



2019-10-24

팀 명: **Squirrel**

김희상 지한늬 송창훈 조선웅

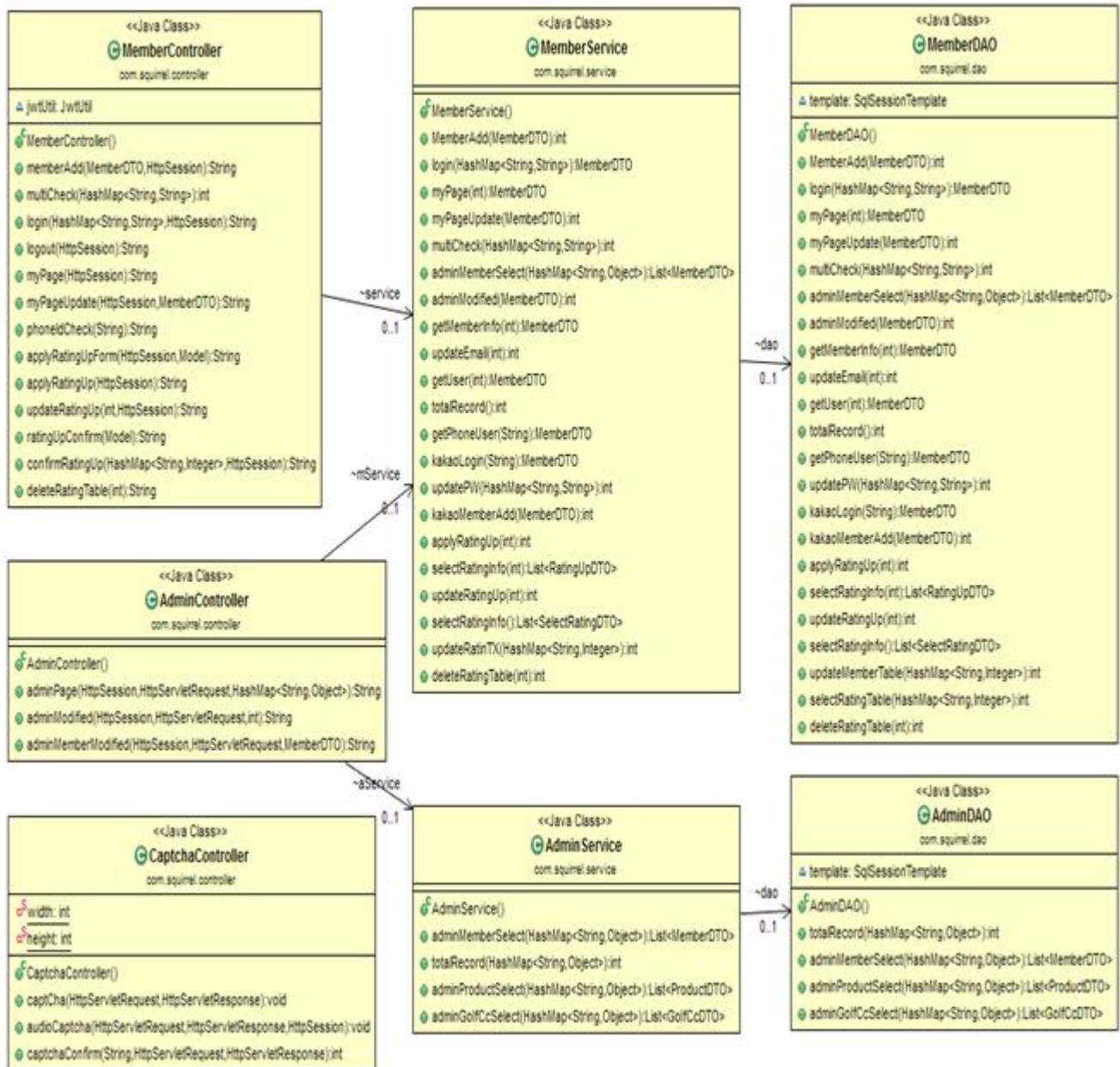
목차

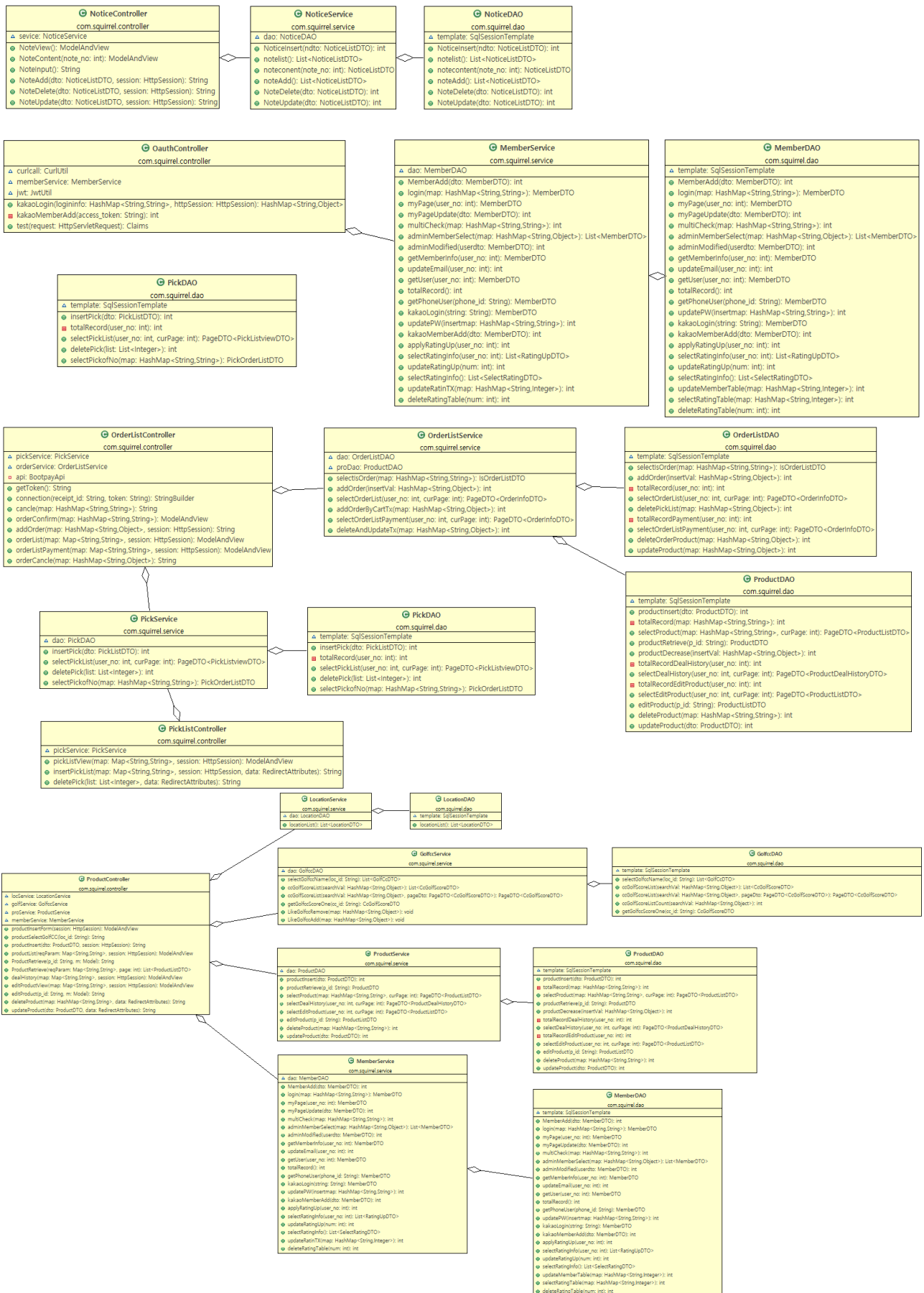
A. Controller/Service/DAO	4
A. Controller.....	7
i. MemberController.....	7
ii. AdminController.....	9
iii. CaptchaController	9
iv. OrderListController.....	10
v. PickListController.....	11
vi. ProductController.....	12
vii. ReviewController	13
viii. Re_CommentController.....	14
B. Service.....	15
i. MemberService.....	15
ii. AdminService.....	17
iii. OrderListService.....	17
iv. PickService	18
v. ProductService.....	19
vi. ReviewListService.....	20
vii. ReCommentService	21
viii. GolfccService.....	22
ix. LocationService.....	22
C. DAO	23
i. MemberDAO	23
ii. AdminDAO.....	25
iii. OrderListDAO	26
iv. PickDAO	27
v. ProductDAO.....	28
vi. ReviewListDAO	29
vii. ReCommentDAO	30
viii. GolfccDAO.....	30
ix. LocationDAO.....	31
.....	31
D. Other Class	32

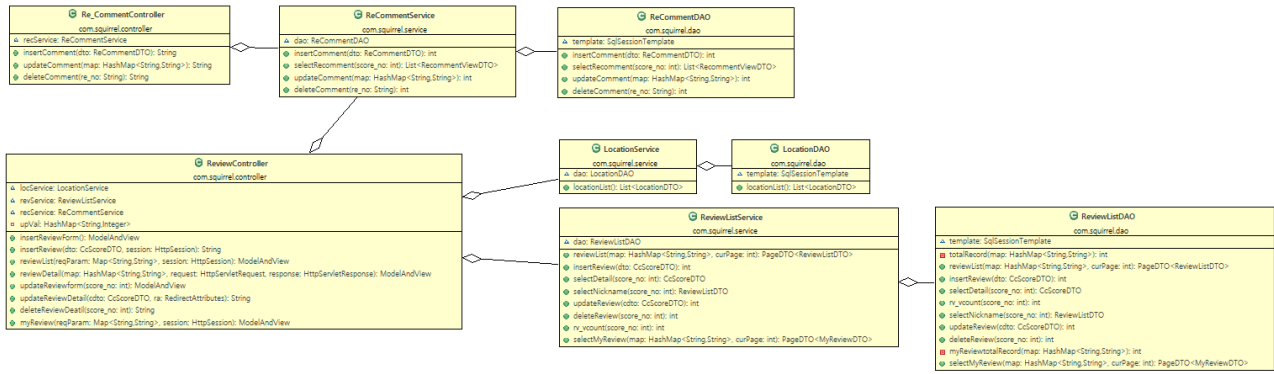
i. LoginCheckInterCeptor	32
ii. MailForUtil	32

클래스 다이어그램 Overview

A. Controller/Service/DAO







2. 클래스 다이어그램 Detail

A. Controller

i. MemberController

MemberController com.squirrel.controller
△ service: MemberService △ jwtUtil: JwtUtil
● memberAdd(mDTO: MemberDTO, session: HttpSession): String ● multiCheck(map: HashMap<String,String>): int ● login(map: HashMap<String,String>, session: HttpSession): String ● logout(session: HttpSession): String ● myPage(session: HttpSession): String ● myPageUpdate(session: HttpSession, mDTO: MemberDTO): String ● phoneIdCheck(phone_id: String): String ● applyRatingUpForm(session: HttpSession, data: Model): String ● applyRatingUp(session: HttpSession): String ● updateRatingUp(num: int, session: HttpSession): String ● ratingUpConfirm(data: Model): String ● confirmRatingUp(map: HashMap<String,Integer>, session: HttpSession): String ● deleteRatingTable(num: int): String

Method Summary	
public String	memberAdd (MemberDTO mDTO, HttpSession session) // 회원추가 method
public int	multiCheck (HashMap<String, String> map) // ajax를 이용한phone_id, password and email 확인 method
public String	login (HashMap<String, String> map, HttpSession session) // 로그인 method
Public String	logout (HttpSession session) // 로그아웃 method
Public String	myPage (HttpSession session) // 회원정보관리 method
public String	myPageUpdate (HttpSession session, MemberDTO mDTO) // 회원정보 수정 method
public String	phoneidCheck (String phone_id) // 회원 비밀번호 찾기 시 회원 id와 이메일 확인 메서드
public String	applyRatingUpForm (HttpSession session, Model model) // 일반회원 등업페이지
public String	applyRatingUp (HttpSession session) // 일반회원 등업 요청
public String	updateRatingUp (int num, HttpSession session) // 일반회원 등업 재요청
public String	ratingUpConfirm (Model data) // 관리자가 일반회원들이 요청한 등업 리스트 조회
public String	confirmRatingUp (HashMap<String, String> map, HttpSession session) // 관리자가 회원 등업 승인
public String	deleteRatingTable (int num) // 관리자가 회원 등업 거절

ii. AdminController

AdminController com.squirrel.controller	
▲ aService: AdminService	
▲ mService: MemberService	
● adminPage(session: HttpSession, request: HttpServletRequest, map: HashMap<String, Object>): String	
● adminModified(session: HttpSession, request: HttpServletRequest, user_no: int): String	
● adminMemberModified(session: HttpSession, request: HttpServletRequest, mDTO: MemberDTO): String	

Method Summary

public String	adminPage (HttpSession session, HttpServletRequest request, HashMap<String, Object> map) // 관리자가 회원, 상품, 골프장 정보를 검색하는 method
public String	adminmodified (HttpSession session, HttpServletRequest request) // 검색된 정보를 클릭하였을 시 정보를 수정할 수 있는 method
public String	adminMembermodified (HttpSession session, HttpServletRequest request, MemberDTO mDTO) // 관리자가 회원정보 수정 method
public String	adminProductmodified (HttpSession session, HttpServletRequest request, ProductDTO pDTO) // 관리자가 상품정보 수정 method - 미구현
public String	adminGolfCCmodified (HttpSession session, HttpServletRequest request, GolfCCDTO gDTO) // 관리자가 골프장정보 수정 method - 미구현

iii. CaptchaController

CaptchaController com.squirrel.controller	
width: int	
height: int	
● captcha(request: HttpServletRequest, response: HttpServletResponse): void	
● audioCaptcha(request: HttpServletRequest, response: HttpServletResponse, session: HttpSession): void	
● captchaConfirm(answer: String, request: HttpServletRequest, response: HttpServletResponse): int	

Method Summary






public void	captcha (HttpSession session, HttpServletRequest request) // captcha 실행 method
public void	audioCaptcha (HttpSession session, HttpServletRequest request) // audio를 생성해주는 method
public String	captchaConfirm (HttpSession session, HttpServletRequest request) // 입력한 값을 확인하는 method

iv. OrderListController

OrderListController com.squirrel.controller
<ul style="list-style-type: none"> pickService: PickService orderService: OrderListService api: BootpayApi
<ul style="list-style-type: none"> getToken(): String connection(receipt_id: String, token: String): StringBuilder candle(map: HashMap<String,String>): String orderConfirm(map: HashMap<String,String>): ModelAndView addOrder(map: HashMap<String,Object>, session: HttpSession): String orderList(map: Map<String,String>, session: HttpSession): ModelAndView orderListPayment(map: Map<String,String>, session: HttpSession): ModelAndView orderCandle(map: HashMap<String,Object>): String


Method Summary	
public String	getToken () // 결제 검증을 위한 인증토큰 얻기
public StringBuilder	connection (String receipt_id, String token,) // bootpay 서버로 결제검증 요청
public String	candle (HashMap<String, String> map) // bootpay 서버에 결제 취소 요청
public ModelAndView	orderConfirm (HashMap<String, String> map) // 결제 상품 확인(장바구니에서 구매, 바로구매)
public String	addOrder (HashMap<String, Object> map, HttpSession session) // 상품 결제(예약)
public ModelAndView	orderList (Map<String, String> map, HttpSession session) // 예약상품 조회
Public ModelAndView	orderListPayment (Map<String, String> map, HttpSession session) // 결제 완료된 상품 조회
private String	orderCandle (HashMap<String, Object> map) // 상품 예약 취소

v. PickListController

 PickListController com.squirrel.controller
 pickService: PickService
 pickListView(map: Map<String,String>, session: HttpSession): ModelAndView
 insertPickList(map: Map<String,String>, session: HttpSession, data: RedirectAttributes): String
 deletePick(list: List<Integer>, data: RedirectAttributes): String

Method Summary	
public ModelAndView	pickListView (Map<String, String> map, HttpSession session) // 장바구니 페이지로 이동
public String	insertPickList (Map<String, String> map, HttpSession session, RedirectAttribute flash) // 상품 장바구니에 담기
public String	deletePick (List<integer> list, RedirectAttribute data) // 회원정보 수정 메소드

vi. ProductController

 ProductController com.squirrel.controller
▲ locService: LocationService ▲ golfService: GolfccService ▲ proService: ProductService ▲ memberService: MemberService
● productInsertForm(session: HttpSession): ModelAndView ● productSelectGolfCC(loc_id: String): String ● productInsert(dto: ProductDTO, session: HttpSession): String ● productList(reqParam: Map<String,String>, session: HttpSession): ModelAndView ● ProductRetrieve(p_id: String, m: Model): String ● ProductRetrieve(reqParam: Map<String,String>, page: int): List<ProductListDTO> ● dealHistory(map: Map<String,String>, session: HttpSession): ModelAndView ● editProductView(map: Map<String,String>, session: HttpSession): ModelAndView ● editProduct(p_id: String, m: Model): String ● deleteProduct(map: HashMap<String,String>, data: RedirectAttributes): String ● updateProduct(dto: ProductDTO, data: RedirectAttributes): String

Method Summary	
public ModelAndView	productInsertForm (HttpSession session) // 상품등록 페이지
public String	productSelectGolfCC (String loc_id) // 지역에 따른 골프장을 보여주는 메서드
public String	productInsert (ProductDTO dto, HttpSession session) // 상품 등록
public ModelAndView	productList (Map<String, String> reqParam, HttpSession session) // 상품 리스트, 조건검색 가능
public String	productRetrieve (String p_id, Model m) // 브라우저 접속 시 상품 자세히 보기
public List<ProductListDTO>	productRetrieve (Map<String, String> reqParam, int page) // 브라우저 접속 시 상품 자세히 보기
public ModelAndView	dealHistory (Map<String, String> map, HttpSession session) // 매니저 등급의 회원이 자신이 등록한 상품에 대한 거래내역 조회
public ModelAndView	editProductView (Map<String, String> map, HttpSession session) // 상품 관리목록 페이지
public String	editProduct (String p_id, Model m) // 상품 수정/삭제 페이지
public String	deleteProduct (HashMap<String, Object> map, RedirectAttribute data) // 상품 삭제
public String	deleteProduct (ProductDTO dto, RedirectAttribute data) // 상품 수정

vii. ReviewController

 ReviewController com.squirrel.controller
▲ locService: LocationService ▲ revService: ReviewListService ▲ recService: ReCommentService
● insertReviewForm(): ModelAndView ● insertReview(dto: CcScoreDTO, session: HttpSession): String ● reviewList(reqParam: Map<String,String>, session: HttpSession): ModelAndView ● reviewDetail(map: HashMap<String,String>, request: HttpServletRequest, response: HttpServletResponse): ModelAndView ● updateReviewform(score_no: int): ModelAndView ● updateReviewDetail(cdto: CcScoreDTO, ra: RedirectAttributes): String ● deleteReviewDeatil(score_no: int): String ● myReview(reqParam: Map<String,String>, session: HttpSession): ModelAndView

Method Summary	
public ModelAndView	insertReviewForm() // 후기 게시판 게시물 등록 페이지
public String	insertReview (CcScoreDTO dto, HttpSession session) // 게시물 등록
public ModelAndView	reviewList (Map<String, String> reqParam, HttpSession session) // 후기 게시판 리스트
public ModelAndView	reviewDetail (HashMap<String, String> map, HttpServletRequest request, HttpServletResponse response) // 게시물 자세히 보기
public ModelAndView	updateReviewform (int score_no) // 게시물 수정/삭제 페이지
public String	updateReviewDetail (CcScoreDTO cdto, RedirectAttribute ra) // 게시물 수정
public String	deleteReviewDetail (int score_no l) // 게시물 삭제
public ModelAndView	myReview (Map<String, String> reqParam, HttpSession session) // 회원 본인이 작성한 게시물 리스트 페이지



viii. Re_CommentController

 Re_CommentController com.squirrel.controller
 recService: ReCommentService
 insertComment(dto: ReCommentDTO): String
 updateComment(map: HashMap<String,String>): String
 deleteComment(re_no: String): String

Method Summary	
public String	insertComment (ReCommentDTO dto) // Ajax 비동기처리 댓글 등록
public String	updateComment (HashMap<String, String> map) // Ajax 비동기처리 댓글 수정
public String	deleteComment (List<List<String>> list) // Ajax 비동기처리 댓글 삭제

B. Service







i. MemberService

 MemberService com.squirrel.service
 dao: MemberDAO
<ul style="list-style-type: none">● MemberAdd(dto: MemberDTO): int● login(map: HashMap<String,String>): MemberDTO● myPage(user_no: int): MemberDTO● myPageUpdate(dto: MemberDTO): int● multiCheck(map: HashMap<String,String>): int● adminMemberSelect(map: HashMap<String,Object>): List<MemberDTO>● adminModified(userdto: MemberDTO): int● getMemberInfo(user_no: int): MemberDTO● updateEmail(user_no: int): int● getUser(user_no: int): MemberDTO● totalRecord(): int● getPhoneUser(phone_id: String): MemberDTO● kakaoLogin(string: String): MemberDTO● updatePW(insertmap: HashMap<String,String>): int● kakaoMemberAdd(dto: MemberDTO): int● applyRatingUp(user_no: int): int● selectRatingInfo(user_no: int): List<RatingUpDTO>● updateRatingUp(num: int): int● selectRatingInfo(): List<SelectRatingDTO>● updateRatinTX(map: HashMap<String,Integer>): int● deleteRatingTable(num: int): int

Method Summary

public int	memberAdd (MemberDTO mDTO) // 회원추가 method
public MemberDTO	login (HashMap<String, String> map) // 로그인 method
public MemberDTO	myPage (int) // 회원정보관리 method
public int	myPageUpdate (MemberDTO mDTO) // 회원정보 수정 method
public int	multiCheck (HashMap<String, String> map) // ajax를 이용한phone_id, password and email 확인 method
public List<MemberDTO>	adminMemberSelect (HashMap<String, Object> map) // 관리자가 Member카테고리로 검색했을때.
public int	adminModified (MemberDTO mDTO) // 관리자가Member의 정보를 수정할 때
public int	totalRecord () // paging 처리를 위한 전체 레코드
public MemberDTO	getAdminMemberInfo (int user_no) // 관리자가 특정 회원의 정보 조회
public int	updateEmail (String passwd) // 회원 이메일 인증 여부 확인
public MemberDTO	getUser (int user_no) // 일치하는 유저의 정보 중 일부 (전화번호,이름,등급) 조회
public MemberDTO	getPhoneUser (String phone_id) // 회원 비밀번호 찾기 시 회원 id와 이메일 확인 메서드
public MemberDTO	kakaoLogin (String string) // 카카오 로그인 api 실행 후 카카오id 값을 통해 로그인
public int	updatePW (HashMap<String, String> insertmap) // 임시비밀번호 이메일로 전송될 때 회원 비밀번호를 임시비밀번호로 수정
public int	kakaoMemberAdd (MemberDTO dto) // 카카오톡 계정 첫 접속 시 로그인 처리
public int	applyRatingUp (int user_no) // 일반회원 등급 처리
public List<RatingUpDTO>	selectRatingInfo (int user_no) // 관리자가 등급된 매니저 회원 조회
public int	updateRatingUp (int user_no) // 일반회원 등급 재요청
public List<SelectRatingDTO>	selectRatingInfo () // 관리자가 일반회원들이 요청한 등급 리스트 조회
public int	updateRatinTX (HashMap<String, Integer> map) // 관리자가 일반회원 등급 승인 및 등급테이블 조회 트랜잭션처리
public int	deleteRatingTable (int user_no) // 관리자가 일반회원 등급 거절










ii. AdminService

 AdminService com.squirrel.service	
 dao: AdminDAO	
	adminMemberSelect(map: HashMap<String, Object>): List<MemberDTO>
	totalRecord(map: HashMap<String, Object>): int
	adminProductSelect(map: HashMap<String, Object>): List<ProductDTO>
	adminGolfCcSelect(map: HashMap<String, Object>): List<GolfCcDTO>

Method Summary

public List<MemberDTO>	adminMemberSelect (HashMap<String, Object>) // 관리자가 회원카테고리로 검색했을 시
public List<ProductDTO>	adminProductSelect (HashMap<String, Object>) // 관리자가 상품카테고리로 검색했을 시
public List<GolfCcDTO>	adminGolfCcSelect (HashMap<String, Object>) // 관리자가 골프장카테고리로 검색했을 시
public int	totalRecord () // paging 처리를 위한 전체 레코드

iii. OrderListService







 OrderListService com.squirrel.service	
 dao: OrderListDAO	
 proDao: ProductDAO	
	selectIsOrder(map: HashMap<String, String>): IsOrderListDTO
	addOrder(insertVal: HashMap<String, Object>): int
	selectOrderList(user_no: int, curPage: int): PageDTO<OrderInfoDTO>
	addOrderByCartTx(map: HashMap<String, Object>): int
	selectOrderListPayment(user_no: int, curPage: int): PageDTO<OrderInfoDTO>
	deleteAndUpdateTx(map: HashMap<String, Object>): int

Method Summary

public IsOrderListDTO	selectIsOrder (HashMap<String, String> map) // 구매 진행 확인 페이지 조회
public int	addOrder (HashMap<String, Object> insertVal) // 구매(예약) 메서드, 구매 시 상품 조인 인원 감소 트랜잭션 처리
public PageDTO<OrderInfoDTO>	selectOrderList (int user_no, int curPage) // 예약내역 조회











Method Summary	
Public int	addOrderByCartTx (HashMap<String, Object> map) // 장바구니에서 구매 시 상품 조인인원 감소, 장바구니 해당상품 삭제, 예약내역 추가 트랜잭션 처리
public PageDTO<OrderInfoDTO>	selectOrderListPayment (int user_no, int curPage) // 구매 완료 내역 조회
Public int	deleteAndUpdateTx (HashMap<String, Object> map) // 예약상품 취소, 예약내역 상품 삭제, 판매상품 수정 트랜잭션 처리

iv. PickService

 PickService com.squirrel.service	
 dao: PickDAO	
 insertPick(dto: PickListDTO): int	
 selectPickList(user_no: int, curPage: int): PageDTO<PickListviewDTO>	
 deletePick(list: List<Integer>): int	
 selectPickofNo(map: HashMap<String,String>): PickOrderListDTO	











Method Summary	
public int	insertPick (PickListDTO dto) // 상품 장바구니에 담기
public PageDTO<PickListviewDTO>	selectPickList (int user_no, int curPage) // 장바구니 리스트 보기
public int	deletePick (List<Integer> list) // 장바구니에 상품 삭제
public PickOrderListDTO	selectPickofNo (HashMap<String, String> map) // 장바구니에서 구매진행시 해당 상품 조회

v. ProductService

 ProductService com.squirrel.service
 dao: ProductDAO
 productInsert(dto: ProductDTO): int  productRetrieve(p_id: String): ProductDTO  selectProduct(map: HashMap<String,String>, curPage: int): PageDTO<ProductListDTO>  selectDealHistory(user_no: int, curPage: int): PageDTO<ProductDealHistoryDTO>  selectEditProduct(user_no: int, curPage: int): PageDTO<ProductListDTO>  editProduct(p_id: String): ProductListDTO  deleteProduct(map: HashMap<String,String>): int  updateProduct(dto: ProductDTO): int

Method Summary	
public int	productInsert (ProductDTO dto) // 상품 등록
public ProductDTO	productRetrieve (String p_id) // 상품 자세히 보기
public PageDTO<ProductListDTO>	selectProduct (HashMap<String, String> map, int curPage) // 검색 조건에 따른 상품 조회
public PageDTO<ProductDealHistoryDTO>	selectDealHistory (int user_no, int curPage) // 회원등급이 매니저인 회원이 본인이 등록한 상품에 대한 거래내역 조회
public PageDTO<ProductListDTO>	selectEditProduct (int user_no, int curPage) // 회원등급이 매니저인 회원이 본인이 등록한 상품 관리 페이지
public ProductListDTO	editProduct (String p_id) // 상품내용 수정 페이지
public int	deleteProduct (HashMap<String, String> map) // 상품 삭제
public int	updateProduct (ProductDTO dto) // 상품 수정







vi. ReviewListService

 ReviewListService com.squirrel.service	
 dao: ReviewListDAO	
	reviewList(map: HashMap<String,String>, curPage: int): PageDTO<ReviewListDTO>
	insertReview(dto: CcScoreDTO): int
	selectDetail(score_no: int): CcScoreDTO
	selectNickname(score_no: int): ReviewListDTO
	updateReview(cdto: CcScoreDTO): int
	deleteReview(score_no: int): int
	rv_vcount(score_no: int): int
	selectMyReview(map: HashMap<String,String>, curPage: int): PageDTO<MyReviewDTO>

Method Summary






public PageDTO<ReviewListDTO>	reviewList (HashMap<String, String> map, int curPage) // 후기 게시판 검색에 따른 게시글 리스트 조회
public int	insertReview (CcScoreDTO dto) // 후기 게시글 등록
public CcScoreDTO	selectDeatil (int score_no) // 후기 게시글 자세히 보기
public ReviewListDTO	selectNickname (int score_no) // 후기 게시판 작성자 조건검색
public int	updateReview (CcScoreDTO cdto) // 회원 본인의 후기 게시글 수정
public int	deleteReview (int score_no) // 회원 본인의 후기 게시글 삭제
public int	rv_vcount (int score_no) // 게시글 조회수 증가
public PageDTO<MyReviewDTO>	selectMyReview (HashMap<String, String> map, int curPage) // 회원 본인의 후기 게시글 리스트 조회

vii. ReCommentService

 ReCommentService com.squirrel.service
 dao: ReCommentDAO
 insertComment(dto: ReCommentDTO): int  selectRecomment(score_no: int): List<RecommentViewDTO>  updateComment(map: HashMap<String,String>): int  deleteComment(re_no: String): int

Method Summary	
public int	insertComment (ReCommentDTO dto) // 게시글에 대한 댓글 등록
public List<RecommentViewDTO>	selectRecomment (int score_no) // 게시글에 대한 댓글 리스트 조회
public int	updateComment (HashMap<String, String> map) // 게시글에 대한 댓글 수정
public int	deleteComment (int re_no) // 게시글에 대한 댓글 삭제

viii. GolfccService

 GolfccService com.squirrel.service	
 dao: GolfccDAO	
	selectGolfccName(loc_id: String): List<GolfCcDTO>
	ccGolfScoreList(searchVal: HashMap<String, Object>): List<CcGolfScoreDTO>
	ccGolfScoreList(searchVal: HashMap<String, Object>, pageDto: PageDTO<CcGolfScoreDTO>): PageDTO<CcGolfScoreDTO>
	getGolfccScoreOne(cc_id: String): CcGolfScoreDTO
	LikeGolfccRemove(map: HashMap<String, Object>): void
	LikeGolfccAdd(map: HashMap<String, Object>): void

Method Summary

public List<GolfCcDTO>	selectGolfccName (String loc_id) // 지역에 따른 골프장 조회
public List<CcGolfScoreDTO>	ccGolfScoreList (HashMap<String, Object> searchVal) // 골프장 평점 조회
public PageDTO<CcGolfScoreDTO>	ccGolfScoreList (HashMap<String, Object> searchVal, PageDTO<CcGolfScoreDTO> pageDto) // 골프장 평점 조회 페이지징
public CcGolfScoreDTO	getGolfccScoreOne (String cc_id) // 골프장 평점 조회 (json)
public void	LikeGolfccRemove (HashMap<String, Object> map) // 관심 골프장 삭제
public void	LikeGolfccAdd (HashMap<String, Object> map) // 관심 골프장 추가

ix. LocationService

 LocationService com.squirrel.service	
 dao: LocationDAO	
	locationList(): List<LocationDTO>

Method Summary

public List<LocationDTO>	locationList () // 지역 조회
--------------------------	------------------------------------

C. DAO

i. MemberDAO

MemberDAO com.squirrel.dao
▲ template: SqlSessionTemplate
<ul style="list-style-type: none">● MemberAdd(dto: MemberDTO): int● login(map: HashMap<String,String>): MemberDTO● myPage(user_no: int): MemberDTO● myPageUpdate(dto: MemberDTO): int● multiCheck(map: HashMap<String,String>): int● adminMemberSelect(map: HashMap<String,Object>): List<MemberDTO>● adminModified(userdto: MemberDTO): int● getMemberInfo(user_no: int): MemberDTO● updateEmail(user_no: int): int● getUser(user_no: int): MemberDTO● totalRecord(): int● getPhoneUser(phone_id: String): MemberDTO● updatePW(insertmap: HashMap<String,String>): int● kakaoLogin(string: String): MemberDTO● kakaoMemberAdd(dto: MemberDTO): int● applyRatingUp(user_no: int): int● selectRatingInfo(user_no: int): List<RatingUpDTO>● updateRatingUp(num: int): int● selectRatingInfo(): List<SelectRatingDTO>● updateMemberTable(map: HashMap<String,Integer>): int● selectRatingTable(map: HashMap<String,Integer>): int● deleteRatingTable(num: int): int

Method Summary

public int	memberAdd (MemberDTO mDTO) // 회원추가 method
public MemberDTO	login (HashMap<String, String> map) // 로그인 method
public MemberDTO	myPage (int) // 회원정보관리 method
public int	myPageUpdate (MemberDTO mDTO) // 회원정보 수정 method
public int	multiCheck (HashMap<String, String> map) // ajax를 이용한phone_id, password and email 확인 method
public List<MemberDTO>	adminMemberSelect (HashMap<String, Object> map) // 관리자가 Member카테고리로 검색했을때.
public int	adminModified (MemberDTO mDTO) // 관리자가Member의 정보를 수정할 때
public MemberDTO	getAdminMemberInfo (int user_no) // 관리자가 특정 회원의 정보 조회
public int	updateEmail (String passwd) // 회원 이메일 인증 여부 확인
public MemberDTO	getUser (int user_no) // 유저아이디를 통하여 유저정보의 일부분 (등급, 닉네임, 전화번호) 조회
public int	totalRecord () // paging 처리를 위한 전체 레코드
public MemberDTO	getPhoneUser (String phone_id) // 회원 비밀번호 찾기 시 회원 id와 이메일 확인 메서드
public int	updatePW (HashMap<String, String> insertmap) // 임시비밀번호 이메일로 전송될 때 회원 비밀번호를 임시비밀번호로 수정
public MemberDTO	kakaoLogin (String string) // 카카오id를 통하여 로그인
public int	kakaoMemberAdd (MemberDTO dto) // 카카오 로그인 첫회 실행 시 카카오 정보를 이용하여 유저 등록
public int	applyRatingUp (int user_no) // 일반회원 등급 처리
public List<RatingUpDTO>	selectRatingInfo (int user_no) // 관리자가 등급된 매니저 회원 조회
public int	updateRatingUp (int user_no) // 일반회원 등급 재요청
public List<SelectRatingDTO>	selectRatingInfo () // 관리자가 일반회원들이 요청한 등급 리스트 조회
public int	UpdateMemberTable (HashMap<String, Integer> map) // 관리자가 일반회원 등급 승인 시 해당 회원 등급 매니저로 수정
public int	selectRatingTable (int user_no) // 관리자가 등급테이블 조회
public int	deleteRatingTable (int user_no) // 관리자가 일반회원 등급 거절












ii. AdminDAO

AdminDAO com.squirrel.dao
template: SqlSessionTemplate
totalRecord(map: HashMap<String, Object>): Int
adminMemberSelect(map: HashMap<String, Object>): List<MemberDTO>
adminProductSelect(map: HashMap<String, Object>): List<ProductDTO>
adminGolfCcSelect(map: HashMap<String, Object>): List<GolfCcDTO>

Method Summary








public int	totalRecord (HashMap<String, Object>) // paging 처리를 위한 전체 레코드
public List<MemberDTO>	adminMemberSelect (HashMap<String, Object>) // 관리자가 회원카테고리로 검색했을 시
public List<ProductDTO>	adminProductSelect (HashMap<String, Object>) // 관리자가 상품카테고리로 검색했을 시
public List<GolfCcDTO>	adminGolfCcSelect (HashMap<String, Object>) // 관리자가 골프장카테고리로 검색했을 시

iii. OrderListDAO

 OrderListDAO com.squirrel.dao
 template: SqlSessionTemplate
 selectIsOrder(map: HashMap<String,String>): IsOrderListDTO  addOrder(insertVal: HashMap<String,Object>): int  totalRecord(user_no: int): int  selectOrderList(user_no: int, curPage: int): PageDTO<OrderInfoDTO>  deletePickList(map: HashMap<String,Object>): int  totalRecordPayment(user_no: int): int  selectOrderListPayment(user_no: int, curPage: int): PageDTO<OrderInfoDTO>  deleteOrderProduct(map: HashMap<String,Object>): int  updateProduct(map: HashMap<String,Object>): int

Method Summary	
public IsOrderListDTO	selectIsOrder (HashMap<String, String> map) // 구매 진행 확인 페이지 조회
public int	addOrder (HashMap<String, Object> insertVal) // 구매(예약) 메서드, 구매 시 상품 조인 인원 감소 트랜잭션 처리
public int	totalRecord (int user_no) // 예약내역 조회 페이징 처리를 위한 전체 레코드 수 반환
public PageDTO<OrderInfoDTO>	selectOrderList (int user_no, int curPage) // 예약내역 조회
Public int	deletePickList (HashMap<String, Object> map) // 장바구니의 상품 삭제
public int	totalRecordPayment (int user_no) // 구매 완료 내역 조회 페이징 처리를 위한 전체 레코드 수 반환
public PageDTO<OrderInfoDTO>	selectOrderListPayment (int user_no, int curPage) // 구매 완료 내역 조회
Public int	deleteOrderProduct (HashMap<String, Object> map) // 예약상품 취소 시 예약내역 상품 삭제
Public int	updateProduct (HashMap<String, Object> map) // 상품 데이터 수정

iv. PickDAO

 PickDAO com.squirrel.dao	
	template: SqlSessionTemplate
	insertPick(dto: PickListDTO): int
	totalRecord(user_no: int): int
	selectPickList(user_no: int, curPage: int): PageDTO<PickListviewDTO>
	deletePick(list: List<Integer>): int
	selectPickofNo(map: HashMap<String,String>): PickOrderListDTO

Method Summary	
public int	insertPick (PickListDTO dto) // 상품 장바구니에 담기
public int	totalRecord (int user_no) // 장바구니 리스트 보기 페이징 처리를 위한 전체 레코드 수 반환
public PageDTO<PickListviewDTO>	selectPickList (int user_no, int curPage) // 장바구니 리스트 보기
public int	deletePick (List<Integer> list) // 장바구니에 상품 삭제
public PickOrderListDTO	selectPickofNo (HashMap<String, String> map) // 장바구니에서 구매진행시 해당 상품 조회













v. ProductDAO

ProductDAO com.squirrel.dao
▲ template: SqlSessionTemplate ● productInsert(dto: ProductDTO): int ■ totalRecord(map: HashMap<String,String>): int ● selectProduct(map: HashMap<String,String>, curPage: int): PageDTO<ProductListDTO> ● productRetrieve(p_id: String): ProductDTO ● productDecrease(insertVal: HashMap<String,Object>): int ■ totalRecordDealHistory(user_no: int): int ● selectDealHistory(user_no: int, curPage: int): PageDTO<ProductDealHistoryDTO> ■ totalRecordEditProduct(user_no: int): int ● selectEditProduct(user_no: int, curPage: int): PageDTO<ProductListDTO> ● editProduct(p_id: String): ProductListDTO ● deleteProduct(map: HashMap<String,String>): int ● updateProduct(dto: ProductDTO): int

Method Summary

public int	productInsert (ProductDTO dto) // 상품 등록
public int	totalRecord (HashMap<String, String> map) // 검색 조건에 따른 상품 조회 페이지 처리를 위한 전체 레코드 수 반환
public PageDTO<ProductListDTO>	selectProduct (HashMap<String, String> map, int curPage) // 검색 조건에 따른 상품 조회
public ProductDTO	productRetrieve (String p_id) // 상품 자세히 보기
public int	productDecrease (HashMap<String, Object> insertVal) // 상품 구매 시 구매 인원만큼 상품의 조인 인원 수 감소
public int	totalRecordDealHistory (int user_no) // 회원등급이 매니저인 회원이 본인이 등록한 상품에 대한 거래내역 조회 페이지 처리를 위한 전체 레코드 수 반환
public PageDTO<ProductDealHistoryDTO>	selectDealHistory (int user_no, int curPage) // 회원등급이 매니저인 회원이 본인이 등록한 상품에 대한 거래내역 조회
public int	totalRecordEditProduct (int user_no) // 회원등급이 매니저인 회원이 본인이 등록한 상품 관리 페이지 처리를 위한 전체 레코드 수 반환
public PageDTO<ProductListDTO>	selectEditProduct (int user_no, int curPage) // 회원등급이 매니저인 회원이 본인이 등록한 상품 관리 페이지
public ProductListDTO	editProduct (String p_id) // 상품내용 수정 페이지
public int	deleteProduct (HashMap<String, String> map) // 상품 삭제
public int	updateProduct (ProductDTO dto) // 상품 수정







vi. ReviewListDAO

 ReviewListDAO com.squirrel.dao	
	template: SqlSessionTemplate
	totalRecord(map: HashMap<String,String>): int
	reviewList(map: HashMap<String,String>, curPage: int): PageDTO<ReviewListDTO>
	insertReview(dto: CcScoreDTO): int
	selectDetail(score_no: int): CcScoreDTO
	rv_vcount(score_no: int): int
	selectNickname(score_no: int): ReviewListDTO
	updateReview(cdto: CcScoreDTO): int
	deleteReview(score_no: int): int
	myReviewtotalRecord(map: HashMap<String,String>): int
	selectMyReview(map: HashMap<String,String>, curPage: int): PageDTO<MyReviewDTO>

Method Summary

public int	totalRecord (HashMap<String, String> map) // 후기 게시판 검색에 따른 게시글 리스트 조회 페이징 처리를 위한 전체 레코드 수 반환
public PageDTO<ReviewListDTO>	reviewList (HashMap<String, String> map, int curPage) // 후기 게시판 검색에 따른 게시글 리스트 조회
public int	insertReview (CcScoreDTO dto) // 후기 게시글 등록
public CcScoreDTO	selectDeatil (int score_no) // 후기 게시글 자세히 보기
public int	rv_vcount (int score_no) // 게시글 조회수 증가
public ReviewListDTO	selectNickname (int score_no) // 후기 게시판 작성자 조건검색
public int	updateReview (CcScoreDTO cdto) // 회원 본인의 후기 게시글 수정
public int	deleteReview (int score_no) // 회원 본인의 후기 게시글 삭제
public int	myReviewtotalRecord (HashMap<String, String> map) // 회원 본인의 후기 게시글 리스트 조회 페이징 처리를 위한 전체 레코드 수 반환
public PageDTO<MyReviewDTO>	selectMyReview (HashMap<String, String> map, int curPage) // 회원 본인의 후기 게시글 리스트 조회








vii. ReCommentDAO

 ReCommentDAO com.squirrel.dao	
	template: SqlSessionTemplate
	insertComment(dto: ReCommentDTO): int
	selectRecomment(score_no: int): List<RecommentViewDTO>
	updateComment(map: HashMap<String,String>): int
	deleteComment(re_no: String): int

Method Summary

public int	insertComment (ReCommentDTO dto) // 게시글에 대한 댓글 등록
public List<RecommentViewDTO>	selectRecomment (int score_no) // 게시글에 대한 댓글 리스트 조회
public int	updateComment (HashMap<String, String> map) // 게시글에 대한 댓글 수정
public int	deleteComment (int re_no) // 게시글에 대한 댓글 삭제

viii. GolfccDAO

 GolfccDAO com.squirrel.dao	
	template: SqlSessionTemplate
	selectGolfccName(loc_id: String): List<GolfCcDTO>
	ccGolfScoreList(searchVal: HashMap<String,Object>): List<CcGolfScoreDTO>
	ccGolfScoreList(searchVal: HashMap<String,Object>, pageDto: PageDTO<CcGolfScoreDTO>): PageDTO<CcGolfScoreDTO>
	ccGolfScoreListCount(searchVal: HashMap<String,Object>): int
	getGolfccScoreOne(cc_id: String): CcGolfScoreDTO

Method Summary

public List<GolfCcDTO>	selectGolfccName (String loc_id) // 지역에 따른 골프장 조회
public List<CcGolfScoreDTO>	ccGolfScoreList (HashMap<String, Object> searchVal) // 골프장 평점 조회
public PageDTO<CcGolfScoreDTO>	ccGolfScoreList (HashMap<String, Object> searchVal, PageDTO<CcGolfScoreDTO> pageDto) // 골프장 평점 조회 페이지징
public CcGolfScoreDTO	getGolfccScoreOne (String cc_id) // 골프장 평점 조회 (json)

ix. LocationDAO



 LocationDAO
com.squirrel.dao
 template: SqlSessionTemplate
 locationList(): List<LocationDTO>

Method Summary

public List<LocationDTO>	locationList () // 지역 조회
--------------------------	------------------------------------

D. Other Class









i. LoginCheckInterCeptor

 com.interceptor.LoginCheckInterceptor
 preHandle(request: HttpServletRequest, response: HttpServletResponse, handler: Object): boolean

Method Summary

public boolean	preHandle (HttpServletRequest request, HttpServletResponse response, Object handler) // 로그인 권한, 혹은 특정 등급의 권한을 요구하는 페이지 요청 시, 세션의 로그인 정보를 통하여 권한 체크
----------------	--

ii. MailForUtil

 MailForUtil com.squirrel.util.email
 mailSender: JavaMailSender  memService: MemberService  aesManager: AESManager
 mailSending(map: HashMap<String,String>): ModelAndView  testMD5(str: String): String  sendMail(session: HttpSession): String  emailCheck(isTime: String, isCode: String, session: HttpSession, redData: RedirectAttributes): String

Method Summary

public ModelAndView	mailSending (HashMap<String, String> map) // 회원 비밀번호 찾기 시 임시 비밀번호 발송
public String	testMD5 (String str) // 이메일 인증을 위한 인증키 해시화
public String	sendMail (HttpSession session) // 이메일 인증 보내기, 회원정보+서버시간 암호화 하고 이것을 인증키로 해시화
public String	getGolfccScoreOne (String isTime, String isCode, HttpSession session, RedirectAttributes redData) // 회원이 이메일 인증시 인증시간과 서버시간 비교 후 인증키 비교하여 이메일 인증완료 하는 작업