| 기능명세서 | 문서번호 |  |
| --- | --- | --- |
| 작성자 | 1조 |
| 작성일자 | 21.08.05 |

* **package com.project.rural**

| **구분** | **이름** | **역할** |
| --- | --- | --- |
| **패키지** | **package com.project.rural** | **비회원, 회원 접속** |
| **클래스** | **Index** | **메인화면** |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural** | **개발자** |
| --- | --- | --- |
| **클래스** | **DBUtil** | **DB 연동** |
| **메소드** | Connection open() | 등록한 jdbc 오라클 계정에 접속 |

| **패키지** | **package com.project.rural** | **개발자** |
| --- | --- | --- |
| **클래스** | **EncodingFilter** | **encoding 자동 지정** |
| **메소드** | init() | encoding 초기화 |
| doFilter() | 미리 등록한 encoding으로 필터링 |

| **패키지** | **package com.project.rural** | **개발자** |
| --- | --- | --- |
| **클래스** | **CheckMember** | **로그인이 필요한 페이지별 접속 시 팝업창 알림** |
| **메소드** | check() | 비로그인 / 로그인 사용자를 session으로 검사해서 팝업창 알림 |

* **package com.project.rural.admin**

| **패키지** | **package com.project.rural.admin** | **관리자** |
| --- | --- | --- |
| **클래스** | **ChartDAO()** | **홈페이지 통계(차트)를 위한 DB에 접속 후 제공하는 클래스** |
| **메소드** | ChartDAO() | DBUtil 클래스를 호출하여 DB 접속 |
| ArrayList<ChartDTO> list1() | 사용자의 이름, 작성글 수 데이터를 불러오는 메소드 |
| ChartDTO list2() | 카테고리별 이름, 게시글 수 데이터를 불러오는 메소드 |
| ChartDTO list3() | 사용자의 상태(정상/정지/탈퇴) 데이터를 불러오는 메소드 |
| ChartDTO list4() | 사용자의 유형(일반/농부) 데이터를 불러오는 메소드 |

| **패키지** | **package com.project.rural.admin** | **관리자** |
| --- | --- | --- |
| **클래스** | **ChartDTO()** | **홈페이지 통계(차트)를 위한 getter & setter 클래스** |
| **변수** | String name | 이름 |
| String cnt | 개수 |
| String community | 커뮤니티 |
| String exp | 농촌체험 |
| String farm | 주말농장 |
| String market | 농작물직거래 |
| String notice | 알림마당 |
| String normal | 정상 |
| String out | 탈퇴 |
| String stop | 정지 |
| String ordinary | 일반 사용자 |
| String farmer | 농업인 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.admin** | **관리자** |
| --- | --- | --- |
| **클래스** | **Manage** | **홈페이지 관리** |
| **변수** | CheckMember | 로그인 여부 확인 |
| ChartDAO dao | ChartDAO 클래스 객체 생성 |
| ChartDTO list2, 3, 4 | ChartDTO 클래스 객체 생성 |
| ArrayList<ChartDTO> list1 | ArrayList<ChartDTO> 클래스 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.admin** | **관리자** |
| --- | --- | --- |
| **클래스** | **MemberManage** | **회원 관리** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String column | 검색종류 |
| String search | 검색내용 |
| String sort | 정렬 |
| HashMap<String, String> map | HashMap<String, String> 객체 생성 |
| MemberDAO dao | MemberDAO 클래스 객체 생성 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 페이지 바 제작 (출력되는 페이지 변수) |
| int blockSize | 페이지 바 제작 (출력되는 페이지 변수) |
| int loop | 페이지 바 제작 (루프 변수) |
| ArrayList<MemberDTO> list | ArrayList<MemberDTO> 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.admin** | **관리자** |
| --- | --- | --- |
| **클래스** | **MemberStopN** | **정지 상태인 회원 조작하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String id | 아이디 |
| MemberDAO dao | MemberDAO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.admin** | **관리자** |
| --- | --- | --- |
| **클래스** | **MemberStopY** | **정상 상태인 회원 조작하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String id | 아이디 |
| MemberDAO dao | MemberDAO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

* **package com.project.rural.community**

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **Add** | **커뮤니티 - 글 작성하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **AddOk** | **커뮤니티 - 작성된 글을 DB작업하는 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String realFolder | 실제 저장 경로 |
| String saveFolder | 파일 저장 경로 |
| String encType | encoding 타입 |
| int maxSize | 파일 최대 크기 |
| CommunityDAO dao | CommunityDAO 클래스 객체 생성 |
| CommunityDTO dto | CommunityDTO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.community** | **회원, 비회원** |
| --- | --- | --- |
| **클래스** | **List** | **커뮤니티 - 메인 화면** |
| **변수** | String column | 검색 종류 |
| String search | 검색 내용 |
| String category | 카테고리(일손돕기, 농촌체험, 주말농장, 농작물 직거래) |
| String sort | 정렬(최신, 댓글, 조회순) |
| String id | 아이디 |
| String isSearch | 검색 유무 |
| HashMap<String, String> map | HashMap<String, String> 객체 생성 |
| CommunityDAO dao | CommunityDAO 클래스 객체 생성 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 페이지 바 제작 (출력되는 페이지 변수) |
| int blockSize | 페이지 바 제작 (출력되는 페이지 변수) |
| int loop | 페이지 바 제작 (루프 변수) |
| ArrayList<CommunityDTO> list | ArrayList<CommunityDTO> 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.community** | **회원, 비회원** |
| --- | --- | --- |
| **클래스** | **View** | **커뮤니티 - 글 상세 내용** |
| **변수** | String column | 검색 종류 |
| String search | 검색 내용 |
| String seq | 해당 글의 번호 |
| CommunityDAO dao | CommunityDAO 클래스 객체 생성 |
| CommunityDTO dto | CommunityDTO 클래스 객체 생성 |
| CommentDAO cdao | CommentDAO 클래스 객체 생성 |
| String title | 글 제목 |
| String detail | 글 내용 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 페이지 바 제작 (출력되는 페이지 변수) |
| int blockSize | 페이지 바 제작 (출력되는 페이지 변수) |
| int loop | 페이지 바 제작 (루프 변수) |
| ArrayList<String> ilist | ArrayList<String> 객체 생성 |
| ArrayList<CommentDTO> clist | ArrayList<CommentDTO>객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **Edit** | **커뮤니티 - 글 수정하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String seq | 해당 글의 번호 |
| CommunityDAO dao | CommunityDAO 클래스 객체 생성 |
| CommunityDTO dto | CommunityDTO 클래스 객체 생성 |
| ArrayList<String> ilist | ArrayList<String> 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **EditOk** | **커뮤니티 - 수정된 글을 DB작업하는 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String realFolder | 실제 저장 경로 |
| String saveFolder | 파일 저장 경로 |
| String encType | encoding 타입 |
| int maxSize | 파일 최대 크기 |
| CommunityDAO dao | CommunityDAO 클래스 객체 생성 |
| CommunityDTO dto | CommunityDTO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **DelOk** | **커뮤니티 - 자신이 올린 글 삭제하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String seq | 해당 글 번호 |
| CommunityDAO dao | CommunityDAO 클래스 객체 생성 |
| CommentDAO cdao | CommentDAO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **AddComment** | **커뮤니티 - 해당 글에 댓글 작성하기** |
| **변수** | String postSeq | 해당 글 번호 |
| String detail | 댓글 내용 |
| CommentDAO dao | CommentDAO 클래스 객체 생성 |
| CommentDTO dto | CommentDTO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **DelComment** | **커뮤니티 - 해당 글에 작성한 댓글 삭제하기** |
| **변수** | String seq | 해당 댓글 번호 |
| String postSeq | 해당 글 번호 |
| CommentDAO dao | CommentDAO 클래스 객체 생성 |
| CommentDTO dto | CommentDTO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **AddReply** | **커뮤니티 - 해당 글에 작성된 댓글 아래 답글 작성하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String pseq | 해당 댓글 번호 |
| String postSeq | 해당 글 번호 |
| String detail | 해당 답글 내용 |
| CommentDAO dao | CommentDAO 클래스 객체 생성 |
| CommentDTO dto | CommentDTO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **CommunityDTO** | **커뮤니티 - 게시글에 필요한 getter & setter** |
| **변수** | String seq | 번호 |
| String id | 아이디 |
| String category | 카테고리 |
| String title | 제목 |
| String detail | 내용 |
| String regdate | 작성일자 |
| String cnt | 조회수 |
| String name | 이름 |
| String isnew | 최신글 |
| ArrayList<String> images | 이미지 |
| String pseq | 댓글 번호 |
| String ccnt | 댓글 수 |
| String imgcnt | 이미지 수 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **CommentDTO** | **커뮤니티 - 게시글 댓글에 필요한 getter & setter** |
| **변수** | String seq | 번호 |
| String pseq | 댓글번호 |
| String postSeq | 게시글번호 |
| String id | 아이디 |
| String detail | 내용 |
| String regdate | 작성일자 |
| String name | 이름 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **CommunityDAO** | **커뮤니티 - 게시글에 필요한 데이터를 DB에 접속 후 제공하는 클래스** |
| **변수** | Connection conn | DB연결 객체 |
| PreparedStatement pstat | SQL InJection |
| ResultSet rs | 데이터 결과값 |
| **메소드** | CommunityDAO() | DB접속 |
| int add(CommunityDTO dto) | 게시글 작성 시 Insert |
| void addImg(ArrayList<String> images) | 게시글 작성 시 이미지가 있다면 추가로 Insert |
| ArrayList<CommunityDTO> list(HashMap<String, String> map) | 리스트에 출력되는 조건 별로 Select |
| int getTotalCount(HashMap<String, String> map) | 페이지 바 구현을 위한 조건 별 총 페이지 수 Select |
| void updateReadCount(String seq) | 게시글 조회수를 위한 Update |
| CommunityDTO get(String seq) | 게시글의 모든 정보를 불러오는 Select |
| ArrayList<String> listImg(String seq) | 게시글의 모든 이미지를 불러오는 Select |
| int edit(CommunityDTO dto) | 게시글의 수정을 위한 Update |
| int editImg(ArrayList<String> images, String seq) | 게시글의 이미지 수정을 위한 새로운 이미지 Insert |
| int del(String seq) | 게시글 삭제를 위한 Delete |
| int delAllImg(String seq) | 게시글 이미지 삭제를 위한 Delete |
| int getViewTotalCount(HashMap<String, String> map) | 게시글 댓글 수를 위한 Select |

| **패키지** | **package com.project.rural.community** | **회원** |
| --- | --- | --- |
| **클래스** | **CommentDAO** | **커뮤니티 - 게시글 댓글에 필요한 데이터를 DB에 접속 후 제공하는 클래스** |
| **변수** | Connection conn | DB연결 객체 |
| PreparedStatement pstat | SQL InJection |
| ResultSet rs | 데이터 결과값 |
| **메소드** | CommentDAO() | DB접속 |
| int addComment(CommentDTO dto) | 게시글에 댓글 작성을 위한 Insert |
| ArrayList<CommentDTO> listComment(HashMap<String, String> map) | 게시글에 작성된 댓글 목록을 보기 위한 Select |
| int delComment(String seq) | 게시글에 작성된 댓글을 삭제하기 위한 Delete |
| void delAllComment(String seq) | 게시글 삭제를 위한 갖고있는 댓글 전부 삭제를 위한 Delete |
| int addReply(CommentDTO dto) | 게시글에 작성된 댓글 아래 답글 작성을 위한 Insert |
| ArrayList<CommentDTO> listReply(String seq) | 게시글에 작성된 댓글 아래 답글 목록을 보기 위한 Select |

* **package com.project.rural.exp**

| **패키지** | **package com.project.rural.exp** | **관리자** |
| --- | --- | --- |
| **클래스** | **ExpDAO()** | **농촌체험관련 데이터를 관리하기 위해 DB에 접속하는 클래스** |
| **메소드** | ExpDAO() | DBUtil 클래스를 호출하여 DB 접속 |
| int add(ExpDTO dto) | (농부)사용자가 농촌체험 공고를 입력하여 데이터에 해당 정보를 추가하는 메소드 |
| ArrayList<ExpDTO> list(HashMap<String, String> map) | 농촌체험 전체 정보, 검색별 정보, 신청가능한 정보등을 불러오는 메소드 |
| ExpDTO get(String seq, String id) | (상세보기페이지) 글번호를 받고 글번호에 해당되는 레코드 내용을 전부 DTO에 담아준다. |
| int edit(ExpDTO dto | (수정하기페이지)에서 수정한 내용으로 데이터를 update 시키기 위한 메소드 |
| ArrayList<ExpDTO> addList(HashMap<String, String> map) | (농부)사용자가 등록한 공고 내역들을 불러오는 메소드 |
| int del(String seq) | (농부)사용자가 등록한 모집공고를 삭제하여 모집공고 데이터에서 데이터를 삭제하기 위한 메소드 |
| int apply(ExpApplyDTO dto) | 일반회원이 농부가 올린 모집공고에 지원하여 지원내역 테이블에 데이터를 추가하는 메소드 |
| ArrayList<ExpApplyDTO> applyList(HashMap<String, String> map) | 일반 사용자가 자신이 신청한 지원내역을 불러오는 메소드 |
| ArrayList<ExpApplyDTO> appliCantList(HashMap<String, String> map) | (농부)사용자가 자신이 올린 공고에 신청한 지원자들의 내역을 불러오는 메소드 |
| int addReview(ExpReviewDTO dto) | 일반사용자가 상세보기 페이지에서 리뷰를 추가할수 있는 메소드 |
| ArrayList<ExpReviewDTO> reviewList(String pseq) | 자세히보기 페이지에서 글번호 받아 해당글에 속한 리뷰 목록들을 불러오는 메소드 |
| int expApplyDel(String seq) | 일반사용자가 지원신청한내용을 취소(삭제)하는 메소드 |
| int expReviewDel(String seq) | 일반사용자가 자신이 남긴 리뷰를 삭제하는 메소드 |
| void delAllReview(String seq) | (농부)사용자가 모집공고글을 삭제하기전에 해당 글에 달린 리뷰들을 모두 삭제하는 메소드 |
| void delAllApply(String seq) | (농부)사용자가 모집공고글을 삭제하기전에 해당 글에 신청된 신청내역을 모두 삭제하는 메소드 |
| int passApply(String[] seqs) | 신청내역을 일괄적으로 승인기위한 메소드 (미승인/승인 내역중에 미승인 내역을 한꺼번에) |
| int getTotalCount(HashMap<String, String> map) | 농촌체험 메인페이지에서 페이징을 위헤 총 게시글(모집공고)의 숫자를 알아내기 위한 메소드 |
| int totalAddCount(HashMap<String, String> map) | (농촌)사용자의 등록내역페이지에서 페이징을 위헤 총 게시글(모집공고)의 숫자를 알아내기 위한 메소드 |
| String starAvg(String seq) | 해당 게시글에 달린 리뷰들의 개수와 별(1~5)점의 평균을 불러오기위한 메소드 |
| int totalApplyCount(HashMap<String, String> map) | 일반사용자의 신청내역페이지에서 페이징을 위해 신청내역 갯수를 알아내기 위한 메소드 |

| **패키지** | **package com.project.rural.exp** | **관리자** |
| --- | --- | --- |
| **클래스** | **ExpDTO()** | **농촌체험 카테고리의 모집공고정보를 위한 getter & setter 클래스** |
| **변수** | String seq | 글번호 |
| String address | 주소 |
| String town | 마을명 |
| String expInfo | 체험 대분류 |
| String name | 체험명 |
| String startDate | 체험 시작날짜 |
| String endDate | 체험 종료날짜 |
| String tel | 전화번호 |
| String site | 사이트 |
| String townDetail | 마을 설명 |
| String detail | 체험 설명 |
| String totalPerson | 모집인원 |
| String image | 첨부 이미지 파일명 |
| String regDate | 등록날짜 |
| String isnew | 새로운 게시글 |
| String userName | 사용자 이름-(등록한 농부이름) |
| String map | 지도를 띄우기위해 가공된 주소 |
| String count | 신청자의 명수 |
| String starRs | 별점 평균 |
| int applyCnt | 지원한 사람의 명수 |
| int applyperson | 신청 완료한 사용자 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.exp** | **관리자** |
| --- | --- | --- |
| **클래스** | **ExpApplyDTO()** | **농촌체험 카테고리의 신청내역을 위한 getter & setter 클래스** |
| **변수** | String seq | 신청글번호 |
| String pseq | 신청한 모집공고의 글번호 |
| String id | 아이디 |
| String startDate | 체험시작날짜 |
| String endDate | 체험종료날짜 |
| String regDate | 체험 신청날짜 |
| String totalPerson | 신청인원 |
| String detail | 신청내용 |
| String isPass | 승인/미승인 여부 |
| String expName | 체험명 |
| String expInfo | 체험대분류 |
| Strgin name | 신청자 이름 |
| String gender | 신청자 성별 |
| String age | 신청자 나이 |
| String tel | 신청자 전화번호 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.exp** | **관리자** |
| --- | --- | --- |
| **클래스** | **ExpReviewDTO()** | **농촌체험 카테고리의 모집공고에 달린 리뷰를 위한 getter & setter 클래스** |
| **변수** | String seq | 리뷰글번호 |
| String pseq | 리뷰를 작성한 모집공고의 글번호 |
| String id | 아이디 |
| String regDate | 리뷰 작성 날짜 |
| String star | 별점(1~5) |
| String detail | 리뷰 내용 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.exp** | **농부회원** |
| --- | --- | --- |
| **클래스** | **Add** | **농촌체험(농부)에서 글을 작성하기 위한 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| **메소드** | protected void doGet() | 해당 페이지가 Get방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **농부회원** |
| --- | --- | --- |
| **클래스** | **AddOk** | **농촌체험(농부)에서 작성된 글을 DB작업하기 위한 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String realPath | 파일이 저장되는 실제 경로 |
| File dir | 파일 저장 경로 |
| int sizeLimit | 파일 크기 제한 설정 |
| MultipartRequest multpartRequest | MultipartRequest 객체를 생성 (파일업로드수행위해서) |
| String image | 등록한 파일명 |
| String address | 등록한 주소 |
| String town | 등록한 마을명 |
| String expInfo | 등록한 체험대분류 |
| String name | 등록한 체험명 |
| String startDate | 등록한 체험시작날짜 |
| String endDate | 등록한 체험종료날짜 |
| String tel | 등록한 전화번호 |
| String site | 등록한 사이트 |
| String townDetail | 등록한 마을설명 |
| String detail | 등록한 체험설명 |
| String totalPerson | 등록한 모집인원 |
| HttpSession session | 로그인한 상태의 id를 받아오기 위한 변수 |
| expDAO dao | ExpDAO 클래스 객체 생성 |
| expDTO dto | ExpDTO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doPost() | 해당 페이지가 Post방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **공통회원** |
| --- | --- | --- |
| **클래스** | **List** | **농촌체험(공통)카테고리 메인페이지** |
| **변수** | String paramSido | 시도 검색 |
| String paramGugun | 구군검색 |
| String paramExpInf | 체험 대분류(농사체험/생태체험..등) |
| String paramRegion | 지역검색 |
| String paramOnlyApply | 신청가능한내역 |
| String fSearch | 검색 유무 |
| HashMap<String, String> map | HashMap<String, String> 객체 생성 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| String page | 매개변수로 받아오 페이지번호 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 페이지 바 제작 (출력되는 페이지 변수) |
| int blockSize | 페이지 바 제작 (출력되는 페이지 변수) |
| int loop | 페이지 바 제작 (루프 변수) |
| ArrayList<ExpDTO> list | ArrayList<ExpDTO> 객체 생성 |
| String pagebar | 페이지바를 만드는 html 코드 |
| **메소드** | protected void doGet() | 해당 페이지가 Get방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **공통회원** |
| --- | --- | --- |
| **클래스** | **View** | **농촌체험(공통) 카테고리 상세보기페이지** |
| **변수** | String seq | 상세히볼 모집공고의 글번호 |
| String startDate | 체험 시작 날짜 |
| String endDate | 체험 종료 날짜 |
| HttpSession session | 세션 클래스 객체 생성 |
| String id | 로그인한 사람의 아이디 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| ExpDTO dto | ExpDTO 클래스 객체 생성 |
| ArrayList<ExpReviewDTO> rlist | ArrayList<ExpReviewDTO> 객체 생성 |
| String starAvg | 리뷰 평점 |
| **메소드** | protected void doGet() | 해당 페이지가 Get방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **농부회원** |
| --- | --- | --- |
| **클래스** | **Edit** | **농촌체험 카테고리 모집공고 수정하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String seq | 해당 글의 번호 |
| HttpSession session | 세션 클래스 객체 생성 |
| String id | 로그인한 사람의 아이디 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| ExpDTO dto | ExpDTO dto 클래스 객체 생성 |
| String startDate | 체험 시작 날짜 |
| String endDate | 체험 종료 날짜 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.exp** | **농부회원** |
| --- | --- | --- |
| **클래스** | **EditOk** | **농촌체험 카테고리 수정된 모집공고를 DB작업하는 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String realPath | 파일이 저장되는 실제 경로 |
| File dir | 파일 저장 경로 |
| int sizeLimit | 파일 크기 제한 설정 |
| MultipartRequest multpartRequest | MultipartRequest 객체를 생성 (파일업로드수행위해서) |
| String seq | 수정된 글번호 |
| String image | 수정된 파일명 |
| String address | 수정된 주소 |
| String town | 수정된 마을명 |
| String expInfo | 수정된 체험대분류 |
| String name | 수정된 체험명 |
| String startDate | 수정된 체험시작날짜 |
| String endDate | 수정된 체험종료날짜 |
| String tel | 수정된 전화번호 |
| String site | 수정된 사이트 |
| String townDetail | 등록한 마을설명 |
| String detail | 등록한 체험설명 |
| String totalPerson | 등록한 모집인원 |
| HttpSession session | 로그인한 상태의 id를 받아오기 위한 변수 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| ExpDTO dto | ExpDTO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doPost() | 해당 페이지가 Post방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **농부회원** |
| --- | --- | --- |
| **클래스** | **DelOk** | **농촌체험 카테고리 모집공고 삭제하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String seq | 삭제될 글번호 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| HttpSession session | 세션 클래스 객체 생성 |
| String lv | 로그인한 사람의 회원레벨 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doGet() | 해당 페이지가 Get방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **일반회원** |
| --- | --- | --- |
| **클래스** | **Apply** | **농촌체험(일반)에서 원하는 모집공고에 신청하기 위한 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
|  | String seq | 신청할 모집공고의 글번호 |
|  | String startDate | 신청할 모집공고의 체험시작날짜 |
|  | String endDate | 신청할 모집공고의 체험종료날짜 |
| **메소드** | protected void doGet() | 해당 페이지가 Get방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **일반회원** |
| --- | --- | --- |
| **클래스** | **ApplyOk** | **농촌체험(일반)에서 신청된 내용을 DB작업하기 위한 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String pseq | 파일이 저장되는 실제 경로 |
| String totalPerson | 파일 저장 경로 |
| String startDate | 파일 크기 제한 설정 |
| String endDate | MultipartRequest 객체를 생성 (파일업로드수행위해서) |
| String detail | 등록한 파일명 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| ExpApplyDTO dto | ExpApplyDTO 클래스 객체 생성 |
| HttpSession session | 로그인한 상태의 id를 받아오기 위한 변수 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doPost() | 해당 페이지가 Post방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **일반회원** |
| --- | --- | --- |
| **클래스** | **ApplyList** | **농촌체험에서 사용자(일반)가 신청한 내역을 목록으로 불러오는 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| HashMap<String, String> map | 페이지시작/끝/아이디를 넣을 Hashmap 객체 생성 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| String page | 매개변수로 받아오 페이지번호 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 페이지 바 제작 (출력되는 페이지 변수) |
| int blockSize | 페이지 바 제작 (출력되는 페이지 변수) |
| int loop | 페이지 바 제작 (루프 변수) |
| ArrayList<ExpDTO> list | ArrayList<ExpDTO> 객체 생성 |
| String pagebar | 페이지바를 만드는 html 코드 |
| **메소드** | protected void doGet() | 해당 페이지가 Get방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **일반회원** |
| --- | --- | --- |
| **클래스** | **ApplyDelOk** | **농촌체험에서 사용자(일반)가 지원신청한내용을 취소(삭제)하는 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String seq | 삭제될 신청 글번호 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| ExpApplyDTO dto | ExpApplyDTO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doGet() | 해당 페이지가 Get방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **농부회원** |
| --- | --- | --- |
| **클래스** | **AddList** | **농촌체험에서 사용자(농부)가 등록한 내역을 목록으로 불러오는 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| HashMap<String, String> map | 페이지시작/끝/아이디를 넣을 Hashmap 객체 생성 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| String page | 매개변수로 받아오 페이지번호 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 페이지 바 제작 (출력되는 페이지 변수) |
| int blockSize | 페이지 바 제작 (출력되는 페이지 변수) |
| int loop | 페이지 바 제작 (루프 변수) |
| ArrayList<ExpDTO> list | ArrayList<ExpDTO> 객체 생성 |
| String pagebar | 페이지바를 만드는 html 코드 |
| **메소드** | protected void doGet() | 해당 페이지가 Get방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **농부회원** |
| --- | --- | --- |
| **클래스** | **Applicant** | **농촌체험에서 사용자(농부)가 등록한 내역의 신청한 지원자의 현황을 목록으로 불러오는 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String seq | 신청현황을 확인할 모집공고 글번호 |
| String name | 체험명 |
| String isView | 목록을 띄울떄 전체보기/승인/미승인 3개중 어떤 변수인지 |
| HashMap<String, String> map | 페이지시작/끝/아이디를 넣을 Hashmap 객체 생성 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| ArrayList<ExpDTO> list | ArrayList<ExpDTO> 객체 생성 |
| **메소드** | protected void doGet() | 해당 페이지가 Get방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **농부회원** |
| --- | --- | --- |
| **클래스** | **ApplyPassOk** | **농촌체험에서 사용자(농부)가 일반회원들의 신청내역을 승인처리하는 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String[] seqs | 승인할 신청내역들의 신청글번호 |
| String pseq | 신청한 모집공고의 글번호 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| PrintWriter writer | 바로 alert창을 띄우기위한 PrintWriter 객체 생성 |
| **메소드** | protected void doGet() | 해당 페이지가 Get방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **일반회원** |
| --- | --- | --- |
| **클래스** | **AddReviewOk** | **농촌체험에서 사용자(일반)가 리뷰를 작성하기 위한 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String pseq | 리뷰를 남기는 모집공고의 글번호 |
| String star | 사용자가 남긴 별점 |
| String detail | 사용자가 남긴 리뷰내용 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| ExpReviewDTO dto | ExpReviewDTO 클래스 객체 생성 |
| HttpSession session | 로그인한 상태의 id를 받아오기 위한 변수 |
| int result | 성공, 실패 유무 확인 |
| HttpSession session | 로그인한 상태의 id를 받아오기 위한 변수 |
| PrintWriter writer | 바로 alert창을 띄우기위한 PrintWriter 객체 생성 |
| **메소드** | protected void doPost() | 해당 페이지가 Post방식으로 호출됨 |

| **패키지** | **package com.project.rural.exp** | **일반회원** |
| --- | --- | --- |
| **클래스** | **ReviewDelOk** | **농촌체험에서 사용자(일반)가 자신이 작성한 리뷰를 삭제하기 위한 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String seq | 삭제될 리뷰번호 |
| String pseq | 해당 리뷰가 있는 모집공고의 글번호 |
| ExpDAO dao | ExpDAO 클래스 객체 생성 |
| ExpReviewDTO dto | ExpReviewDTO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doGet() | 해당 페이지가 Get방식으로 호출됨 |

* **package com.project.rural.farm**

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **Add** | **주말농장 등록(추가)** |
| **변수** | CheckMember cm | 회원인지 확인하는 객체 |
| **메소드** | doGet() | add.jsp 페이지를 GET방식으로 호출 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **AddOk** | **주말농장 등록(추가) DB 반영** |
| **변수** | String name | 농장명 |
| String area | 면적 |
| String price | 임대료 |
| String tel | 전화번호 |
| String address | 주소 |
| String detail | 상세내용 |
| String image | 프로필 이미지 파일명 |
| MultipartRequest multi | 파일 업로드를 위한 객체 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| FarmDTO dto | 주말농장 관련 데이터를 담기 위한 DTO 객체 |
| HttpSession session | 세션 정보를 담는 객체 |
| **메소드** | doPost() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **List** | **주말농장 목록 조회** |
| **변수** | String sido | 시/도 |
| String gugunl | 시/군/구 |
| String address | 주소 |
| String area | 면적 |
| String price | 임대료 |
| String column | 검색 분류 |
| String search | 검색어 |
| String isSearch | 검색 유무 |
| HashMap<String,String> map | 페이지에 데이터를 전달하기 위한 변수 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 출력되는 페이지 번호 |
| int loop | while 루프 변수 |
| int blockSize | 한번에 표시될 페이지 수 |
| String page | 선택된 페이지 번호 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| String pagebar | 페이지바 html 코드 |
| ArrayList<FarmDTO> list | 주말농장 목록을 담는 객체 |
| **메소드** | doGet() | list.jsp 페이지를 GET방식으로 호출 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **View** | **주말농장 조회** |
| **변수** | String seq | 주말농장 seq 번호 |
| String page | 선택된 페이지 번호 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 출력되는 페이지 번호 |
| int loop | while 루프 변수 |
| int blockSize | 한번에 표시될 페이지 수 |
| HashMap<String,String> map | 페이지에 데이터를 전달하기 위한 변수 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| String pagebar | 페이지바 html 코드 |
| FarmDTO dto | 주말농장 관련 데이터를 담기 위한 DTO 객체 |
| String detail | 개행문자 처리한 상세내용을 담기 위한 변수 |
| ArrayList<ReviewDTO> rlist | 주말농장 리뷰 목록을 담는 객체 |
| **메소드** | doGet() | view.jsp 페이지를 GET방식으로 호출 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **Edit** | **주말농장 수정** |
| **변수** | CheckMember cm | 회원인지 확인하는 객체 |
| String seq | 주말농장 seq 번호 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| FarmDTO dto | 주말농장 관련 데이터를 담기 위한 DTO 객체 |
| String detail | 개행문자 처리한 상세내용을 담기 위한 변수 |
| **메소드** | doGet() | edit.jsp 페이지를 GET방식으로 호출 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **EditOk** | **주말농장 수정 DB 반영** |
| **변수** | String seq | 주말농장 seq 번호 |
| String name | 농장명 |
| String area | 면적 |
| String price | 임대료 |
| String tel | 전화번호 |
| String address | 주소 |
| String detail | 상세내용 |
| String image | 프로필 이미지 파일명 |
| String isEnd | 마감 유무 |
| String orgImage | 원본 프로필 이미지 파일명 |
| MultipartRequest multi | 파일 업로드를 위한 객체 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| FarmDTO dto | 주말농장 관련 데이터를 담기 위한 DTO 객체 |
| int result | DB 처리결과를 담기 위한 변수 |
| **메소드** | doPost() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **DelOk** | **주말농장 삭제 DB 반영** |
| **변수** | CheckMember cm | 회원인지 확인하는 객체 |
| String seq | 주말농장 seq 번호 |
| String image | 프로필 이미지 파일명 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| int result | DB 처리결과를 담기 위한 변수 |
| **메소드** | doPost() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **AddReviewOk** | **리뷰 작성 DB 반영** |
| **변수** | CheckMember cm | 회원인지 확인하는 객체 |
| String pseq | 주말농장 seq 번호 |
| String detail | 리뷰 내용 |
| String star | 리뷰 별점 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| ReviewDTO dto | 주말농장 리뷰 관련 데이터를 담기 위한 DTO 객체 |
| HttpSession session | 세션 정보를 담는 객체 |
| int result | DB 처리결과를 담기 위한 변수 |
| **메소드** | doPost() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **DelReviewOk** | **리뷰 삭제 DB 반영** |
| **변수** | CheckMember cm | 회원인지 확인하는 객체 |
| String seq | 리뷰 seq 번호 |
| String pseq | 주말농장 seq 번호 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| int result | DB 처리결과를 담기 위한 변수 |
| **메소드** | doPost() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **AddList** | **주말농장 등록내역 조회** |
| **변수** | CheckMember cm | 회원인지 확인하는 객체 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| HttpSession session | 세션 정보를 담는 객체 |
| String id | 세션에 저장된 아이디를 담기 위한 변수 |
| ArrayList<FarmDTO> list | 회원의 주말농장 등록 내역을 담는 객체 |
| **메소드** | doGet() | addlist.jsp 페이지를 GET방식으로 호출 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **AddApplyOk** | **주말농장 임대 신청 DB 반영** |
| **변수** | CheckMember cm | 회원인지 확인하는 객체 |
| HttpSession session | 세션 정보를 담는 객체 |
| String id | 회원 id |
| String pseq | 주말농장 seq 번호 |
| String area | 임대 신청 면적 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| ApplyDTO dto | 주말농장 임대 신청 관련 데이터를 담기 위한 DTO 객체 |
| int result | DB 처리결과를 담기 위한 변수 |
| **메소드** | doPost() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **ApplyList** | **주말농장 임대 신청내역 조회** |
| **변수** | CheckMember cm | 회원인지 확인하는 객체 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| HttpSession session | 세션 정보를 담는 객체 |
| String id | 회원 id |
| ArrayList<ApplyDTO> list | 회원의 주말농장 임대 신청 내역을 담는 객체 |
| **메소드** | doGet() | applylist.jsp 페이지를 GET방식으로 호출 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **DelApplyOk** | **주말농장 임대신청 취소(삭제) DB 반영** |
| **변수** | CheckMember cm | 회원인지 확인하는 객체 |
| String seq | 주말농장 임대신청 seq 번호 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| int result | DB 처리결과를 담기 위한 변수 |
| **메소드** | doGet() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **ApplyPassOk** | **주말농장 임대신청 승인 DB 반영** |
| **변수** | CheckMember cm | 회원인지 확인하는 객체 |
| String[] seqs | 주말농장 임대신청 승인할 인대신청 seq 번호 목록 |
| String pseq | 주말농장 seq 번호 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| int result | DB 처리결과를 담기 위한 변수 |
| **메소드** | doPost() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **Applicant** | **주말농장 임대 신청현황 조회** |
| **변수** | CheckMember cm | 회원인지 확인하는 객체 |
| String seq | 주말농장 seq 번호 |
| String isView | 전체/승인/미승인 필터링 변수 |
| String page | 선택된 페이지 번호 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 출력되는 페이지 번호 |
| int loop | while 루프 변수 |
| int blockSize | 한번에 표시될 페이지 수 |
| HashMap<String,String> map | 페이지에 데이터를 전달하기 위한 변수 |
| FarmDAO dao | 주말농장 관련 DB 작업을 위한 DAO 객체 |
| String pagebar | 페이지바 html 코드 |
| ArrayList<ApplyDTO> list | 주말농장의 임대 신청 내역을 담는 객체 |
| **메소드** | doGet() | applylist.jsp 페이지를 GET방식으로 호출 |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **FarmDTO** | **주말농장 관련 정보를 담기 위한 클래스** |
| **변수** | String seq | 번호 |
| String id | 등록자 아이디 |
| String name | 농장 이름 |
| String tel | 전화번호 |
| String address | 주소 |
| String area | 총면적 |
| String price | 임대료 |
| String detail | 상세내용 |
| String isEnd | 마감여부 |
| String regDate | 등록날짜 |
| String image | 프로필 이미지 파일명 |
| String extraArea | 잔여면적 |
| String avgStar | 리뷰 평균 별점 |
| String reviewCount | 총 리뷰 개수 |
| String ownerName | 농장주 이름 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **ReviewDTO** | **주말농장 리뷰 관련 정보를 담기 위한 클래스** |
| **변수** | String seq | 번호 |
| String pseq | 주말농장 번호 |
| String id | 작성자 아이디 |
| String detail | 상세내용 |
| String regDate | 등록날짜 |
| String star | 별점 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **ApplyDTO** | **주말농장 임대신청 정보를 담기 위한 클래스** |
| **변수** | String seq | 번호 |
| String pseq | 주말농장 번호 |
| String id | 신청자 아이디 |
| String area | 신청 면적 |
| String regDate | 등록날짜 |
| String isPass | 승인여부 |
| String passDate | 승인날짜 |
| String name | 농장 이름 |
| String price | 임대료 |
| String address | 주말농장 주소 |
| String tel | 주말농장 전화번호 |
| String ownerId | 농장주 아이디 |
| String ownerName | 농장주 이름 |
| String userName | 신청자 이름 |
| String userTel | 신청자 전화번호 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.farm** | **주말농장** |
| --- | --- | --- |
| **클래스** | **FarmDAO** | 주말농장 관련 DB 작업을 위한 클래스 |
| **변수** | Connection conn | DB 연결 객체 |
| PreparedStatement pstat | sql 쿼리 실행을 위한 객체 |
| ResultSet rs | sql 쿼리 실행 결과를 담을 객체 |
| **생성자** | public FarmDAO() | 생성과 동시에 DB 연결 |
| **메소드** | public int add(FarmDTO dto) | 주말농장 데이터 삽입 (Insert) |
| public ArrayList<FarmDTO> list(HashMap<String, String> map) | 주말농장 데이터 목록 조회 (Select) |
| public FarmDTO get(String seq) | 주말농장 데이터 조회 (Select) |
| public int edit(FarmDTO dto) | 주말농장 데이터 수정 (Update) |
| public int del(String seq) | 주말농장 데이터 삭제 (Delete) |
| public int addReview(ReviewDTO dto) | 주말농장 리뷰 데이터 삽입 (Insert) |
| public ArrayList<ReviewDTO> listReview(HashMap<String, String> map) | 주말농장 리뷰 데이터 목록 조회 (Select) |
| public int delReview(String seq) | 주말농장 리뷰 데이터 삭제 (Delete) |
| public void delAllReview(String seq) | 특정 주말농장에 딸린 모든 리뷰 데이터 삭제 (Delete) |
| public void delAllapply(String seq) | 특정 주말농장에 딸린 모든 신청 데이터 삭제 (Delete) |
| public ArrayList<FarmDTO> addList(String id) | 특정 회원이 등록한 모든 주말농장 데이터 조회 (Select) |
| public int addApply(ApplyDTO dto) | 주말농장 신청 데이터 삽입 (Insert) |
| public ArrayList<ApplyDTO> applyList(String id) | 특정 회원의 모든 주말농장 신청 데이터 조회 (Select) |
| public int delApply(String seq) | 주말농장 신청 데이터 삭제 (Delete) |
| public ArrayList<ApplyDTO> applicant(HashMap<String, String> map) | 특정 주말농장의 모든 신청 데이터 조회 (Select) |
| public int passApply(String[] seqs) | 주말농장 신청 데이터 승인으로 수정 (Update) |
| public int getTotalCount(HashMap<String, String> map) | 검색 결과에 따른 총 주말농장 수를 반환 |
| public int getTotalRevCount(String seq) | 특정 주말농장에 딸린 총 리뷰 개수를 반환 |
| public int getTotalApplicant(HashMap<String, String> map) | 특정 주말농장의 총 신청 개수를 반환 |

* **package com.project.rural.market**

| **패키지** | **package com.project.rural.market** | **농부회원** |
| --- | --- | --- |
| **클래스** | **Add** | **농산물직거래 - 글 등록하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.market** | **농부회원** |
| --- | --- | --- |
| **클래스** | **AddList** | **농산물직거래 - 등록내역 리스트** |
| **변수** | CheckMember | 로그인 여부 확인 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 페이지 바 제작 (출력되는 페이지 변수) |
| int blockSize | 페이지 바 제작 (출력되는 페이지 변수) |
| int loop | 페이지 바 제작 (루프 변수) |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |
| **메소드** | ArrayList<NoticeDTO> addlist | ArrayList<MarketDTO> 객체 생성 |

| **패키지** | **package com.project.rural.market** | **농부회원** |
| --- | --- | --- |
| **클래스** | **AddOk** | **농산물직거래 - 글 등록하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String realFolder | 실제 저장 경로 |
| String saveFolder | 파일 저장 경로 |
| String encType | encoding 타입 |
| int maxSize | 파일 최대 크기 |
| String image | 이미지 저장 |
| String marketInfo | 카테고리 |
| String name | 제품명 |
| String brandName | 브랜드이름 |
| String tel | 전화번호 |
| String address | 주소 |
| String detail | 상세내용 |
| String site | 사이즈주소 |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.market** | **일반회원** |
| --- | --- | --- |
| **클래스** | **AddQnA** | **농산물직거래 - 문의사항 등록하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| result | 결과값을 받아오는 변수 |
| pseq | 보고 있던 글의 글 번호 |
| detailq | 문의글 상세내용 |
| title | 문의글 제목 |
| MarketDAO dao | MarketDAO 클래스 객체 생성 |
| MarketQADTO dto | MarketQADTO 클래스 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.market** | **농부회원** |
| --- | --- | --- |
| **클래스** | **AddQnA** | **농산물직거래 - 답변사항 등록하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| result | 결과값을 받아오는 변수 |
| isa | 답변이 있으면 y, 없으면 n |
| pseq | 보고 있던 글의 글 번호 |
| detaila | 답변글 상세내용 |
| MarketDAO dao | MarketDAO 클래스 객체 생성 |
| MarketQADTO dto | MarketQADTO 클래스 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.market** | **농부회원** |
| --- | --- | --- |
| **클래스** | **DelOk** | **농산물직거래 - 글 삭제하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| seq | 삭제할 글 번호 |
| MarketDAO dao | MarketDAO 클래스 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.market** | **일반회원** |
| --- | --- | --- |
| **클래스** | **DelQnA** | **농산물직거래 - 문의 삭제하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| pseq | 삭제할 글 번호 |
| seq | 삭제할 문의 글 번호 |
| MarketDAO dao | MarketDAO 클래스 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.market** | **농부회원** |
| --- | --- | --- |
| **클래스** | **DelReqna** | **농산물직거래 - 답변 삭제하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| pseq | 삭제할 글 번호 |
| seq | 삭제할 문의 글 번호 |
| detaila | 답변 글 내용 |
| isa | 답변이 있으면 y, 없으면 n |
| MarketDAO dao | MarketDAO 클래스 객체 생성 |
| MarketQADTO dto | MarketQADTO 클래스 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.market** | **농부회원** |
| --- | --- | --- |
| **클래스** | **Edit** | **농산물직거래 - 수정하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| seq | 삭제할 글 번호 |
| ArrayList<String> ilist | 이미지 리스트(상세) |
| MarketDAO dao | MarketDAO 클래스 객체 생성 |
| MarketDTO dto | MarketDTO 클래스 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.market** | **농부회원** |
| --- | --- | --- |
| **클래스** | **EditOk** | **농산물직거래 - 수정하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String realFolder | 실제 저장 경로 |
| String saveFolder | 파일 저장 경로 |
| String encType | encoding 타입 |
| int maxSize | 파일 최대 크기 |
| ArrayList<String> images | 이미지 리스트 |
| String seq | 글번호 |
| String marketInfo | 카테고리 |
| String name | 제품명 |
| String brandName | 브랜드 이름 |
| String tel | 전화번호 |
| String address | 주소 |
| String detail | 상세내용 |
| String site | 사이트 주소 |
| MarketDAO dao | MarketDAO 클래스 객체 생성 |
| MarketDTO dto | MarketDTO 클래스 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.market** | **공통** |
| --- | --- | --- |
| **클래스** | **LikeDel** | **농산물직거래 - 좋아요 삭제** |
| **변수** | CheckMember | 로그인 여부 확인 |
| seq | 글 번호 |
| MarketDAO dao | MarketDAO 객체 생성 |
| LikeDTO dto | LikeDTO 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.market** | **공통** |
| --- | --- | --- |
| **클래스** | **LikeDTO** | **농산물직거래 - 좋아요 정보에 필요한 getter & setter** |
| **변수** | String id | 좋아요 누른 아이디 |
| String seq | 좋아요 누른 글번호 |
| **메소드** | set/get | setter/getter |

| **패키지** | **package com.project.rural.market** | **공통** |
| --- | --- | --- |
| **클래스** | **LikeDTO** | **농산물직거래 - 좋아요 정보에 필요한 getter & setter** |
| **변수** | String id | 좋아요 누른 아이디 |
| String seq | 좋아요 누른 글번호 |
| **메소드** | set/get | setter/getter |

| **패키지** | **package com.project.rural.market** | **공통** |
| --- | --- | --- |
| **클래스** | **LikeOk** | **농산물직거래 - 좋아요** |
| **변수** | String seq | 좋아요 누른 글번호 |
| MarketDAO dao | MarketDAO 객체 생성 |
| LikeDTO dto | LikeDTO 객체 생성 |
| **메소드** | set/get | setter/getter |

| **패키지** | **package com.project.rural.market** | **공통** |
| --- | --- | --- |
| **클래스** | **List** | **농산물직거래 - 목록** |
| **변수** | String id | 아이디 |
| String search | 키워드 검색 |
| String isSearch | 검색어가 있으면 y, 없으면 n |
| String isLike | 좋아요체크박스를 체크하면y 아니면 n |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 페이지 바 제작 (출력되는 페이지 변수) |
| int blockSize | 페이지 바 제작 (출력되는 페이지 변수) |
| int loop | 페이지 바 제작 (루프 변수) |
| ArrayList<LikeDTO> likelist | 좋아요 하트 체크여부 확인 |
| ArrayList<MarketDTO> list | MarketDTO 반환 |
| MarketDAO dao | MarketDAO 객체 생성 |
| LikeDTO dto | LikeDTO 객체 생성 |
| **메소드** | set/get | setter/getter |

| **패키지** | **package com.project.rural.market** | **공통** |
| --- | --- | --- |
| **클래스** | **MarketDAO** | **농산물직거래와 관련된 내용을 DB접속후 제공하는 클래스** |
| **변수** | Connection conn | DB연결 객체 |
| PreparedStatement pstat | SQL InJection |
| ResultSet rs | 데이터 결과값 |
| **메소드** | public MarketDAO() | 기본생성자 |
| public int add(MarketDTO dto) | 농작물직거래에 글 등록 |
| public ArrayList<MarketDTO> list(HashMap<String, String> map) | 농작물 직거래 목록보기, 검색하기, 좋아요 여부 |
| public MarketDTO get(String seq) | 직거래 상세 내역보기 |
| public int del(String seq) | 글 삭제 |
| public int edit(MarketDTO dto) | 글 수정 |
| public ArrayList<MarketDTO> addlist(HashMap<String, String> map) | 등록내역 리스트 확인 |
| public int addQna(MarketQADTO dto) | 질문 작성 |
| public ArrayList<MarketQADTO> listQna(String seq) | qna목록보기 |
| public int addReqna(MarketQADTO dto) | 답변 추가 |
| public int delReqna(MarketQADTO dto) | 답변 삭제 |
| public int addlike(LikeDTO dto) | 좋아요 누르기 |
| ArrayList<LikeDTO> likelist(String id) | 좋아요 목록보기 |
| public void addImg(ArrayList<String> images) | 이미지 추가 |
| public ArrayList<String> listImg(String seq) | 이미지 리스트 |
| public int delAllImg(String seq) | 이미지 삭제 |
| public int editImg(ArrayList<String> images, String seq) | 이미지 수정 |
| public int delAllQna(String seq) | 글에 딸린 모든 qna삭제 |
| public int delQna(String seq) | qna삭제 |
| public int delAlllike(String seq) | 글에 딸린 모든 좋아요 삭제 |
| public int getTotalCount(HashMap<String, String> map) | 페이지구현시필요, 모든 게시물수 |
| public int dellike(LikeDTO dto) | 좋아요 취소 |
| public int getTotalCountAdd(HashMap<String, String> map) | 등록내역 리스트 |
| public int likecnt(String seq) | 모든 사용자들이 좋아요 누른 갯수 |

| **패키지** | **package com.project.rural.market** | **공통** |
| --- | --- | --- |
| **클래스** | **MarketDTO** | **tblMarket에 있는 모든 정보** |
| **변수** | String seq | 글번호 |
| String id | 글쓴이(농부) |
| String marketInfo | 카테고리 |
| String brandName | 브랜드이름 |
| String tel | 전화번호 |
| String address | 주소 |
| String site | 사이트주소 |
| String detail | 상세내용 |
| String regDate | 글쓴일자 |
| String image | 썸네일이미지 |
| String isnew | 얼마나 오래되었는지? |
| String farmername | 농부 이름 |
| **메소드** | set/get | setter/getter |

| **패키지** | **package com.project.rural.market** | **공통** |
| --- | --- | --- |
| **클래스** | **MarketQADTO** | **tblMarketQA에 있는 모든 정보** |
| **변수** | String seq | 글번호 |
| String pseq | 부모글번호 |
| String id | Qna글쓴이 |
| String title | 제목 |
| String detailq | 문의내용 |
| String detaila | 답변내용 |
| String regdatea | 문의쓴날짜 |
| String regdateq | 답변쓴날짜 |
| String isa | 답변있으면 y, 없으면 n |
| **메소드** | set/get | setter/getter |

| **패키지** | **package com.project.rural.market** | **공통** |
| --- | --- | --- |
| **클래스** | **View** | **농산물직거래 - 상세** |
| **변수** | String session | 로그인 세션 |
| String seq | 글번호 |
| MarketDAO dao | MarketDAO 객체 생성 |
| MarketDTO dto | MarketDTO 객체 생성 |
| ArrayList<MarketQADTO> qlist | 현재보고 있는 글에 달린 QnA목록 가져오기 |
| ArrayList<String> ilist | 이미지 불러오기 |
| int cnt | 좋아요 몇개인지 불러오기 |
| **메소드** | doGet() | DB 작업 결과에 따른 페이지 이동 |

* **package com.project.rural.member**

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **MemberDTO** | **사용자 - 사용자 정보에 필요한 getter & setter** |
| **변수** | String id | 아이디 |
| String pw | 패스워드 |
| String name | 이름 |
| String birth | 생년월일 |
| String gender | 성별 |
| String tel | 전화번호 |
| String email | 이메일 |
| String address | 주소 |
| String lv | 권한등급 |
| String regDate | 가입일자 |
| String isOut | 탈퇴유무 |
| String isStop | 정지유무 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **MemberDAO** | **사용자 - 사용자의 정보와 관련된 내용을 DB접속 후 제공하는 클래스** |
| **변수** | Connection conn | DB연결 객체 |
| PreparedStatement pstat | SQL InJection |
| ResultSet rs | 데이터 결과값 |
| **메소드** | MemberDAO() | DB접속 |
| MemberDTO login(MemberDTO dto) | 로그인시 아이디/패스워드가 맞는지 Select |
| int signup(MemberDTO dto) | 회원가입시 입력한 폼 내용을 Insert |
| MemberDTO idFind(MemberDTO dto) | 아이디 찾기를 위한 입력한 정보 Select |
| MemberDTO pwFind(MemberDTO dto) | 비밀번호 찾기를 위한 입력한 정보 Select |
| void pwClear(String id, int u\_pw) | 비밀번호 찾기 성공 시 기존 비밀번호 초기화 Update |
| MemberDTO get(String id) | 자신의 정보를 확인 하기 위한 Select |
| int edit(MemberDTO dto) | 자신의 정보를 수정하기 위한 Update |
| int pwCheck(String id, String nowpw) | 해당 사용자의 패스워드 인증을 위한 Select |
| void del(String id) | 회원 탈퇴를 위한 Update |
| boolean checkId(String id) | 해당 사용자의 정보를 불러오기 위한 Select |
| ArrayList<MemberDTO> memberList(HashMap<String, String> map) | 모든 사용자의 정보를 조건 별로 불러오기 위한 Select |
| int getTotalCount(HashMap<String, String> map) | 회원관리 페이지에서 조건 별 총 페이지수를 구하기 위한 Select |
| int stopY(String id) | 해당 사용자를 정상 -> 정지 변경을 위한 Update |
| int stopN(String id) | 해당 사용자를 정지 -> 정상 변경을 위한 Update |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **FindIdOk** | **아이디 찾기** |
| **변수** | String name | 이름 |
| String email | 이메일 |
| MemberDAO dao | 회원 관련 DB 작업을 위한 DAO 객체 |
| MemberDTO dto | 회원 관련 데이터를 담기 위한 DTO 객체 |
| MemberDTO result | DB 처리결과를 담기 위한 변수 |
| **메소드** | doPost() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **FindPwOk** | **비밀번호 찾기** |
| **변수** | Random rnd | 난수 생성을 위한 객체 |
| String name | 이름 |
| String id | 아이디 |
| String email | 이메일 |
| MemberDAO dao | 회원 관련 DB 작업을 위한 DAO 객체 |
| MemberDTO dto | 회원 관련 데이터를 담기 위한 DTO 객체 |
| MemberDTO result | DB 처리결과를 담기 위한 변수 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **Login** | **로그인** |
| **메소드** | doGet() | login.jsp 페이지를 GET방식으로 호출 |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **LoginFind** | **아이디/비밀번호 찾기** |
| **메소드** | doGet() | loginfind.jsp 페이지를 GET방식으로 호출 |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **LoginOk** | **로그인** |
| **변수** | String id | 아이디 |
| String pw | 패스워드 |
| MemberDTO result | DB 처리결과를 담기 위한 변수 |
| **메소드** | doPost() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **LoginOk** | **로그아웃** |
| **변수** | HttpSession session | 로그인 세션 |
| **메소드** | doGet() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **LoginOk** | **로그아웃** |
| **메소드** | doGet() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **LoginOk** | **로그인** |
| **변수** | String id | 아이디 |
| String pw | 패스워드 |
| MemberDAO dao | MemberDAO 클래스 객체 생성 |
| **메소드** | doPost() | DB 작업 결과에 따른 페이지 이동 |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **MyEdit** | **내 정보 수정하기** |
| **변수** | String id | 아이디 |
| String tel | 전화번호 앞자리 |
| String tel1 | 전화번호 중간자리 |
| String tel2 | 전화번호 끝자리 |
| String email1 | 이메일 아이디 |
| String email2 | 이메일 주소 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **MyEditOk** | **내 정보 수정하기 DB 작업** |
| **변수** | String id | 아이디 |
| String name | 이름 |
| String gender | 성별 |
| String tel | 전화번호 |
| String email | 이메일 |
| String nowpw | 현재 비밀번호 |
| String pw | 비밀번호 |
| String address | 주소 |
| int check | 입력한 현재 비밀번호 DB 작업 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **MyPage** | **마이페이지** |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **SignUp** | **회원가입** |
| **변수** | String step | 회원가입 스텝 |
| String lv | 레벨 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.member** | **회원** |
| --- | --- | --- |
| **클래스** | **SignUpOK** | **회원가입 DB작업** |
| **변수** | String id | 아이디 |
| String pw | 비밀번호 |
| String name | 이름 |
| String birth | 생년월일 |
| String gender | 성별 |
| String tel | 전화번호 |
| String email | 이메일 |
| String address | 주소 |
| String lv | 레벨 |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

* **package com.project.rural.notice**

| **패키지** | **package com.project.rural.notice** | **관리자** |
| --- | --- | --- |
| **클래스** | **Add** | **알림마당 - 글 작성하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.notice** | **관리자** |
| --- | --- | --- |
| **클래스** | **AddOk** | **알림마당 - 작성된 글을 DB작업하는 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String realFolder | 실제 저장 경로 |
| String saveFolder | 파일 저장 경로 |
| String encType | encoding 타입 |
| int maxSize | 파일 최대 크기 |
| NoticeDAO dao | NoticeDAO 클래스 객체 생성 |
| NoticeDTO dto | NoticeDTO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.notice** | **회원, 비회원** |
| --- | --- | --- |
| **클래스** | **List** | **알림마당 - 메인 화면** |
| **변수** | String column | 검색 종류 |
| String search | 검색 내용 |
| String category | 카테고리(이벤트, 지역행사, 공지사항) |
| String sort | 정렬(최신, 조회순) |
| String isSearch | 검색 유무 |
| HashMap<String, String> map | HashMap<String, String> 객체 생성 |
| NoticeDAO dao | NoticeDAO 클래스 객체 생성 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 페이지 바 제작 (출력되는 페이지 변수) |
| int blockSize | 페이지 바 제작 (출력되는 페이지 변수) |
| int loop | 페이지 바 제작 (루프 변수) |
| ArrayList<NoticeDTO> list | ArrayList<NoticeDTO> 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.notice** | **회원, 비회원** |
| --- | --- | --- |
| **클래스** | **View** | **알림마당 - 글 상세 내용** |
| **변수** | String column | 검색 종류 |
| String search | 검색 내용 |
| String seq | 해당 글의 번호 |
| NoticeDAO dao | NoticeDAO 클래스 객체 생성 |
| NoticeDTO dto | NoticeDTO 클래스 객체 생성 |
| String title | 글 제목 |
| String detail | 글 내용 |
| ArrayList<String> ilist | ArrayList<String> 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.notice** | **관리자** |
| --- | --- | --- |
| **클래스** | **Edit** | **알림마당 - 글 수정하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String seq | 해당 글의 번호 |
| NoticeDAO dao | NoticeDAO 클래스 객체 생성 |
| NoticeDTO dto | NoticeDTO 클래스 객체 생성 |
| ArrayList<String> ilist | ArrayList<String> 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.notice** | **관리자** |
| --- | --- | --- |
| **클래스** | **EditOk** | **알림마당 - 수정된 글을 DB작업하는 클래스** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String realFolder | 실제 저장 경로 |
| String saveFolder | 파일 저장 경로 |
| String encType | encoding 타입 |
| int maxSize | 파일 최대 크기 |
| NoticeDAO dao | NoticeDAO 클래스 객체 생성 |
| NoticeDTO dto | NoticeDTO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.notice** | **관리자** |
| --- | --- | --- |
| **클래스** | **DelOk** | **알림마당 - 관리자 자신이 올린 글 삭제하기** |
| **변수** | CheckMember | 로그인 여부 확인 |
| String seq | 해당 글 번호 |
| NoticeDAO dao | NoticeDAO 클래스 객체 생성 |
| NoticeDAO cdao | NoticeDAO 클래스 객체 생성 |
| int result | 성공, 실패 유무 확인 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.notice** | **관리자** |
| --- | --- | --- |
| **클래스** | **NoticeDTO** | **알림마당 - 게시글에 필요한 getter & setter** |
| **변수** | String seq | 번호 |
| String id | 아이디 |
| String category | 카테고리 |
| String title | 제목 |
| String detail | 내용 |
| String regdate | 작성일자 |
| String cnt | 조회수 |
| String name | 이름 |
| String isnew | 최신글 |
| ArrayList<String> images | 이미지 |
| String imgcnt | 이미지 수 |
| **메소드** | getter / setter | get, set |

| **패키지** | **package com.project.rural.notice** | **관리자** |
| --- | --- | --- |
| **클래스** | **NoticeDAO** | **알림마당 - 게시글에 필요한 데이터를 DB에 접속 후 제공하는 클래스** |
| **변수** | Connection conn | DB연결 객체 |
| PreparedStatement pstat | SQL InJection |
| ResultSet rs | 데이터 결과값 |
| **메소드** | NoticeDAO() | DB접속 |
| int add(NoticeDTO dto) | 게시글 작성 시 Insert |
| void addImg(ArrayList<String> images) | 게시글 작성 시 이미지가 있다면 추가로 Insert |
| ArrayList<NoticeDTO> list(HashMap<String, String> map) | 리스트에 출력되는 조건 별로 Select |
| int getTotalCount(HashMap<String, String> map) | 페이지 바 구현을 위한 조건 별 총 페이지 수 Select |
| void updateReadCount(String seq) | 게시글 조회수를 위한 Update |
| NoticeDTO get(String seq) | 게시글의 모든 정보를 불러오는 Select |
| ArrayList<String> listImg(String seq) | 게시글의 모든 이미지를 불러오는 Select |
| int edit(NoticeDTO dto) | 게시글의 수정을 위한 Update |
| int editImg(ArrayList<String> images, String seq) | 게시글의 이미지 수정을 위한 새로운 이미지 Insert |
| int del(String seq) | 게시글 삭제를 위한 Delete |
| int delAllImg(String seq) | 게시글 이미지 삭제를 위한 Delete |

* **package com.project.rural.worker**

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **Add** | **일손돕기 - 등록하기 클래스** |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **AddList** | **일손돕기 - 등록내역보기 클래스** |
| **변수** | int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 페이지바 제작 |
| int loop | 페이지바 제작(루프변수) |
| int blockSize | 페이지바 제작 |
| String page | 페이지 |
| String pagebar | 페이지바 |
| ArrayList<WorkerDTO> list | ArrayList<WorkerDTO> 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **AddOk** | **커뮤니티 - 등록DB작업하는 클래스** |
| **변수** | String title | 제목 |
| String isEnd | 현재 상태 |
| String tel | 전화번호 |
| String startDate | 시작날짜 |
| String endDate | 종료날짜 |
| String sido | 시 (지역) |
| String gugun | 군 (지역) |
| String isCareer | 경력 유무 |
| String totalPerson | 인원수 |
| String minAge | 최소나이대 |
| String maxAge | 최대나이대 |
| String cropInfo | 작물종류 |
| String pay | 급여 |
| String detail | 글 내용 |
| String address | 싱세 주소 |
| int result | 결과(성공, 실패) |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **Applicant** | **일손돕기 - 신청현황보기 클래스** |
| **변수** | String seq | 게시물 번호 |
| ArrayList<WorkApplyDTO> list | ArrayList<WorkApplyDTO> 객체 생성 |
| String regDate | 날짜 |
| String tmpAge | 나이 |
| String detail | 상세 내용 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **ApplicantOk** | **일손돕기 - 신청승인하기DB작업 클래스** |
| **변수** | String seq | 게시물 번호 |
| int result | 결과(성공, 실패) |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **Apply** | **일손돕기 - 신청하기클래스** |
| **변수** | String seq | 게시물 번호 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **ApplyDelOk** | **일손돕기 - 신청취소하기DB작업 클래스** |
| **변수** | String seq | 게시물 번호 |
| int result | 결과(성공, 실패) |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **ApplyList** | **일손돕기 - 신청내역보기 클래스** |
| **변수** | String id | 로그인한 아이디 |
| ArrayList<WorkerDTO> list | ArrayList<WorkerDTO> 객체 생성 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **ApplyOk** | **일손돕기 - 신청하기 DB작업 클래스** |
| **변수** | String pseq | 신청내역 번호 |
| String totalApply | 인원수 |
| String detail | 상세 내용 |
| String isCareer | 경력유무 |
| int result | 결과(성공, 실패) |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **DelOk** | **일손돕기 - 삭제하기 DB작업 클래스** |
| **변수** | String seq | 글번호 |
| int result | 결과(성공, 실패) |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **Edit** | **일손돕기 - 수정하기 클래스** |
| **변수** | String seq | 글번호 |
| **메소드** | protected void doGet() | 페이지를 Get방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원** |
| --- | --- | --- |
| **클래스** | **EditOK** | **일손돕기 - 수정하기 DB작업 클래스** |
| **변수** | String totalPerson | 인원수 |
| String minAge | 최소 나이대 |
| String maxAge | 최대 나이대 |
| String cropInfo | 작물종류 |
| String pay | 수당 |
| String address | 상세 주소 |
| String detail | 상세 내역 |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **회원, 비회원** |
| --- | --- | --- |
| **클래스** | **List** | **일손돕기 - 메인화면 (목록)** |
| **변수** | String cropInfo | 인원수 |
| String pay | 최소 나이대 |
| String sido | 최대 나이대 |
| String gugun | 작물종류 |
| String possible | 수당 |
| String search | 상세 주소 |
| String isSearch | 상세 내역 |
| int nowPage | 현재 페이지 번호 |
| int totalCount | 총 게시물 수 |
| int pageSize | 한 페이지당 출력할 게시물 수 |
| int totalPage | 총 페이지 수 |
| int begin | 가져올 게시물 시작 위치 |
| int end | 가져올 게시물 끝 위치 |
| int n | 페이지바 제작 |
| int loop | 페이지바 제작(루프변수) |
| int blockSize | 페이지바 제작 |
| String page | 페이지 |
| String pagebar | 페이지바 |
| String startDate | 시작날짜 |
| String endDate | 종료 날짜 |
| String title | 글제목 |
| **메소드** | protected void doPost() | 페이지를 Post방식으로 불러옴 |

| **패키지** | **package com.project.rural.worker** | **개발자** |
| --- | --- | --- |
| **클래스** | **WorkApplyDAO** | **일손돕기 - DB에 접속하는 클래스** |
| **변수** | String sql | 쿼리문 |
| Connection conn | DB 접속. 연결정보 |
| Statement stat | 모든 SQL 구문을 실행하는 역할 |
| PreparedStatement pstat | 모든 SQL 구문을 실행하는 역할 |
| ResultSet rs | 데이터 결과셋 |
| **메소드** | public int apply(WorkApplyDTO dto) | 일손 지원하기 insert문 |
| public int cancel(String seq) | 게시물 삭제하기 delete문 |
| public ArrayList<WorkApplyDTO> applycant(String seq) | 신청 내역 가져오기 select문 |
| public int applycantok(WorkApplyDTO dto) | 신청내역 승인하기 update문 |
| public WorkApplyDTO get(String seq) | 신청 내역 가져오기 select문 |
| public void allApplydel(String seq) | 신청 취소하기 delete문 |

| **패키지** | **package com.project.rural.worker** | **개발자** |
| --- | --- | --- |
| **클래스** | **WorkApplyDTO** | **일손돕기 - 신청내역 getter & setter** |
| **변수** | String id | 아이디 |
| String seq | 신청내역번호 |
| String pseq | 글번호 |
| String totalApply | 인원수 |
| String detail | 글내용 |
| String isPass | 승인여부 |
| String regDate | 날짜 |
| String isCareer | 경력여부 |
| String name | 이름 |
| String age | 나이 |
| String gender | 성별 |
| String tel | 전화번호 |
| String birth | 생년월일 |
| **메소드** | getter & setter | getter, setter |

| **패키지** | **package com.project.rural.worker** | **개발자** |
| --- | --- | --- |
| **클래스** | **WorkerDAO** | **일손돕기 - DB에 접속하는 클래스** |
| **변수** | String sql | 쿼리문 |
| Connection conn | DB 접속. 연결정보 |
| Statement stat | 모든 SQL 구문을 실행하는 역할 |
| PreparedStatement pstat | 모든 SQL 구문을 실행하는 역할 |
| ResultSet rs | 데이터 결과셋 |
| **메소드** | public ArrayList<WorkerDTO> list(HashMap<String, String> map) | 일손돕기 공고 list select문 |
| public WorkerDTO get(String seq) | 일손돕기 공고 list select문 |
| public ArrayList<WorkerDTO> addList(HashMap<String, String> map) | 등록내역 list select문 |
| public int add(WorkerDTO dto) | 등록하기 insert문 |
| public ArrayList<WorkerDTO> applyList(String id) | 신청 내역 가져오기 select문 |
| public int del(String seq) | 등록내역 삭제하기 delete문 |
| public int editok(WorkerDTO dto) | 등록내역 수정하기 select문 |
| public int getTotalCount(HashMap<String, String> map) | 총 페이지 개수 |
| public int addListTotalCount(HashMap<String, String> map) | 총 페이지 개수 |

| **패키지** | **package com.project.rural.worker** | **개발자** |
| --- | --- | --- |
| **클래스** | **WorkApplyDTO** | **일손돕기 - 등록내역 getter & setter** |
| **변수** | String seq | 글번호 |
| String id | 아이디 |
| String city | 시(지역) |
| String gu | 군(지역) |
| String address | 주소 |
| String cropInfo | 작물종류 |
| String totalPerson | 인원수 |
| String pay | 수당 |
| String minAge | 최소 나이대 |
| String maxAge | 최대 나이대 |
| String gender | 성별 |
| String tel | 전화번호 |
| String startDate | 시작일 |
|  | String endDate | 종료일 |
|  | String regDate | 날짜 |
|  | String detail | 상세내역 |
|  | String isCareer | 경력 유무 |
|  | String isEnd | 마감 유무 |
|  | String title | 제목 |
|  | String isPass | 승인여부 |
|  | String pseq | 신청번호 |
| **메소드** | getter & setter | getter, setter |