'quarterly'){//분기별인 경우 분기에 따라 종료일 셋팅  endDay = gEndDate;  }**elseif**(optValue == 'year'){//년도별인 경우 올 해 마지막일 셋팅  endDay = year+'-12-31';  }  $('[name="gStartDate"]').val(today);  $('[name="gEndDate"]').val(endDay)  });  **var** bookInfoList = []; //도서 검색 리스트 담을 변수  **var** totalPage = 0; //전체페이지  //캘린더 생성  **var** calendar = **new** FullCalendar.Calendar($('#bookCalendar')[0], {  plugins: [ 'dayGrid','interaction'],  header: {  left: 'dayGridMonth',  center: 'title'  },  eventLimit: **true**,  fixedWeekCount: **false**,  contentHeight: 720,  navLinks: **true**,  locale: 'ko',  events: **function**(info, successCallback, failureCallback ){  //ajax로 DB에서 data가져오기  $.ajax({  url: '/lifrary/getMyCalenderList',  dataType: 'json',  success: **function**(data) {  **var** events = [];  //반복문  $.each(data, **function**(index, list) {  events.push({  id: list.cCode,  start: list.cStartDate,  imageurl: list.biImg  });  });  successCallback(events);  } //success  });//ajax  },  eventRender: **function**(info) {//날짜에 이미지 넣기  **if** (info.event.extendedProps.imageurl) {  info.el.firstChild.innerHTML = info.el.firstChild.innerHTML + '<img src="' + info.event.extendedProps.imageurl +'" style="height: 100px; width=100%;">';  }  },  //날짜클릭시 일정등록 Modal 띄우기  navLinkDayClick: **function**(date, jsEvent){  **var** selectDate =moment(date).format('YYYY-MM-DD');  $('#bookInsertModal').modal('show');  $('#selectDate').text(selectDate);  $('#insertForm [name="cStartDate"]').val(selectDate);  },  //날짜 칸 도서 이미지 클릭 시 일정수정 Modal 띄우기  eventClick: **function**(info) {  **var** cCode = info.event.id;  $('#bookUpdateModal').modal('show');  //일정 정보 가져오는 ajax  **var** request = $.ajax({  url: "/lifrary/getMyCalender",  data: { cCode : cCode },  method: "POST",  dataType: "json"  });  request.done(**function**(data) {  **var** bookInfo = data.bookInformation;  totalPage = bookInfo.bsTotalPage;//총페이지  $('#updateForm strong').eq(1).text(bookInfo.biName);//도서명  $('#updateForm strong').eq(3).text(bookInfo.biIsbn);//ISBN  $('#updateForm strong').eq(5).text(bookInfo.biAuthor);//저자명  $('#updateForm strong').eq(7).text(bookInfo.biPublisher);//발행자  $('#updateForm [name="cCode"]').val(data.cCode);//읽은 페이지  $('#updateForm [name="cTitle"]').val(data.cTitle);//읽은 페이지  $('#updateForm [name="cContent"]').val(data.cContent);//읽은 페이지  $('#updateForm [name="cCurrentPage"]').val(data.cCurrentPage);//읽은 페이지  $('#updateForm input').eq(3).attr('placeholder','전체페이지: '+data.cCurrentPage+'/'+ totalPage);  $('#updateForm [name="cStartDate"]').val(data.cStartDate);//등록한 날짜  $('#updateForm [name="cBookFinish"]').val(data.cBookFinish);//등록한 날짜 });//request.done  request.fail(**function**( jqXHR, textStatus ) {  alert('수정도서 정보를 불러오는데 실패하였습니다');  });//request.fail  }  });//calendar  calendar.render();    //도서 검색 & 검색 리스트  $('#bookInfoBtn').click(**function**(){  //리스트출력부분 초기화  $('#insertForm').find('ul').css("display","none").html('');    **var** biName = $('#bookName').val();  **var** request = $.ajax({  url: "/lifrary/getBooKInfo",  data: { biName : biName },  method: "POST",  dataType: "json"  });  request.done(**function**( data ) {  console.log(data);  bookInfoList = data;  **if**(data.length > 0){//리턴값 있는 경우 리스트출력  **for**(**var** i = 0; i<data.length; i++){  **var** list = data[i];  **var** ul = $('#insertForm').find('ul');  ul.css("display","block");  ul.append('<li></li>');  **var** li = ul.children('li').eq(i);  li.text(list.biName);  li.attr('value',i);  }  }**else**{//리턴값 없는 경우  **var** ul = $('#insertForm').find('ul');  ul.css("display","block");  ul.append('<li></li>');  **var** li = ul.children('li')  li.text('검색조건에 맞는 도서가 없습니다');  li.css( "height", "50px" );  li.css( "text-align", "center" );  }  });//request.done  request.fail(**function**( jqXHR, textStatus ) {  alert('검색도서 정보를 불러오는데 실패하였습니다');  });//request.fail  })  //도서검색 리스트 클릭시 해당 도서 정보 화면에 출력  $('#insertForm').on('click','li',**function**(){  $('#insertForm').find('ul').css("display","none");  **var** liText = $(**this**).text();//검색도서명  $('#insertForm input').eq(0).val(liText);    **var** i = $(**this**).val();  **var** selectBook = bookInfoList[i];//검색도서 정보  totalPage = selectBook.bsTotalPage  //검색도서 정보 화면에 출력  $('#insertForm strong').eq(1).text(selectBook.biName);//도서명  $('#insertForm strong').eq(3).text(selectBook.biIsbn);//ISBN  $('#insertForm strong').eq(5).text(selectBook.biAuthor);//저자명  $('#insertForm strong').eq(7).text(selectBook.biPublisher);//발행자  $('#insertForm input').eq(3).attr('placeholder','전체페이지: /'+totalPage);  $('#insertForm [name="bsCode"]').val(selectBook.bsCode);    });  });//document.ready |

|  |
| --- |
| 북 캘린더 페이지 |
| 목표량을 등록하여 진행상황을 확인할 수 있다. 북 캘린더에 도서 일정을 등록, 수정, 삭제할 수 있다. |

**(31) 내 독후감**

|  |
| --- |
|  |

|  |
| --- |
| 내 독후감 페이지 |
| 독후감 리스트를 확인하고 검색할 수 있다. 제목을 클릭하여 독후감 상세 페이지로 이동할 수 있다. 수정 버튼을 클릭하여 독후감 수정 페이지로 이동할 수 있다. 삭제버튼을 클릭하여 삭제할 수 있다. |

**(32) 나만의 서재**

|  |
| --- |
|  |
| [그림 1]  카테고리 버튼 클릭 시 해당 카테고리에 등록된 도서만 표시된다.    [그림 2]  도서에 마우스 커서를 올리면 도서 정보가 표시된다. 상세보기를 클릭하면 도서 상세페이지로 이동한다. 수정, 삭제를 클릭하면 카테고리 수정 또는 카테고리에서 삭제할 수 있다.    [그림 3]  등록 클릭 시 서재 카테고리 등록 Modal이 나타난다. 새로운 카테고리명을 등록할 수 있다.    [그림 4]  수정 클릭 시 서재 카테고리 수정 Modal이 나타난다. 등록된 카테고리 리스트가 표시되며 카테고리를 수정, 삭제할 수 있다. |

|  |
| --- |
| 나만의 서재 카테고리 |
| 서재 카테고리를 등록, 수정, 삭제할 수 있다. 서재 카테고리에 등록된 도서리스트를 확인할 수 있다. 서재 카테고리에 도서를 등록, 수정, 삭제할 수 있다. |

**(33)내 프로그램 신청 내역**

|  |
| --- |
|  |

■HTML: 신청한 프로그램 리스트를 출력한다. 없으면 "프로그램이없습니다"가출력된다.

|  |
| --- |
| <th:blockth:if=*"${#lists.size(applyList)} > 0"*>  <trth:each=*"apply, al : ${applyList}"*>  <tdth:text=*"${al.count}"*></td>  <td><ath:href=*"@{/lifrary/programApplyDetail(paCode = ${apply.paCode})}"*th:text=*"${apply.pmName}"*></a></td>  <tdth:text=*"${apply.uId}"*></td>  <tdth:text=*"${apply.paParticipants}"*></td>  <tdth:text=*"${apply.paGender}"*></td>  <tdth:text=*"${apply.paBirth}"*></td>  <tdth:text=*"${apply.paTel}"*></td>  <tdth:text=*"${apply.paCondition}"*></td>  <tdth:text=*"${apply.paDate}"*></td>  </tr>  </th:block>  <th:blockth:unless=*"${#lists.size(applyList)} > 0"*>  <tr>  <tdcolspan=*"9"*style="height: *150px*; text-align: *center*; vertical-align: *middle*;">신청프로그램이없습니다</td>  </tr>  </th:block> |

|  |
| --- |
| 프로그램 신청 리스트 페이지 |
| 신청한 프로그램 리스트를 확인할 수 있다.  신청한 프로그램의 이름을 클릭하면 상세보기 페이지로 이동 가능하다. |

**(34)내 공공시설 신청 내역**

|  |
| --- |
|  |

■javaScript:연장 버튼 클릭시 사용 종료시간에서 1시간을 더해준다. 퇴실 버튼 클릭시 사용종료 시간을 현재시간으로 update한다.

|  |
| --- |
| <script>  $(document).ready(**function**(){  /\* 연장버튼클릭시 \*/  $('.extension').click(**function**(){  **var**frCode = $(**this**).parents('tr').find('input[name=frCode]').val();    **var** request = $.ajax({ // fr코드받아서  url: "/lifrary/extension",  method: "POST",  data: { "frCode" : frCode },  dataType: "text"  });    request.done(**function**( data ) {  console.log(data)  **if**(data == '연장완료'){  alert('연장이완료되었습니다.')  location.reload(**true**);    }**elseif**(data == '연장불가'){  alert('연장이불가능합니다.');  }  });    request.fail(**function**( jqXHR, textStatus ) {  alert( "Request failed: " + textStatus );  });  })  /\* 퇴실버튼클릭시 \*/  $('.leaving').click(**function**(){  **var** frCode = $(**this**).parents('tr').find('input[name=frCode]').val();    **var** request = $.ajax({ // fr코드받아서  url: "/lifrary/leaving",  method: "POST",  data: { "frCode" : frCode },  dataType: "text"  });    request.done(**function**( data ) {  alert(data);  location.reload(**true**);  });    request.fail(**function**( jqXHR, textStatus ) {  alert( "Request failed: " + textStatus );  });  })    $('.impossible').click(**function**(){  **var** result = $(**this**).text();  alert(result+ '이불가능합니다.');  })  })  </script> |

■SQL : 연장 버튼 클릭시 종료시간에서 1시간을 더하고, 연장상태를 'O'로 변경하는쿼리문 실행

|  |
| --- |
| <!--연장버튼누른것에대하여연장처리해줌. -->  <updateid=*"extensionChange"*parameterType=*"String"*>  UPDATE facility\_reservation AS fr  SET  fr.fr\_end\_date = DATE\_ADD(fr.fr\_end\_date, INTERVAL 1 HOUR),  fr.fr\_extension='O'  WHERE  fr.fr\_code = #{frCode}  </update> |

■SQL : 시설 종류에 따라 예약 리스트 출력을 다르게 하는 쿼리문 실행

|  |
| --- |
| <!--마이페이지시설종류에따른예약리스트출력 -->  <selectid=*"getFacilityReservationList"*parameterType=*"String"*resultMap=*"reserveResultMap"*>  SELECT  fr.\*,  f.f\_kinds,  f.f\_name,  f.f\_location  FROM  facility\_reservation AS fr INNER JOIN facility AS f  ON fr.f\_code = f.f\_code  WHERE  fr.u\_id = #{uId} AND  fr.l\_code = #{libNum} AND  f.f\_kinds = #{fKinds}  </select> |

|  |
| --- |
| 공공시설 신청 내역 리스트 페이지 |
| DB에서 접속한 계정이 신청한 공공시설 리스트를 확인할 수 있다.  연장 버튼을 누르면 1회에 한해 연장처리가 된다.  연장 처리가 되면 연장 상태가 'O'로 업데이트 되며, 연장상태가 'O'인것은 연장이 불가하다.  퇴실 버튼을 누르면 예약 종료시간이 현재시간으로 업데이트 되어 바로 사용 종료된다.  연장, 퇴실버튼을 클릭후 처리가 완료되면 페이지가 새로고침 된다. |

**(35) 내 희망도서 신청 내역 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 희망도서 신청 확인페이지(마이페이지) |
| 권한이 회원인 경우만 접근이 가능하며 해당 회원이 신청한 도서의 정보와, 신청일, 진행상황, 취소사유를 보여준다. |

**6.3.2 관리자 사이트**

**(1)로그인**

|  |
| --- |
|  |

|  |
| --- |
| <selectid=*"adminLoginCheck"*resultMap=*"userResultMap"*>  SELECT  u.u\_id,  u.u\_pw,  u.l\_code,  u.u\_division,  u.u\_name,  ll.ll\_insert,  ll.ll\_carry,  ll.ll\_book\_admin,  ll.ll\_stats,  ll.ll\_book\_stock  FROM  user AS u INNER JOIN librarian\_level AS ll  ON  u.u\_id = ll.u\_id  WHERE  u.u\_id = #{uId} and (u.u\_division = '관리자' or u.u\_division = '사서')  </select> |

|  |
| --- |
| 사서채널 - 로그인 페이지 |
| 접속한 서서 페이지의 도서관 코드와 회원 코드가 일치하는 경우,  해당 사서 페이지의 회원이므로 로그인이 된다.  로그인시 DB에서 접속한 아이디와 일치하는 사서 권한을 가져온다. |

**(2) 도서관 관리자 메인 페이지**

|  |
| --- |
|  |

■Controller : 관리자 index 진입시 4개의 select 쿼리문 실행하여 소장도서,금일 대출도서, 사서인원,회원 인원이 화면에 출력된다.

■Controller : 관리자 index 진입시 1달전~현제 까지 api를 활용하여 조건에맞는 이미지, 제목을 가져와

화면에 출력해준다.

■SQL : 조건에 맞느 4가지 카운트 하는 select 쿼리문 실행

|  |
| --- |
| <selectid=*"bookStockCount"*parameterType=*"String"*resultType=*"String"*>  SELECT  COUNT(bs\_stock\_state) AS bsStockState  FROM  book\_stock  WHERE bs\_stock\_state = '소장' AND l\_code = #{libnum}  </select>    <selectid=*"bookLendCount"*parameterType=*"String"*resultType=*"String"*>  SELECT  COUNT(bl\_lend\_date) AS blLendDate  FROM  book\_lend  WHERE bl\_lend\_date = DATE(NOW()) AND l\_code = #{libnum}  </select>    <selectid=*"librarianCount"*parameterType=*"String"*resultType=*"String"*>  SELECT  COUNT(u\_division) AS librarian  FROM  user  WHERE u\_division = '사서' AND l\_code = #{libnum}  </select>    <selectid=*"memberCount"*parameterType=*"String"*resultType=*"String"*>  SELECT  COUNT(u\_division) AS member  FROM  user  WHERE u\_division = '회원' AND l\_code = #{libnum}  </select> |

■OpenApi : 조건에맞는 도서명, 이미지를 불러온다.

|  |
| --- |
| package ksmart.pentagon.controller;  import java.io.BufferedInputStream;  import java.net.URL;  import java.util.ArrayList;  import org.xmlpull.v1.XmlPullParser;  import org.xmlpull.v1.XmlPullParserFactory;  public class AreaOpenApi {      private static String PHARM\_URL;  private static ArrayList<ArrayList> list;  private static ArrayList<String> book;  public ArrayList<ArrayList> OpenApi(String startDt , String endDt, String aera) {  System.out.println("Api 실행");  if(aera != "0") {  PHARM\_URL = "http://data4library.kr/api/loanItemSrch?authKey="  + "86b2aa39b6cd044028fdadb621d0907b5982a7b8a9f5e77514e3bebd85cfccb5"  + "&startDt="+startDt  + "&endDt="+endDt  + "&region="+aera;  }else if(aera == "0") {  PHARM\_URL = "http://data4library.kr/api/loanItemSrch?authKey="  + "86b2aa39b6cd044028fdadb621d0907b5982a7b8a9f5e77514e3bebd85cfccb5"  + "&startDt="+startDt  + "&endDt="+endDt;  }  try {  apiParserSearch();  } catch (Exception q) {  q.printStackTrace();  }  return list;  }  /\*\*  \*  \* @throws Exception  \*/  public void apiParserSearch() throws Exception {  URL url = new URL(getURLParam(null));  XmlPullParserFactory factory = XmlPullParserFactory.newInstance();  factory.setNamespaceAware(true);  XmlPullParser xpp = factory.newPullParser();  BufferedInputStream bis = new BufferedInputStream(url.openStream());  xpp.setInput(bis, "utf-8");  String tag = null;  int event\_type = xpp.getEventType();  System.out.println("xpp : " +xpp.getEventType());  list = new ArrayList<ArrayList>();  int i = 0;  String bookname = "";  String bookImageURL = "";  while (event\_type != XmlPullParser.END\_DOCUMENT) {  if (event\_type == XmlPullParser.START\_TAG) {  tag = xpp.getName();  } else if (event\_type == XmlPullParser.TEXT) {  /\*\*  \* 책의 이름을 가져온다.  \*/  if(tag.equals("bookname")){  if(i < 10) {  book = new ArrayList<String>();  System.out.println("bookname 담김 : " + bookname);  bookname = xpp.getText();  book.add(bookname);  }  }  /\*\*  \* 책의 이미지URL을 가져온다.  \*/  if(tag.equals("bookImageURL")){  if(i < 10) {  i += 1;  System.out.println("bookImageURL 담김 : " + bookImageURL);  bookImageURL = xpp.getText();  book.add(bookImageURL);  list.add(book);  }  }  }  event\_type = xpp.next();  }  printList(list);  }  /\*\*  \* 결과 값을 출력해본다.  \* @param list  \*/  private void printList(ArrayList<ArrayList> list){  for(ArrayList entity : list){  System.out.println(entity);  }  }  private String getURLParam(String search){  String url = PHARM\_URL;  if(search != null){  url = url+"&yadmNm"+search;  }  return url;  }      } |

|  |
| --- |
| 관리자 메인 페이지 |
| 소장도서,금일대출도서,사서,회원 수를 확인할 수 있다.  인기도서를 지역,연령,성별에 맞는 버튼을 누르면 ajax로 도서명,도서 이미지를 화면에 출력해준다. |

**(3) 관리자 회원 검색리스트**

|  |
| --- |
|  |

|  |
| --- |
| 관리자 회원 전체 검색리스트 |
| 회원 전체 검색리스트에서 회원정보를 볼 수 있다 |

**(4) 관리자 회원수정 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 관리자 회원 수정 페이지 |
| 관리자/사서 회원을 수정한다 |

**(5) 관리자 회원 상세보기 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 관리자 회원 상세보기 페이지 |
| 관리자/사서 회원 정보를 상세 본다 |

**(6) 관리자 회원정보 삭제페이지**

|  |
| --- |
|  |

|  |
| --- |
| 관리자 회원 정보 삭제 페이지 |
| 관리자/사서가 회원을 삭제 할 때 모달창이 나오고 비밀번호가 맞으면 삭제된다. |

|  |
| --- |
| <!-- teamScript(각페이지에서각자구현할자바스크립트) -->  <th:blocklayout:fragment=*"teamHeadScript"*>  <scripttype=*"text/javascript"*th:inline=*"javascript"*>  $(document).ready(**function**(){    $('#deleteBtn').click(**function**(){  **var** SAID = [[${session.SAID}]]  console.log(SAID);  **var** write = $('#uPw').val();  console.log(write);  **var**uId = $('#uId').val();  console.log(uId);      **var** request = $.ajax({  url:"/deleteUser",  method: "POST",  data : { 'said' : SAID , 'write' : write , 'uId' : uId},  dataType: "text"  });  //응답완료후실행 - 정상처리  request.done(**function**( data ) {  console.log(data);  **if**(data == '비밀번호가틀렸습니다'){  alert(data);  $('#uPw').focus();  }**elseif**(data =='회원삭제가완료되었습니다'){  alert(data);  $('#modals-default').modal('hide');  location.reload(**true**);  }  });  request.fail(**function**( jqXHR, textStatus ) {  alert( "Request failed: " + textStatus );  });  });    });  </script>  </th:block>  <!-- / teamScript(각페이지에서각자구현할자바스크립트) -->    <! ---------------------------—modal 창 ----------------------->  <divclass=*"modal fade"*id=*"modals-default"*>  <divclass=*"modal-dialog"*>  <formclass=*"modal-content"*>  <divclass=*"modal-header"*>  <h5class=*"modal-title"*>  회원목록삭제  <br>  <smallclass=*"text-muted"*>정말삭제하시겠습니까? 삭제하시는경우복구가불가능한목록입니다.<br>  삭제를위하여본인 id의 password를작성해주세요.</small>  </h5>  <ath:href=*"@{/admin/userSearchList}"*type=*"button"*class=*"close"*data-dismiss=*"modal"*aria-label=*"Close"*>X</a>  </div>  <divclass=*"modal-body"*>  <divclass=*"form-row"*>  <divclass=*"form-group col"*>  <labelclass=*"form-label"*>ID</label>  <inputid=*"uId"*type=*"text"*class=*"form-control"*th:value=*"${session.SAID}"*>  </div>  </div>  <divclass=*"form-row"*>  <divclass=*"form-group col"*>  <labelclass=*"form-label"*>PASSWORD</label>  <inputid=*"uPw"*type=*"password"*class=*"form-control"*placeholder=*"\*\*\*\*\*\*\*\*"*>  </div>  </div>  </div>  <divclass=*"modal-footer"*>  <buttontype=*"button"*class=*"btnbtn-default"*data-dismiss=*"modal"*>Close</button>  <ath:href=*"@{/admin/userSearchList}"*id=*"deleteBtn"*type=*"button"*class=*"btnbtn-primary"*>삭제</a>  </div>  </form>  </div>  </div> |

**(7) 관리자 회원 등급 리스트 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 관리자 회원 등급리스트 페이지 |
| 관리자/사서가 회원등급 리스트를 조회한다. |

**(8) 관리자 회원등급 등록페이지**

|  |
| --- |
|  |

|  |
| --- |
| 관리자 회원등급등록 페이지 |
| 관리자/사서가 회원등급을 등록한다. |

**(9) 관리자 회원등급 수정 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 관리자 회원등급수정 페이지 |
| 관리자/사서가 회원등급을 수정한다. |

**(10) 관리자 회원등급 검색내역 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 관리자 회원등급 검색내역 페이지 |
| 관리자/사서가 회원등급 검색 내역을 조회, 삭제 한다. |

**(11) 관리자 회원권한 리스트페이지**

|  |
| --- |
|  |

|  |
| --- |
| 관리자회원권한 리스트페이지 |
| 관리자/사서가 회원권한을 등록, 조회한다. |

**(12) 관리자 회원권한 등록페이지**

|  |
| --- |
|  |

|  |
| --- |
| 관리자회원권한 리스트페이지 |
| 관리자/사서가 회원권한을 등록한다. |

**(13) 관리자 회원권한 수정페이지**

|  |
| --- |
|  |

|  |
| --- |
| 관리자회원권한 수정페이지 |
| 관리자/사서가 회원권한을 수정한다. |

**(14) 관리자 회원권한 내역 검색리스트페이지**

|  |
| --- |
|  |

|  |
| --- |
| 관리자회원권한 내역 검색리스트 페이지 |
| 관리자/사서가 회원권한 내역 검색리스트를 조회, 삭제 한다. |

**(15) 사서관리- 사서 정보 상세보기 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 사서관리- 사서 정보 상세보기 페이지 |
| 관리자가 로그인한 경우 사서관리메뉴(사서 나의관리, 사서 권한 설정) 가 보입니다.  사서 나의 관리메뉴를 누르면 내 정보를 상세내용이 확인 가능하다. |

**(16) 사서관리- 사서 정보 수정 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 사서관리- 사서 정보 수정 페이지 |
| 유효성 검사를 다 마친후, 비밀번호 확인이 맞으면 수정이 완료된다. |

**(17) 사서관리- 사서 권한리스트 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 사서관리- 사서 권한 리스트 페이지 |
| 관리자로 로그인 한 후, 사서권한 리스트를 조회한다. |

**(18) 사서관리- 사서 권한 등록 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 사서관리- 사서 권한 등록페이지 |
| 관리자로 로그인 한 후, 사서를 등록한다.  사서권한은 선택하지 않으면 미부여가 된다. |

**(19) 사서관리- 사서 권한 수정 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 사서관리- 사서 권한 수정페이지 |
| 관리자로 로그인 한 후, 선택한 사서를 수정한다.  유효성 검사가 끝나고 난뒤 비밀번호 확인 후 수정이 완료된다. |

**(20) 사서관리- 사서 정보 상세보기 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 사서관리- 사서 정보 상세보기 페이지 |
| 관리자로 로그인 한 후, 선택한 사서 상세내용을 확인한다.  리스트로이동버큰을 누르면 리스트로 이동한다. |

**(21) 사서관리- 사서 정보 상세보기 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 사서관리- 사서 정보 상세보기 페이지 |
| 사서로로그인을 하면, 사서관리메뉴 증 ‘사서 나의관리’ 메뉴만 보여지고 내 상세내용을 확인가능 하다. |

**(22) 포인트 관리**

|  |
| --- |
|  |
| [그림 1]  포인트 등록 버튼을 클릭하면 포인트 등록 Modal이 나타난다. 포인트명, 포인트, 포인트 지급 기준을 입력하고 등록한다.    [그림 2]  수정 아이콘을 클릭하면 포인트 수정 Modal이 나타난다. 포인트명, 포인트, 포인트 지급 기준을 입력하고 수정한다 |

|  |
| --- |
| 포인트 관리 페이지 |
| 포인트 리스트를 확인하고 검색할 수 있다. 포인트를 등록, 수정, 삭제할 수 있다. |

**(23)포인트 사용 내역**

|  |
| --- |
|  |

|  |
| --- |
| 포인트 사용 내역 페이지 |
| 포인트 사용 내역 리스트를 확인하고 검색할 수 있다. 사용 내역을 삭제할 수 있다. 회원 상세보기 페이지로 이동할 수 있다. |

|  |
| --- |
|  |

**(24) 희망도서 신청 리스트**

|  |
| --- |
|  |

|  |
| --- |
| 희망도서 신청 리스트 |
| 도서관 페이지에서 회원이 신청한 희망도서리스트를 조회한다.  상세버튼 클릭 시 각각 신청건의 상세정보를 조회한다.  진행상황의 경우 버튼을 클릭하면 모달화면이 나오며, 상태를 수정가능하다 |

**(25) 구매예정 도서 리스트**

|  |
| --- |
|  |
|  |

|  |
| --- |
| 구매예정 도서 리스트 |
| 구매예정 도서를 조회한다.  삭제의 경우 버튼을 클릭하면 모달화면이 나오며, 수서권한이 있는 사서의 경우 자신의 password입력시 목록 삭제가 가능하다. |

**(26) 구매예정 도서 등록 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 구매예정 도서 등록 페이지 |
| ISBN을 입력 후 [ 도서정보 가져오기 ] 버튼 클릭을 하는 경우, 도서API를 사용하여 해당 도서의 정보를 가져오며 도서정보 테이블에 등록한다.  API가 제공하는 정보 이외의 정보를 입력하면 구매예정 테이블에 등록 후 리스트로 이동한다.  ( 가져온 도서정보를 화면에서 변경하는 경우 조서정보 테이블에 수정이 이루어진다.) |

**(27) 구매예정 도서 상세,수정 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 구매예정 도서 상세,수정 페이지 |
| 구매예정 도서의 상세내용을 확인함과 동시에, 수정내용을 입력 후 수정버튼을 입력하는 경우 수정처리가 이루어진다.  수정처리 후 리스트로 이동한다. |

**(28) 주문 도서 리스트**

|  |
| --- |
|  |

|  |
| --- |
| 주문 도서 리스트 |
| 주문 완료된 도서를 조회한다.  삭제의 경우 구매예정 도서와 동일하다.( 모달, 사서 password확인 ) |

**(29) 주문 도서 등록 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 주문 도서 등록 페이지 |
| 주문 도서의 상세내용을 확인함과 동시에, 수정내용을 입력 후 수정버튼을 입력하는 경우 수정처리가 이루어진다.  수정처리 후 리스트로 이동한다 |

**(30) 주문 도서 상세,수정 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 주문 도서 상세,수정 페이지 |
| 주문 도서의 상세내용을 확인함과 동시에, 수정내용을 입력 후 수정버튼을 입력하는 경우 수정처리가 이루어진다.  수정처리 후 리스트로 이동한다 |

**(31) 기부 신청자 리스트**

|  |
| --- |
|  |

|  |
| --- |
| 기부 신청자 리스트 |
| 기부 신청자를 조회한다.  삭제의 경우 구매예정 도서와 동일하다.( 모달, 사서 password확인 )  기증스티커와 명예전당 상태의 경우 버튼을 클릭하면 상태가 변경된다. |

**(32) 기부 신청자 등록 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 기부 신청자 등록 페이지 |
| 기부 신청자를 등록한다. |

**(33) 기부 신청자 상세,수정 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 기부 신청자 상세, 수정페이지 |
| 기부 신청자의 상세내용을 확인함과 동시에, 수정내용을 입력 후 수정버튼을 입력하는 경우, 수정처리가 이루어진다.  수정처리 후 리스트로 이동한다 |

**(34) 소장도서 리스트**

|  |
| --- |
|  |

|  |
| --- |
| 소장도서 리스트 |
| 소장도서 리스트를 조회한다. |

**(35) 소장도서 등록 페이지 (등록 과정)**

|  |
| --- |
|  |
| [그림 1]  도서 ISBN를 입력 후 [ 도서정보 가져오기 ] 버튼을 클릭한다. |
| [ 그림 2 ]  도서API를 사용하여 해당 도서의 정보를 가져와서 화면에 출력되며, 도서정보 테이블에 등록한다. |
| [ 그림 3 ]  KBC정보가 없는경우 [ 분류기호 선택 ] 버튼을 클릭하여 분류기호를 선택한다. |
| [ 그림 4 ]  [ 청구기호 생성 ]버튼을 클릭하여 저자명과 도서명으로 청구기호를 생성하여 화면에 출력한다. |
| [ 그림 5 ]  나머지 정보를 입력 후 [ 소장 도서 추가 ]버튼을 클릭하면 도서가 등록된 다음 리스트로 이동한다 |

|  |
| --- |
| 소장도서 등록 페이지 |
| 소장도서를 등록한다 |

■도서 API로 도서정보 가져오기

|  |
| --- |
| **public**BookInformationgetBookInfoStock(String biIsbn) {    BookInformationbi = bookStockMapper.getBookInfoStock(biIsbn);  **if** ( bi != **null** ) {  }**elseif**(bi == **null**){    String url="http://data4library.kr/api/srchDtlList?"  +"authKey=86b2aa39b6cd044028fdadb621d0907b5982a7b8a9f5e77514e3bebd85cfccb5&"  +"isbn13="+biIsbn+"&loaninfoYN=Y";    **try**{  DocumentBuilderFactorydbFactory = DocumentBuilderFactory.*newInstance*();  DocumentBuilderdBuilder = dbFactory.newDocumentBuilder();    Document doc = dBuilder.parse(url);  doc.getDocumentElement().normalize();    System.***out***.println("최상위테그노드네임=>"  +doc.getDocumentElement().getNodeName());    NodeListnList = doc.getElementsByTagName("book");  System.***out***.println("파싱할리스트수 =>"+ nList.getLength() );    **for**(**int**temp=0; temp<nList.getLength(); temp++) {  Node nNode = nList.item(temp);    **if**(nNode.getNodeType() == Node.***ELEMENT\_NODE***) {    bi = **new**BookInformation();    Element eElement = (Element)nNode;    String biName= *getTagValue*("bookname",eElement).trim();  String biYear = *getTagValue*("publication\_year",eElement).trim();  String biAuthor = *getTagValue*("authors",eElement).trim();  String biPublisher = *getTagValue*("publisher",eElement).trim();  String biKdc = *getTagValue*("class\_no",eElement).trim();  String biImg = *getTagValue*("bookImageURL",eElement).trim();  String biDtail = *getTagValue*("description",eElement);      bi.setBiIsbn(biIsbn);  bi.setBiYear(biYear);  bi.setBiName(biName);  bi.setBiAuthor(biAuthor);  bi.setBiPublisher(biPublisher);  bi.setBiDtail(biDtail);  bi.setBiImg(biImg);  bi.setBiKdc(biKdc);    bookStockMapper.insertBookInfoStock(bi);    }  }  }**catch**(Exception e) {  e.printStackTrace();  }    }  **return**bi;  }    **privatestatic** String getTagValue(String tag, Element eElement) {  String val = "";  **if**(eElement!=**null**) {  NodeListnlList = eElement.getElementsByTagName(tag);  Element el = (Element)nlList.item(0);  **if**( el != **null**) {  Node nValue = el.getChildNodes().item(0);  **if**(nValue!=**null**) {  val = nValue.getNodeValue();  }  }  }  **return**val;  } |

■청구기호 생성

|  |
| --- |
| **< 한글 자모음 유니코드 class >**  **publicclass**StaticCodeClass {  /\*ㄱㄲㄴㄷㄸㄹㅁㅂㅃㅅㅆㅇㅈㅉㅊㅋㅌㅍㅎ \*/  **publicstaticfinalchar**[] ***CHO*** =  {0x3131, 0x3132, 0x3134, 0x3137, 0x3138, 0x3139, 0x3141, 0x3142, 0x3143, 0x3145,  0x3146, 0x3147, 0x3148, 0x3149, 0x314a, 0x314b, 0x314c, 0x314d, 0x314e};    /\*ㅏㅐㅑㅒㅓㅔㅕㅖㅗㅘㅙㅚㅛㅜㅝㅞㅟㅠㅡㅢㅣ\*/  **publicstaticfinalchar**[] ***JUN*** =  {0x314f, 0x3150, 0x3151, 0x3152, 0x3153, 0x3154, 0x3155, 0x3156, 0x3157, 0x3158,  0x3159, 0x315a, 0x315b, 0x315c, 0x315d, 0x315e, 0x315f, 0x3160, 0x3161, 0x3162,  0x3163};    /\*X ㄱㄲㄳㄴㄵㄶㄷㄹㄺㄻㄼㄽㄾㄿㅀㅁㅂㅄㅅㅆㅇㅈㅊㅋㅌㅍㅎ\*/  **publicstaticfinalchar**[] ***JON*** =  {0x0000, 0x3131, 0x3132, 0x3133, 0x3134, 0x3135, 0x3136, 0x3137, 0x3139, 0x313a,  0x313b, 0x313c, 0x313d, 0x313e, 0x313f, 0x3140, 0x3141, 0x3142, 0x3144, 0x3145,  0x3146, 0x3147, 0x3148, 0x314a, 0x314b, 0x314c, 0x314d, 0x314e};    }  **< 청구기호 생성 service 단 1 –한글 자모음으로 분해 >**  @Service  **publicclass**MakeBsmarkService {  **private** List<Map<String, Integer>>intList;  **private** List<Map<String, Character>>charList;    **privatevoid** action(String str, **boolean**isChartype) {    **if**(!isChartype) intList = **new**ArrayList<Map<String, Integer>>();  **if**(isChartype) charList = **new**ArrayList<Map<String, Character>>();    **if**( str != **null** ) {  **for**(**int**i = 0 ; i<str.length(); i++){  **char**character = str.charAt(i);  **if**(character>= 0xAC00){  // 유니코드  **char**uniVal = (**char**) (character - 0xAC00);  **char**cho = (**char**) (((uniVal - (uniVal % 28))/28)/21);  **char**jun = (**char**) (((uniVal - (uniVal % 28))/28)%21);  **char**jon = (**char**) (uniVal %28);    // 각배열에유니코드넣어서해당하는한글출력  **char**reCho = StaticCodeClass.***CHO***[cho];  **char**reJun = StaticCodeClass.***JUN***[jun];    **if**(isChartype) {    Map<String, Character>charMap = **new**HashMap<String, Character>();  charMap.put("char", (**char**) character);  charMap.put("reCho", (**char**) reCho);  charMap.put("reJun", (**char**) reJun);    **if**((**char**)jon != 0x0000) {  **char**reJon = StaticCodeClass.***JON***[jon];  charMap.put("reJon", (**char**) reJon);  }    charList.add(charMap);  }    **if**(!isChartype) {    Map<String, Integer>intMap = **new**HashMap<String, Integer>();  intMap.put("cho", (**int**) cho);  intMap.put("jun", (**int**) jun);  **if**((**char**)jon != 0x0000) { intMap.put("jon", (**int**) jon);  }  intList.add(intMap);  } }  }  }  }    **public** List<Map<String, Integer>>getIntList(String str){  action(str, **false**);  **return**intList;  }    **public** List<Map<String, Character>>getCharList(String str){  action(str, **true**);  **return**charList;  }    }  **<청구기호 생성 service 단 - 저자기호, 도서기호 생성 >**  @Service  **publicclass**CallNumberService {    @Autowired**private**CallNumberMappercallNumberMapper;  @Autowired**private**MakeBsmarkServicemakeBsmarkService;    // 저작기호 = 저자기호 + 도서기호  // 저자기호구하기  **public** String makeBsmarkAuthor(String biAuthor, String lCode) {    List<Map<String, Character>>charList =  makeBsmarkService.getCharList(biAuthor);    String resultStr = "";  **if**(charList != **null**&&charList.size() > 1) {    resultStr += charList.get(0).get("char");    Map<String, Character>charNextMap = charList.get(1);    Character reCho = charNextMap.get("reCho"); //1번째글자  Character reJun = charNextMap.get("reJun"); //2번째글자    **if**(reCho != **null**&& !"".equals(reCho.toString())) {  resultStr += makeBsmarkNum(reCho.toString());  }  **if**(reJun != **null**&& !"".equals(reJun.toString())) {  resultStr += makeBsmarkNum(reJun.toString());  }    // informaton테이블에같은작가명있는지체크하는메서드  List<String>author = callNumberMapper.checkAuthor(biAuthor);  // stock 테이블에같은작가기호있는지체크하는메서드  List<String>authorMark = callNumberMapper.checkAuthorMark(resultStr,lCode);  String plus = "1";  String resultStrPlus = "";    // 작가명자체가겹칠때 = 같은작가(동일인물) **if**(author.size()>0) {    // 작가명자체가겹치지않을때  }**else** {  **if**(authorMark.size()>0) {  // 작가명이겹치지않으나, 작가기호가겹칠때  resultStr += plus;  }**else**{ // 작가명이겹치지않고작가기호도겹치지않을때  }  }  }  **return**resultStr;  }    // 저자기호구하기 (DB에접근해서 2번째글자의숫자기호구하기)  **public** String makeBsmarkNum(String text) {  System.***out***.println("=============== 서비스메서드실행 ===============");  String result = callNumberMapper.makeBsmarkNum(text);  System.***out***.println("메서드실행..result=>"+result);  **return**result;  }    // 도서기호구하기 1  **public** String makeBsmarkName(String biName) {  System.***out***.println("============ makeBsmarkName ============");  List<Map<String, Character>>nameList = makeBsmarkService.getCharList(biName);  String resultStr = "";  **if**(nameList != **null**&&nameList.size() > 1) {  Map<String, Character>nameNextMap = nameList.get(0);  Character reCho = nameNextMap.get("reCho"); //1번째  **if**(reCho != **null**&& !"".equals(reCho.toString())) {  resultStr += reCho.toString();  }  }  **return**resultStr;  }    // 도서기호구하기2  **public** String makeBsmarkName2(String biName) {  System.***out***.println("============ makeBsmarkName ============");  List<Map<String, Character>>nameList = makeBsmarkService.getCharList(biName);  List<Map<String, Integer>>nIntList =  makeBsmarkService.getIntList(biName);  String resultStr = "";    **if**(nIntList != **null**&&nIntList.size() > 1) {  Map<String, Integer>nameIntNextMap = nIntList.get(0);  **int**a = (**int**)nameIntNextMap.get("cho"); //1번째  **int**b = (**int**)nameIntNextMap.get("jun"); //2번째  **if**(a != 0) {  Character temp = (**char**)(0xAC00 + 28 \* 21 \*(a) + 28 \* (b));  resultStr = temp.toString();  }  }  **return**resultStr;  }  **public** List<String>checkWriterMark(String writer, String lCode) {  **return**callNumberMapper.checkWriterMark(writer,lCode);  }  } |

**(36) 소장도서 상세 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 소장도서 상세 페이지 |
| 소장도서의 상세정보를 확인 가능하다.  [ 도서관 상세페이지로 이동 ]버튼을 클릭하는 경우, 도서관 도서상세 페이지로 이동한다. |

**(37) 소장도서 수정 페이지**

|  |
| --- |
|  |
| [ 그림 1 ]  청구기호 수정의 경우 [ 청구기호 수정 ]버튼을 클릭하면 아래의 상세 수정 폼이 추가되어 나타난다. |

|  |
| --- |
| 소장도서 수정 페이지 |
| 소장도서의 상세정보 수정이 가능하다. |

**(38) 소장도서 삭제**

|  |
| --- |
|  |

|  |
| --- |
| 소장도서 삭제 (모달) |
| 관리자 권한인 경우, 삭제 사유와 해당 관리자의 password를 입력하면 해당 소정도서의 상태를 삭제 상태로 변경 가능하다.( 복구가능 ) |

**(39) 삭제도서 리스트**

|  |
| --- |
|  |

|  |
| --- |
| 삭제도서 리스트 |
| 삭제도서 리스트를 조회한다. |

**(40) 삭제도서 상세 페이지**

|  |
| --- |
|  |
|  |

|  |
| --- |
| 삭제도서 상세 페이지 |
| 삭제도서의 상세정보를 확인 가능하다. ( 삭제사유 등 )  [ 복원 ]버튼을 클릭하는 경우 해당 관리자의 password를 입력하면 도서를 정상소상도서 상태로 복원 할 수 있다. |

**(41) 대출/반납 관리**

|  |
| --- |
|  |
| [그림 1]  소장도서 코드로 도서를 검색하면 도서정보가 출력된다.    [그림 2]  회원번호로 회원을 검색하면 회원 정보가 출력된다.  [그림 3]  검색한 도서의 도서상태, 대출상태와 회원의 대출권한, 대출가능 권수가 0권이 아닌 경우 대출 등록 버튼이 표시된다.    [그림 4]  검색한 회원의 대출권한이 없는 경우 대출제한일이 표시된다.    [그림 5]  검색한 도서가 대출된 도서인 경우 도서정보와 회원정보가 동시에 출력되며 반납등록 버튼이 표시된다.    [그림 6]  예약 도서인 경우 대출상태가 예약으로 표시되며, 검색한 회원과 예약한 회원 정보가 다른 경우 대출권한에 예약자 불일치가 표시된다. |
| [그림 7]  예약 도서의 예약한 회원 정보가 일치하는 경우 대출권한에 예약자 일치가 표시되며, 대출 권한(대출가능)과 대출가능권수가 0이 아닌 경우 대출등록 버튼이 표시된다.    [그림 8]  대출 도서는 연장하기 버튼을 클릭하여 1회 반납예정일(7일 연장)을 연장할 수 있다. 1회 연장된 도서와 반납된 도서는 연장불가가 표시된다.  상세보기 도서, 회원 아이콘을 클릭시 도서, 회원 상세보기페이지로 이동한다. |

■ 검색한 회원정보를 가져오는 Query문

회원 등급에 따른 대출가능 권수와 대출일을 가져온다. 회원 사용 권한 이력을 확인하여 대출권한이 없는 경우 대출제한일을 계산해서 가져온다. 대출, 예약 도서 권수를 확인하여 가져온다.

|  |
| --- |
| <selectid=*"userInfo"*parameterType=*"String"*resultMap=*"user"*>  SELECT  u.u\_id  ,u.u\_name  ,u.u\_number  ,u.u\_overdue\_day  ,ul.ul\_name  ,ul.ul\_lend\_num  ,ul.ul\_lend\_day  ,uas.uas\_lend\_limit  ,DATE\_ADD(uah.uah\_date, INTERVAL uas.uas\_limit\_date DAY) AS u\_authority\_date  ,DATEDIFF( DATE\_ADD(uah.uah\_date, INTERVAL uas.uas\_limit\_date DAY) , NOW() ) AS u\_authority\_days  ,(SELECT  COUNT(1)  FROM  book\_lend AS bl  WHERE  bl.bl\_return\_date IS NULL  AND(  (bl.bl\_lend\_date IS NULL AND bl.bl\_hold\_date IS NOT NULL)  OR  bl.bl\_lend\_date IS NOT NULL)  AND  bl.bl\_id = u.u\_id ) AS u\_lend\_cnt  FROM  user AS u  INNER JOIN  user\_level AS ul  ON  u.ul\_level = ul.ul\_level  INNER JOIN  user\_authority\_set AS uas  ON  u.uas\_code = uas.uas\_code  LEFT JOIN  user\_authority\_history AS uah  ON  u.u\_id = uah.u\_id  WHERE  u.u\_number = #{svUser}  AND  u.l\_code = #{libNum}  ORDER BY  uah.uah\_date DESC  LIMIT 0,1  </select> |

■ 검색한 도서정보 SERVICE에서 처리

청구 기호를 생성한다. 대출 기록이 있는지 확인하고 대출기록이 있는 경우 가장 최근 기록을 가져온다. 대출, 예약도서인지 확인하고 대출된 도서인 경우 회원정보도 가져온다.

|  |
| --- |
| **public** Map<String, Object> bookInfo(String libNum, String svBook) {    BookStock bookStock = **null**;  Map<String, Object>bookInfoMap = **new** HashMap<String, Object>();    //앞뒤공백제거  String svBookTrim =svBook.trim();  //대출기록이 있는 도서인지 확인  **int**bookLendCheck = bookLendMapper.bookLendCheck(libNum, svBookTrim);  **if**(bookLendCheck == 0) bookStock = bookLendMapper.bookInfoStock(libNum, svBookTrim);  **else**bookStock = bookLendMapper.bookInfo(libNum,svBookTrim);    **if**(bookStock != **null**) {  String bsCallNum = "";    //청구기호를 생성을 위한 조건문 처리  String aliasMark = bookStock.getBsAliasMark();//별치기호  String kdc = bookStock.getBookInformation().getBiKdc();//분류기호  String writerMark = bookStock.getBsWriterMark();//저자기호  String bsSecondaryMark = bookStock.getBsSecondaryMark();//부차적기호    **if**(aliasMark != **null**) {  bsCallNum += aliasMark;  }  bsCallNum += kdc;  bsCallNum += writerMark;  **if**(bsSecondaryMark != **null**) {  bsCallNum += bsSecondaryMark;  }  bookStock.setBsCallNum(bsCallNum);  bookInfoMap.put("searchBook", 1);//도서정보있는 경우 1 대입    //대출기록이 있는 경우  **if**(bookStock.getBookLend() != **null**&&bookLendCheck != 0) {  String lendDate = bookStock.getBookLend().getBlLendDate();//대출일  String returnDate = bookStock.getBookLend().getBlReturnDate();//반납일    **if**(lendDate == **null**&&returnDate == **null**) {//예약 도서인지 확인하는 조건문  bookStock.setBsLendState("예약");  bookInfoMap.put("searchBook", 3);//예약도서인 경우 3 대입  }**elseif**(lendDate != **null**&&returnDate == **null**) {//대출된 도서인지 확인하는 조건문  //대출도서 회원정보 가져오기  User user = bookLendMapper.userInfo(libNum, bookStock.getuId());  **if**(user != **null**) {  //대출가능권수 구하기  **int**bookLendNum = 0;  **int**lendNum = user.getUserLevel().getUlLendNum();  **int**lendCnt = user.getuLendCnt();    **if**(lendNum>lendCnt) {  bookLendNum = lendNum-lendCnt;  }**else** {  bookLendNum = 0;  }  user.getUserLevel().setUlLendNum(bookLendNum);  //대출제한일이 null이거나 사용제한일이 지난 경우  **if**(user.getuAuthorityDate() == **null** ||  user.getuAuthorityDays() <= 0) {  user.setuAuthorityDate(" ");  }  }  bookInfoMap.put("resultUser", user);  bookInfoMap.put("searchBook", 2);//대출된 도서인 경우2 대입  }  }  }**else** {  bookInfoMap.put("searchBook", 0);//도서정보 없는 경우 0 대입  }  bookInfoMap.put("resultBook", bookStock);    **return**bookInfoMap;  } |

|  |
| --- |
| 대출/반납 관리 페이지 |
| 도서와 회원을 검색하여 정보를 확인하고 대출/반납을 등록한다. 대출/반납 도서 리스트를 확인하고 검색할 수 있다. 대출 도서의 경우 1회 반납예정일을 연장할 수 있다. 도서, 회원 상세보기 페이지로 이동할 수 있다. |

**(42) 예약 관리**

|  |
| --- |
|  |
| [그림1]  예약도서는 예약취소 버튼을 클릭하여 예약을 취소할 수 있다. 취소 완료된 예약도서는 취소완료가 표시된다. |

|  |
| --- |
| 예약관리 페이지 |
| 예약 도서 리스트를 확인하고 검색할 수 있다. 예약도서를 취소할 수 있다. 도서, 회원 상세보기 페이지로 이동할 수 있다. |

**(43) 공지사항**

|  |
| --- |
|  |

|  |
| --- |
| 관리자 메인 페이지 |
| 공지사항 버튼 클릭 시 리스트를 보여준다.  제목을 클릭 시 상세보기 페이지로 이동한다. |

|  |
| --- |
|  |

|  |
| --- |
| 관리자 페이지 문의사항 |
| 공지사항버튼을 누르면 등록 화면으로 이동 후 에디터를 사용하여 공지사항을 등록할 수 있다. |

|  |
| --- |
|  |

|  |
| --- |
| 관리자 페이지 문의사항 |
| 리스트에서 제목 클릭 시 공지 상세보기 페이지로 이동하여 확인할 수 있다. |

**(44) 문의**

|  |
| --- |
|  |

|  |
| --- |
| 관리자 페이지 문의 리스트 |
| 문의 버튼을 클릭 시 문의 리스트로 이동하여 목록을 볼 수 있다. |

|  |
| --- |
|  |

|  |
| --- |
| 관리자 페이지 문의 상세보기 |
| 제목을 클릭 시 상세보기 페이지로 이동하여 답변을 남길 수 있다.  답변이 없을 경우 답변하기를 누르면 스크립트를 활용하여 text박스가 만들어진다.  답변등록을 누르면 ajax를 활용하여 DB등록 후 등록된 답변을 다시 가져와  스크립트로 text박스가 span테그로 바꾸어 화면이동 없이 출력해 준다. |

■Script : script를 사용하여 답변 등록 후 다시 가져온다.

|  |
| --- |
| <scriptth:inline=*"javascript"*>  $(document).ready(**function**(){  **var** ddd = '<table><tr><td><button class="btn btn-primary d-block" id="commentUpdateDetail" style="margin-top: 20px;">답변수정</button></td>'  +'<td><button type="button" class="btn btn-danger d-block" style="margin-top: 20px;">'  +'<a th:href="@{/admin/inquiryCommentDelete(boardCCode=${board.boardComment.boardCCode},boardCode=${board.boardCode})}" style="color: white">'  +'<span class="ion ion-md-cut">삭제하기</span>'  +'</a></td></tr></button></table>'  $(document).on('click','#commetBtn',**function**(){  **var** commentText = $('#comment').html();  **var** commentTrim = commentText.trim()  console.log(commentTrim);  **if**(commentTrim == **null** || commentTrim == ""){  $('#comment').html('<br><br><br><hr style="border-color: #dcdcdc;">'  +'<h3>문의답변</h3><div><span style="color: #A2A4A6;" id="uId">관리자 : [[${session.SAID}]]</span></div>'  +'<br><div id="commentContent"><div id="boardCContent" contenteditable style="height: 300px; width: 100%; border: 1px solid silver;"></div></div>'  +'<br><hr style="border-color: #dcdcdc;">'  +'<div id="Btn"><button class="btn btn-primary d-block" id="commentInsert" style="margin-top: 20px;">답변등록</button></div>');  $(document).on('click','#commentInsert',**function**(){  console.log('dz');  **var** boardCContent = $('#boardCContent').html();  console.log(boardCContent)  **var** boardCode = $('#boardCode').val();  console.log(boardCode)  **var** request = $.ajax({  url: "/admin/inquiryCommentInsert",  method: "POST",  data: { "boardCContent" : boardCContent , "boardCode" : boardCode},  dataType: "json"  });    request.done(**function**( data ) {  console.log(data);  **var** boardCCode = data.boardCCode;  console.log("1"+boardCCode)  $('#commentContent').html("");  $('#commentContent').html(data.boardCContent);  $('#commentInsert').remove();  $('#Btn').html('<button class="btn btn-primary d-block" id="commentUpdate" style="margin-top: 20px;">수정하기</button>');  $(document).on('click','#commentUpdate',**function**(){  **var** commentContentHtml = $('#commentContent').html();  console.log(commentContentHtml);  $('#commentContent').html("");  $('#commentContent').html('<div id="boardCContent" contenteditable style="height: 300px; width: 100%; border: 1px solid silver;">'+commentContentHtml+'</div>');  $('#commentInsert').remove();  $('#Btn').html('<button class="btn btn-primary d-block" id="commentUpdatePut" style="margin-top: 20px;">수정등록</button>');    $(document).on('click','#commentUpdatePut',**function**(){  console.log('commentUpdatePut')  **var** boardCContentHtml = $('#boardCContent').html();  **var** request = $.ajax({  url: "/admin/inquiryCommentUpdate",  method: "POST",  data: { "boardCContent" : boardCContentHtml , "boardCCode" : boardCCode, "boardCode" : boardCode },  dataType: "json"  });  request.done(**function**( commentUpdate ) {  boardCContentHtml = $('#boardCContent').html();  console.log('update값 : '+commentUpdate.boardCContent);  $('#commentContent').html("");  $('#commentContent').html('<div id="boardCContent">'+commentUpdate.boardCContent+'</div>');  $('#Btn').html('<button class="btn btn-primary d-block" id="commentUpdate" style="margin-top: 20px;">수정하기</button>');    });  request.fail(**function**( jqXHR, textStatus ) {  alert( "Request failed: " + textStatus );  });  })  })  })  request.fail(**function**( jqXHR, textStatus ) {  alert( "Request failed: " + textStatus );  });  })  }**else**{  alert('이미답글이달린문의입니다.')  }  })  $(document).on('click','#commentUpdateDetail',**function**(){  console.log('dzdz')  **var** commentContentHtml = $('#commentContent').html();  console.log(commentContentHtml);  $('#commentContent').html("");  $('#commentContent').html('<div id="boardCContent" contenteditable style="height: 300px; width: 100%; border: 1px solid silver;">'+commentContentHtml+'</div>');  $('#commentUpdateDetail').remove();  $('#Btn').html('<button class="btn btn-primary d-block" id="commentUpdatePutDetail" style="margin-top: 20px;">수정등록</button>');    $(document).on('click','#commentUpdatePutDetail',**function**(){  console.log('commentUpdatePutDetail')  **var** boardCode = $('#boardCode').val();  **var** boardCCode = $('#boardCCode').val();  **var** boardCContentHtml = $('#boardCContent').html();  console.log(boardCContentHtml);  console.log(boardCCode);  console.log(boardCode);  **var** request = $.ajax({  url: "/admin/inquiryCommentUpdate",  method: "POST",  data: { "boardCContent" : boardCContentHtml , "boardCCode" : boardCCode, "boardCode" : boardCode },  dataType: "json"  });  request.done(**function**( commentUpdate ) {  boardCContentHtml = $('#boardCContent').html();  console.log('update값 : '+commentUpdate.boardCContent);  $('#commentContent').html("");  $('#commentContent').html('<div id="boardCContent">'+commentUpdate.boardCContent+'</div>');  $('#Btn').html('<button class="btn btn-primary d-block" id="commentUpdateDetail" style="margin-top: 20px;">수정하기</button>');    });  request.fail(**function**( jqXHR, textStatus ) {  alert( "Request failed: " + textStatus );  });  })  })  })  </script> |

**(45) 추천도서**

|  |
| --- |
|  |

|  |
| --- |
| 관리자 페이지 추천도서 |
| 추천도서 버튼을 클릭 시 리스트를 볼 수 있다. |

|  |
| --- |
|  |

|  |
| --- |
| 관리자 페이지 추천도서 상세보기 |
| 리스트에서 제목을 클릭 시 상세보기 페이지로 이동한다.  수정하기 버튼을 누를 시 서평과 추천인을 수정할 수 있다. |

|  |
| --- |
|  |

|  |
| --- |
| 관리자 페이지 추천도서 등록 |
| 리스트에서 추천도서 등록 버튼을 클릭 시 등록 페이지로 이동한다.  등록 페이지에서 ISBN도서코드를 입력하면 ajax 활용하여 책의 정보들이 출력된다.  서평, 추천인, 공개여부를 입력 등록할 수 있다. |

**(46) 공공시설 등록 리스트**

|  |
| --- |
|  |

■javaScript : 라디오버튼 클릭시 해당되는 값의 리스트를 출력해준다.

|  |
| --- |
| <scriptth:inline=*"javascript"*>  $(document).ready(**function**(){  onload = **function**(){ // 페이지가로드될때실행  **var**nowKinds = [[${nowKinds}]]; // 라디오버튼을클릭후 model에담겨온시설종류  **for**(**var**i = 0; i< $('.kRadio').length; i++){ // 라디오버튼의길이만큼반복  **if**($('.kRadio')[i].value == nowKinds){ // 라디오버튼의i번째 value가넘어온시설명과같으면  $('.kRadio')[i].checked = **true**; // 체크해준다.  }**else**{ // 라디오버튼의i번째 value가넘어온시설명과같지않으면  $('.kRadio')[i].checked = **false**; // 체크를해제한다.  }  }  }  $(document).on('click','.kRadio',**function**(){ //라디오버튼클릭시  **var**fKinds = $(**this**).val(); //누른라디오버튼의 value를fKinds에저장  **if**(fKinds == '전체'){ //value가 '전체'이면  location.href = '/admin/facilityList'; //fKinds가없이경로로보내줌. (컨트롤러에서전체리스트가져옴)  }**else**{ //value가 '전체가아니면  location.href = '/admin/facilityList?fKinds='+fKinds;  //fKinds를 get방식으로보내줌. (컨트롤러에서받은종류의리스트를가져옴)  }  })  })  </script> |

■SQL : 선택하지 않은 경우는 기본적으로 ‘전체’값을 가진다.

라디오버튼 클릭시 해당되는 값의 리스트를 출력해준다.

|  |
| --- |
| <!--시설종류가 xx 인것의리스트를출력 -->  <selectid=*"getFacilityList"*parameterType=*"String"*resultMap=*"FResultMap"*>  SELECT  \*  FROM  facility  WHERE  l\_code = #{libNum}  <iftest=*"fKinds != '전체'"*>  AND f\_kinds = #{fKinds}  </if>  </select>  <!--시설종류가 XX인것의예약리스트출력 -->  <selectid=*"getFacilityReservation"*parameterType=*"String"*resultMap=*"reserveResultMap"*>  SELECT  fr.\*,  f.f\_kinds  FROM  facility\_reservation AS fr INNER JOIN facility AS f  ON fr.f\_code = f.f\_code  WHERE  fr.l\_code = #{libNum}  <iftest=*"fKinds != '전체'"*>  AND f.f\_kinds = #{fKinds}  </if>  </select> |

|  |
| --- |
| 사서채널 – 공공시설 리스트 페이지 |
| DB에 등록된 공공시설을 전부 보여줍니다.  라디오버튼을 클릭시 해당 값으로 쿼리문을 실행하여 조건에 맞는 리스트를 출력해줍니다.  공공시설 이름 등으로 검색하여 시설을 찾을 수 있습니다.  시설 추가 버튼을 누르면 시설 추가 페이지로 이동합니다. |

**(47) 공공시설 등록 페이지**

|  |
| --- |
| [그림 1]  공공시설 등록 페이지를 진입 후,시설종류를 변경합니다. |
| [그림 2]  스터디룸/강연실선택시 위의 폼이 나오게 됩니다. | | |
| [그림 3]  열람실 /사물함 선택시 위의 폼이 나오게 되며,  크기에 맞는 좌석/자리를 생성할 수 있는 input 박스가 2개 나오게 됩니다. | | |
| [그림 3-1]  가로세로 입력후 테이블 만들기 버튼을 클릭하면 입력한 가로 \* 세로만큼의 테이블이 동적으로 생성되고,  자리를 선택하여 사용할 수 있는 자리를 사서가 만들어 줄 수 있습니다. | |
| [그림 3-2]  이미 선택된 자리를 누르게 되면 confirm창이 나오며,확인을 누르면 선택한 자리를 해제할 수 있습니다.    [그림 3-3]  자리를 해제하게 되면 번호가 앞으로 당겨져 배열에 저장됩니다.    [그림 3-4]  자리 선택이 다 끝났으면,자리선택완료 버튼을 눌러줍니다.  누르게 되면 위쪽의 폼에 선택된 좌석의 seat속성이 문자열 한 줄로 자동 입력되고,  가로,세로가 입력됩니다.  등록하기를 누르면 DB에 폼에 입력된 것들이 저장됩니다. | |

■javaScript :테이블을 동적으로 만들어주는 ajax, 선택한 자리를 검정색으로 칠해주는 코드

|  |
| --- |
| <scriptth:inline=*"javascript"*>  $(document).ready(**function**(){  **var** seatArray = **new**Array();    $(document).on('click','.tdClass',**function**(){  //console.log($(this).attr('status') + " <== status")  //console.log($(this).attr('seat') + " <== seat")  **var** tdClass = $('.tdClass'); // tdClass에선택자넣기  **if**($(**this**).attr('status') == 'true'){  seatArray.push($(**this**).attr('seat')) // status의상태가 true이면선택한곳의 'seat'를배열의맨뒤쪽에넣는다.  $(**this**).css("background-color", 'black'); // 선택한곳의배경색을검정으로바꾼다.  $(**this**).text(seatArray.indexOf($(**this**).attr('seat'))+1) // 선택한곳의 text를배열의인덱스(0)+1로변경한다 (1,2,3,4...)  $(**this**).css("color",'white'); // 선택한곳의글자색을화이트로변경  $(**this**).attr('status','false'); // 선택한곳의 status속성을 false로변경    }**else if**($(**this**).attr('status') == 'false'){  **var** con\_test = confirm("이미선택된자리입니다. "+  "[선택자리 : '"+$(**this**).attr('seat')+"'] \n제거하려면확인을누르세요.");  **if**(con\_test == **true**){ //console.log('확인누름'); seatArray.splice(seatArray.indexOf($(**this**).attr('seat')),1); //선택된배열의인덱스부터 1개를지움  $(**this**).css("background-color", 'white'); // 누른곳의배경색을하얀색으로바꾼다.  $(**this**).text($(**this**).attr('seat')); // 누른곳의 text를누른곳의 seat속성으로변경한다.  $(**this**).css("color",'black'); // 누른곳의글자색을블랙으로변경  $(**this**).attr('status', 'true'); // 누른곳의 status속성을 true로변경  /\* 배열의 index에따른선택된값의 text 재배치 \*/    **for**(**var** i = 0 ;i<tdClass.length; i++){ // td 엘리먼트만큼반복 (테이블갯수만큼반복)  **if**(tdClass.eq(i).attr('status') == 'false'){ // i번째의 status속성이 false인것  //console.log(tdClass.eq(i).attr('seat') + '<== false인것의값'); // false인것의 eq(i)번째의 seat속성값.  //console.log(tdClass.eq(i).attr('seat')+' '+seatArray.indexOf(tdClass.eq(i).attr('seat')) + '<== false인, seatArray의인덱스')  tdClass.eq(i).text(seatArray.indexOf(tdClass.eq(i).attr('seat'))+1)  // tdClass.eq(i)번째의 text를seatArray배열의tdClass.eq(i)번째의속성이'seat'인것의위치(index)+1로바꾼다.  }  }  } **elseif**(con\_test == **false**) {  //console.log('취소누름');  }    }  //console.log(seatArray);  })    //ajax로테이블만들기.  $(document).on('click','#tableBtn',**function**(){  seatArray = **new** Array(); // 테이블만들기버튼을다시누르면새로운배열이생성됨.  **var** garo = $('#garo').val();  **var** sero = $('#sero').val();    **var** request = $.ajax({  url: "/libSeat",  method: "POST",  data: { 'garo' : garo, 'sero':sero },  dataType: "json"  });    request.done(**function**( data ) {  **var** getGaro = data.getGaro; // 가로배열(리스트)  **var** getSero = data.getSero; // 세로배열(리스트)  **var** tableDiv = $('#tableDiv');  **var** contents = '';    contents += '<table border="2">';  **for**(**var**i = 1; i<= getSero; i++){  contents += '<tr>';  **for**(**var** j = 1; j <= getGaro; j++){  contents += '<td class="tdClass" style="color:black; text-align: center; font-size:12px; width: 40px; height: 35px;"'+  'seat="'+i+'-'+j+'" status="true">'+i+'-'+j+'</td>';  }  contents += '</tr>';  }  contents += '</table>';  contents += '<small class="form-text text-muted">만들어줄자리을선택후선택완료버튼을눌러주세요.</small>'  contents += '<br><button type = "button" id = "seatMakeBtn" class="btnbtn-primary">자리선택완료</button>'  tableDiv.html(contents); // 만들어진테이블을뿌려줌  });    request.fail(**function**( jqXHR, textStatus ) {  alert( "값을입력후버튼을눌러주세요." );  });  })  //만들어진자리배열 ajax를통해컨트롤러로보내기  $(document).on('click','#seatMakeBtn',**function**(){  **if**(seatArray.length> 0){ // 배열이 1개이상있어야 ajax 실행  **var** dataMap = {'seatArray' :seatArray}    **var** request = $.ajax({  url: "/getArray",  type: "post",  traditional :**true**,  data: dataMap,  dataType :"text"  });    request.done(**function**( data ) {  $('#seatSelected').val(data)  $('#fRow').val($('#garo').val())  $('#fCol').val($('#sero').val())  });    request.fail(**function**( jqXHR, textStatus ) {  alert( "Request failed: " + textStatus );  });  } **else** {  alert('자리를한곳이상선택해주세요')  }  })  })  </script> |

■Controller :자리 선택 후 배열을 문자열로 만들어 전달해주는 메서드

|  |
| --- |
| /\*\*  \* ajax활용. form에입력될배열을문자열로뿌려줌.  \*  \* **@param**seatArray  \* **@return**  \*/  @PostMapping("/getArray")  **public**@ResponseBody String getArray(@RequestParam(value = "seatArray") List<String>seatArray) {  System.***out***.println(seatArray + " <== seatArray");  String array = "";  **for** (**int**i = 0; i<seatArray.size(); i++) {  **if** ((i + 1) != seatArray.size()) {  array += seatArray.get(i) + ",";  } **elseif** ((i + 1) == seatArray.size()) {  array += seatArray.get(i);  }  }  System.***out***.println(array);  **return**array;  } |

■SQL : 폼에 입력된 내용을 가지고 DB에 insert하는 쿼리문을 실행

|  |
| --- |
| <!--공공시설등록하기. -->  <insertid=*"insertFacility"*>  <selectKeykeyProperty=*"fCode"*resultType=*"String"*order=*"BEFORE"*>  SELECT  CONCAT('f-',DATE\_FORMAT(CURDATE(), '%y%m%d'),  (IF(DATE\_FORMAT(CURDATE(),'%y%m%d') > LEFT(SUBSTRING\_INDEX(MAX(f.f\_code), '-', -1),6)  , '00001'  ,LPAD(RIGHT(MAX(f.f\_code),5)+1,5,'0'))))  FROM  facility AS f  </selectKey>  INSERT INTO facility  (f\_code, l\_code, u\_id, f\_name, f\_kinds, f\_location, f\_accommodate, f\_gender, f\_term, f\_row, f\_col, f\_seat\_or\_img, f\_date)  VALUES (#{fCode}, #{lCode}, #{uId}, #{fName}, #{fKinds}, #{fLocation}, #{fAccommodate}, #{fGender}, #{fTerm}, #{fRow}, #{fCol}, #{fSeatOrImg}, NOW())  </insert> |

|  |
| --- |
| 사서채널 – 공공시설 등록 페이지 |
| 옵션에 있는 공공시설을 선택하여,시설명,사용인원등 설정을 해준 후 시설을 DB에 등록할 수 있습니다.  Ajax를 사용하여 사용자가 원하는 만큼의 자리를 동적으로 생성할 수 있습니다.  자리를 선택하여 사용가능한 자리를 만들 수 있습니다. |

**(48) 공공시설 상세 페이지**

|  |
| --- |
|  |

■SQL : 선택된 시설의 코드를 조건으로 하여 시설을 select하는 쿼리문 실행

|  |
| --- |
| <!--fCode 1개를조회하여상세정보출력 -->  <selectid=*"getFacility"*parameterType=*"String"*resultMap=*"FResultMap"*>  SELECT  \*  FROM  facility  WHERE  f\_code=#{fCode} AND l\_code= #{libNum}  </select> |

■SQL : 삭제버튼 클릭시 해당 시설의 정보를 DB에서 delete하는 쿼리문 실행

|  |
| --- |
| <!--공공시설을삭제해주는쿼리 -->  <deleteid=*"deleteFacility"*parameterType=*"String"*>  DELETE FROM  facility  WHERE  f\_code=#{fCode}  </delete> |

|  |
| --- |
| 사서채널 – 공공시설 상세 페이지 |
| 등록된 시설의 상세 정보를 확인할 수 있습니다.  수정버튼을 눌러 해당 시설의 정보를 수정할 수 있습니다.  삭제버튼을 눌러 해당 시설을 DB에서 제거할 수 있습니다. |

**(43) 사서채널 – 공공시설 수정 페이지**

|  |
| --- |
|  |

■SQL : 수정하기 버튼을 누르면 수정된 내용으로 update되는 쿼리문 실행

|  |
| --- |
| <!--공공시설수정하기 / 열람실,사물함 / 스터디룸,강연실다르게해줘야함. fCode를받아서처리-->  <updateid=*"updateFacility"*parameterType=*"ksmart.pentagon.vo.Facility"*>  UPDATE  facility  SET  f\_name= #{fName},  f\_location= #{fLocation},  f\_accommodate= #{fAccommodate},  f\_gender= #{fGender},  f\_row= #{fRow},  f\_col= #{fCol},  f\_seat\_or\_img= #{fSeatOrImg}  WHERE  f\_code = #{fCode}  </update> |

|  |
| --- |
| 사서채널 – 공공시설 수정 페이지 |
| DB에 등록된 시설의 상세 정보를 수정할 수 있습니다.  열람실/사물함의 경우 좌석을 다시 선택후 자리선택완료 버튼을 눌러 수정 할 수 있습니다.  테이블 만들기 버튼을 다시 클릭할 경우 ,선택된 자리가 전부 해제됩니다. |

**(49) 공공시설 신청 내역 리스트 페이지**

|  |
| --- |
|  |

■SQL :공공시설 신청 내역 리스트 페이지진입시, 공공시설 신청 리스트가 출력되는 쿼리문 실행

|  |
| --- |
| <!--시설종류가 XX인것의신청내역리스트출력 -->  <selectid=*"getFacilityReservation"*parameterType=*"String"*resultMap=*"reserveResultMap"*>  SELECT  fr.\*,  f.f\_kinds  FROM  facility\_reservation AS fr INNER JOIN facility AS f  ON fr.f\_code = f.f\_code  WHERE  fr.l\_code = #{libNum}  <iftest=*"fKinds != '전체'"*>  AND f.f\_kinds = #{fKinds}  </if>  </select> |

|  |
| --- |
| 사서채널 – 공공시설 신청 내역 리스트 페이지 |
| DB에 등록된 회원들의 공공시설 신청 내역을 확인할 수 있습니다.  시설 종류별로 리스트 출력 및 검색이 가능합니다. |

**(50) 프로그램 리스트 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 사서채널 –프로그램 리스트 페이지 |
| DB에 등록된 프로그램리스트를 볼 수 있습니다.  상세버튼을 통해 선택한 프로그램의 상세 정보를 확인할 수 있습니다.  프로그램 추가버튼을 클릭하여 새로운 프로그램 및 행사를 등록할 수 있습니다.  검색 기능을 통해 등록된 프로그램을 찾을 수 있습니다. |

**(51 )프로그램 등록 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 사서채널 – 프로그램 등록 페이지 |
| 등록하기 버튼을 눌러 입력한 폼의 내용을 DB에 저장 할 수 있습니다. |

**(52) 프로그램 상세정보 페이지**

|  |
| --- |
|  |

|  |
| --- |
| 사서채널 – 프로그램 상세보기 페이지 |
| DB에 등록되어 있는 프로그램의 상세정보를 확인할 수 있습니다. |

**(53) 프로그램 수정 페이지**

|  |
| --- |
|  |

■SQL :입력된 정보로 프로그램의 상세 정보를 수정하는 쿼리문 실행

|  |
| --- |
| <!--사서채널 / 프로그램수정하기 -->  <updateid=*"updateProgram"*parameterType=*"ksmart.pentagon.vo.ProgramManager"*>  UPDATE  program\_mgr  SET  u\_id= #{uId},  pm\_name= #{pmName},  pm\_target= #{pmTarget},  pm\_place= #{pmPlace},  pm\_start\_receipt= #{pmStartReceipt},  pm\_end\_receipt= #{pmEndReceipt},  pm\_start\_term= #{pmStartTerm},  pm\_end\_term= #{pmEndTerm},  pm\_start\_time= #{pmStartTime},  pm\_end\_time= #{pmEndTime},  pm\_teacher\_name= #{pmTeacherName},  pm\_mnop= #{pmMnop},  pm\_tuition= #{pmTuition},  pm\_tel= #{pmTel},  pm\_detail= #{pmDetail}  WHERE  pm\_code= #{pmCode}  </update> |

|  |
| --- |
| 사서채널 – 프로그램 수정 페이지 |
| DB에 등록되어 있는 프로그램의 상세정보를 수정 할 수 있습니다. |

**7. 프로젝트 후기/보완해야할사항**