**처음에 리스트페이지 들어와서 전부 불러올 때**

**-----------------------------------------------------------------------------------**

**1.**

/\*\*  
  \* 시스템 > 회원관리 > 전체 회원 조회  
\*  
  \* @param model **spring ui model**  
  \* @return 전체 회원 조회 페이지  
\*/  
 @GetMapping("/viewMemberList")  
 public String viewMemberList(MemberSearchDto.Req req, Model model) {  
     PageInfo<MemberSearchDto.Res> pagingList = null;  
     List<CmmnCodeDto> authCodeList = authService.findAuthCode(false);  
     long allCount = memberService.countAllMember();  
  
     if (req.getSbsceDtFrom() != null)  
        pagingList = new OcPageInfo<>(memberService.findMemberForUi(req));  
  
     model.addAttribute("req", req);  
     model.addAttribute("pagingList", pagingList);  
     model.addAttribute("authCodeList", authCodeList);  
     model.addAttribute("allCount", allCount);  
  
     return "/sys/mbrm/member/viewMemberList";  
 }

**-----------------------------------------------------------------------------------**

**2.**

/\*\*  
  \* 권한 정보 코드 리스트 조회.  
\*  
  \* @param **includeSpringInnerAuth 스프링 내부 권한 포함 여부**  
  \* @return 권한 정보 코드 리스트  
\*/  
 public List<CmmnCodeDto> findAuthCode(boolean includeSpringInnerAuth) {  
     return authManageMapper.selectAuthCodeList(includeSpringInnerAuth?"Y":"N");  
 }

**-----------------------------------------------------------------------------------**

**3.**

 public interface AuthManageMapper {  
  
     @DataSourceSelector(name = DatasourceType.AUTH)  
     List<CmmnCodeDto> selectAuthCodeList(String spriInnerAthryYn);

**-----------------------------------------------------------------------------------**

**4.**

public long countAllMember() {  
     return usrUspsMapper.countAllMember();  
 }

 @DataSourceSelector(name = DatasourceType.AUTH)  
 long countAllMember();

**-----------------------------------------------------------------------------------**

**5.**

/\*\*  
  \* 회원 관리 UI용 리스트 조회  
\*  
  \* @return 회원 리스트  
\*/  
 public List<MemberSearchDto.Res> findMemberForUi(MemberSearchDto.Req req) {  
     // UI에서 넘어온 페이지 번호와 페이지 사이즈로 페이징 처리 적용  
     PageHelper.startPage(req.getPageNo(), req.getPageSize());  
  
     // 페이징 처리시 자동으로 ORDER BY를 넣어준다.  
     PageHelper.orderBy("USPS\_SQNO DESC");  
  
     return usrUspsMapper.selectMemberForUi(req);  
 }

**-----------------------------------------------------------------------------------**

**6.**

 @DataSourceSelector(name = DatasourceType.AUTH)  
 Page<MemberSearchDto.Res> selectMemberForUi(MemberSearchDto.Req req);

**-----------------------------------------------------------------------------------**

**7.**

/\*\*  
\* @Method : getMemberInfoById  
  \* @Descryption : 로그인 사용자 정보조회  
\* @Author : eple  
\* @Date : 2021. 3. 3.  
\* @History :  
  \* 2021. 3. 3.  eple     최초 작성  
\* @return  
\*/  
 @GetMapping  
@ResponseBody  
 public ApiResponseDto<MemberInfo> getMemberInfoById() {  
     return ApiResponseDto.success(memberService.getAccessMember());  
 }

**-----------------------------------------------------------------------------------**

**8.**

/\*\*  
\* @return  
\* @Method : getAccessMember  
  \* @Descryption : 인증 사용자 정보  
\* @Author : eple  
\* @Date : 2021. 1. 15.  
  \* @History : 2021. 1. 15. eple   최초 작성  
\*/  
 public MemberInfo getAccessMember() {  
     Authentication auth =  
             Optional.ofNullable(SecurityContextHolder.getContext().getAuthentication())  
                     .orElseThrow(AccountNotFoundException::new);  
     return ((OcUserDetails) auth.getPrincipal()).getMembers();  
 }

**-----------------------------------------------------------------------------------**

**1.**

/\*\*  
  \* @Method : getMemberInfoById  
  \* @Descryption : 로그인 사용자 정보조회  
  \* @Author : eple  
  \* @Date : 2021. 3. 3.  
  \* @History :  
  \* 2021. 3. 3.  eple    최초 작성  
  \* @return  
  \*/  
 @GetMapping  
 @ResponseBody  
 public ApiResponseDto<MemberInfo> getMemberInfoById() {  
     return ApiResponseDto.success(memberService.getAccessMember());  
 }

**-----------------------------------------------------------------------------------**

**2.**

|  |  |
| --- | --- |
| /\*\* 검색 조건.   \*   \* 아이디 검색 - 1 (기본값)   \* 회원이름 - 2   \* 이메일 - 3   \* 핸드폰번호 - 4   \*/  private String searchCondition = "1";  /\*\* 검색 조건에 따른 검색 값 \*/  private String searchValue = ""; | /\*\* 관리자 권한 조회 조건 \*/      private HashSet<String> authCodes = new HashSet<>();        /\*\* 회원구분코드       \*       \* 001 : 일반       \* 002 : 학생       \* 003 : 교사       \* 004 : 순회교사       \* 005 : 방과후강사       \* 006 : 인증만료       \*/      private List<String> mmbrDtlTrgts;  } |
| /\*\*   \* 회원관리 UI 검색조건 클래스.   \*   \* @author Keunhak Lee   \* @version 1.0.0   \* @since 2021-03-26   \* @see com.ebs.oc.admin.cmmn.dto.PageableSearchDto 페이징 검색 적용시 상속   \*/  @Data  @AllArgsConstructor  @NoArgsConstructor  public static class Req extends PageableSearchDto {      /\*\* 가입기간 from \*/      private String sbsceDtFrom;        /\*\* 가입기간 to \*/      private String sbsceDtTo;        /\*\* 체크박스 검색의 회원구분 체크된 값들을 입력받을 리스트 \*/      private List<String> uspsSeCd; | /\*\* 검색 조건.   \*   \* 아이디 검색 - 1 (기본값)   \* 회원이름 - 2   \* 이메일 - 3   \* 핸드폰번호 - 4   \*/  private String searchCondition = "1";    /\*\* 검색 조건에 따른 검색 값 \*/  private String searchValue = "";    /\*\*   \*   회원구분 처리에 따른 신분코드 리스트 조회   \*   HashSet 선언으로 코드가 중복되지 않는 컬렉션으로 선언한다.   \*/  private HashSet<String> scpstCdList = new HashSet<>(); |
| /\*\* 학생 체크 여부 \*/  private String isStudent;    /\*\*   \* 학생여부 값이 Y로 들어오면 신분코드에 학생 검색조건을 추가한다.   \* @param isStudent 학생여부 체크 (="Y")   \*/  public void setIsStudent(String isStudent) {      this.isStudent = isStudent;        if (isStudent != null && isStudent.equals("Y")) {          scpstCdList.add("001"); // 조건에 유아 추가          scpstCdList.add("002"); // 조건에 초등학생 추가          scpstCdList.add("003"); // 조건에 중학생 추가          scpstCdList.add("004"); // 조건에 고등학생 추가          scpstCdList.add("005"); // 조건에 대학생 추가      }  }    /\*\* 선생님 체크 여부 \*/  private String isTeacher;    /\*\*   \* 선생님여부 값이 Y로 들어오면 신분코드에 선생님 검색조건을 추가한다.   \* @param isTeacher 선생님여부 체크 (="Y")   \*/  public void setIsTeacher(String isTeacher) {      this.isTeacher = isTeacher;        if (isTeacher != null && isTeacher.equals("Y")) {          scpstCdList.add("006"); // 조건에 선생님 추가      }  } |  |
|  |  |
|  |  |
|  |  |
|  |  |

권한관리 > 신규권한등록버튼 클릭

권한등록페이지

1. @GetMapping("/isAuthExist")

저장하겠습니까?

2. /\*\*  
  \* 시스템관리 > 회원관리 > 권한 관리 > 권한 등록.  
  \*  
  \* @param req          권한 등록 정보  
  \* @param redirectAttr 한번 쓰고 버려지는 어트리뷰트.  
  \* @return 권한 관리 > 권한 등록 페이지  
  \*/  
 @PostMapping("/createAuth")

3등록되었습니다알림창-권한조회페이지로 다시 리다이렉트

@GetMapping("/viewAuthList")

1. 권리권한명 칼럼에 있는  하이퍼텍스트 링크 클릭
2. @GetMapping("/viewAuthDetail")

1. sys/mbrm/auth/viewAuthDetail로 이등

ex) <http://localhost:58080/sys/mbrm/auth/viewAuthDetail?athryCd=ROLE_ADMIN>

1. 권한정보상세보기 > 탭 권한 관리자 관리

<http://localhost:58080/sys/mbrm/auth/viewAuthDetail?athryCd=ROLE_ADMIN&activeTab=1>

1. 신규 관리자 추가

1. 신규 관리자 추가 >아이디 검색
2. @GetMapping("/findAuthAdminCandidateList")

1. 신규 관리자 추가 >아이디 저장 (해당 회원을 관리자로 추가하시겠습니까? 알림창)

/\*\*  
  \* 권한별 관리자 추가.  
  \*  
  \* @param req 회원 검색 조건  
  \* @return 권한상세 페이지 이동  
  \*/  
 @PostMapping("/createAuthAdminList")

저장되었습니다.> @GetMapping("/viewAuthDetail")로 이동

"adminList" -> {ArrayList@18950}  size = 6

key = "adminList"

value = {ArrayList@18950}  size = 6

  0 = {AuthAdminListDto$Res@18953} "AuthAdminListDto.Res(scrtyEstbsSqno=4, uspsId=gsi\*\*\*\*\*, uspsNm=G\*\*\*\*\*\*\*1, scrtyBgnDt=2021-04-27, ofecCd=, ofecCdNm=전체, ofecCmpteCd=, ofecCmpteCdNm=전체)"

  1 = {AuthAdminListDto$Res@18954} "AuthAdminListDto.Res(scrtyEstbsSqno=6, uspsId=gsi\*\*\*\*\*, uspsNm=G\*\*\*\*\*\*\*3, scrtyBgnDt=2021-04-27, ofecCd=, ofecCdNm=전체, ofecCmpteCd=, ofecCmpteCdNm=전체)"

  2 = {AuthAdminListDto$Res@18955} "AuthAdminListDto.Res(scrtyEstbsSqno=26, uspsId=sic\*\*\*\*, uspsNm=S\*\*\*\*\*\*5, scrtyBgnDt=2022-02-25, ofecCd=, ofecCdNm=전체, ofecCmpteCd=, ofecCmpteCdNm=전체)"

  3 = {AuthAdminListDto$Res@18956} "AuthAdminListDto.Res(scrtyEstbsSqno=67, uspsId=sic\*\*\*\*, uspsNm=온\*\*사, scrtyBgnDt=2022-07-27, ofecCd=, ofecCdNm=전체, ofecCmpteCd=, ofecCmpteCdNm=전체)"

  4 = {AuthAdminListDto$Res@18957} "AuthAdminListDto.Res(scrtyEstbsSqno=103, uspsId=sic\*\*\*\*, uspsNm=쌍\*\*\*1, scrtyBgnDt=2022-08-29, ofecCd=, ofecCdNm=전체, ofecCmpteCd=, ofecCmpteCdNm=전체)"

  5 = {AuthAdminListDto$Res@18958} "AuthAdminListDto.Res(scrtyEstbsSqno=155, uspsId=sic\*\*\*\*, uspsNm=쌍\*\*\*1, scrtyBgnDt=2024-06-25, ofecCd=GG, ofecCdNm=경기, ofecCmpteCd=GG02, ofecCmpteCdNm=경기도고양교육지원청)"