



API 명세서

URL : localhost:8080/api

API 명세서

구현	구분	Controller	기능	메소드	URI	Request Body	성공 Response	실패 Response
<input type="checkbox"/>	사용자	user	회원 정보 조회	GET	/user/{username}		{ "status": 200, "message": "회원 정보 조회 성공", "isMe" : Boolean, "user": { "userName" : String, "followingCount" : Number, "followerCount" : Number, }, "articles" : [{ "articleId" : Number, "articleImage" : String, }], "likeFlowers" : [{ "tag" : String, "flowers" : [{ "flowerImage" : String, "subjectId" : Number, "subject" : String }], "likeArticles" : [{ "articleId" : Number, "articleImage" : String, "userName" : String, "likes" : Number }] }	{ "status": 404, "message": "존재하지 않는 사용자 입니다." }
<input type="checkbox"/>		user	팔로우 조회	GET	/user/{username}/follow		{ "status": 200, "message": "팔로우 조회 성공", "follower" : [{ "userName" : String, "userImage" : String, "isFollowing" : Boolean }], "following" : [{ "userName" : String, "userImage" : String }] }	{ "status": 404, "message": "", }
<input type="checkbox"/>		user	팔로우 추가 / 삭제	POST	/user/follow	{ "isFollow" : Boolean, "userId" : Number //상대방 id }	{ "status": 200, "message": "팔로우 추가/삭제 성공", }	
<input type="checkbox"/>		user	회원 정보 수정	PUT	/user	{ "headers": {Authentication: { }} "userImage" : String, "userName": String, "birthday": Date, "sex": Boolean }	{ "status": 200, "message": "회원 정보 수정 성공", }	{ "status": 403, "message": "올바르지 않은 접근입 니다.", }
<input type="checkbox"/>		user	회원 탈퇴	DELETE	/user		{ "status": 200, "message": "정상적으로 탈퇴되었습니다.", }	{ "status": 403, "message": "", }
<input type="checkbox"/>		auth	JWT	POST	/auth/login	{ "code" : String }	{ "status": 201, "message": "정상적으로 로그인되었습니다.", "token" : String }	{ "status": 409, "message": "로그인에 실패했습니 다.", }
<input checked="" type="checkbox"/>	커뮤니티	community	글 작성 - 이미 지 여러 개 받아 오는 결 로 수정 필요	POST	/community/article	{ "title" : String, "content" : String, "images" : [String], "subjects" : [Number] }	{ "status": 201, "message": "정상적으로 작성되었습니다.", "articleId" : Number }	
<input checked="" type="checkbox"/>		community	글 작성 //목록 목록 주 기위한 용도	GET	/community/article		{ "status": 200, "message": "정상적으로 수정되었습니다.", "subjects" : [{ "subjectId" : Number, "subjectName" : String }] }	
<input checked="" type="checkbox"/>		community	글 수정	PUT	/community/article/{articleId}	{ "title" : String, "content" : String, "images" : [String], "subjects" : [Number] }	{ "status": 200, "message": "정상적으로 수정되었습니다.", "articleId" : Number }	
<input checked="" type="checkbox"/>		community	글 삭제	DELETE	/community/article/{articleId}		{ "status": 200, "message": "정상적으로 삭제되었습니다.", "articleId" : Number }	
<input checked="" type="checkbox"/>		community	글 목록 조회	GET	/community/article/list		{ "status": 200, "message": "정상적으로 조회되었습니다.", "articles" : [{ "articleId" : Number, "articleImage" : String, "userId" : String, "articleTitle" : String, "articleContent" : String, "articleDate" : String, "tags" : [], "commentCount" : Number, "likeCount" : Number, "isFollow" : Boolean }] }	

구현	구분	Controller	기능	메소드	URI	Request Body	성공 Response	실패 Response
✓		community	글 상세 조회	GET	/community/article/{articleId}		{ "status": 200, "message": "정상적으로 조회되었습니다.", "article": { "articleUser": { "userId": Number, "userName": String, "userImage": String, "follower": Number, "following": Number, "isFollow": Boolean, "isMe": Boolean, }, "articleTitle": String, "articleContent": String, "articleImages": [], "articleDate": String, "tags": [{ //품목 태그 "subjectId": Number, "subjectName": String }], "isLike": Boolean, "likeCount": Number, "commentCount": Number, "comments": [{ "userId": Number, "userImage": String, "userName": String, "commentId": Number, "commentContent": String, "isMe": Boolean }] }	
✓		community	좋아요 추가/삭제	POST	/community/article/{articleId}/like	{ "isLike": Boolean }	{ "status": 200, "message": "정상적으로 수정되었습니다.", }	
✓		community	댓글 작성	POST	/community/article/{articleId}/comment	{ "commentContent": String }	{ "status": 201, "message": "정상적으로 작성되었습니다.", }	
✓		community	댓글 수정	PUT	/community/article/{articleId}/comment	{ "commentId": Number, "commentContent": String }	{ "status": 200, "message": "정상적으로 수정되었습니다.", }	
✓		community	댓글 삭제	DELETE	/community/article/{articleId}/comment	{ "commentId": Number }	{ "status": 200, "message": "정상적으로 삭제되었습니다.", }	
✓		community	인기글 > 배치 돌릴때 인기글 토크 날짜 추가	GET	/community/popular		{ "status": 200, "message": "정상적으로 조회되었습니다.", "articles": [{ "articleId": Number, "articleImages": [], "userId": String, "articleTitle": String, "articleContent": String, "articleDate": Date, "tags": [], "commentCount": Number, "likeCount": Number }] }	
✓		flower	품목 상세 페이지 조회	GET	/flower/{subjectId}		{ "status": 200, "message": "꽃 품목 상세 페이지 조회 성공", "flower": { "subjectId": Number, "subjectName": String, "subjectLanguage": String, "kinds": [{ "kindId": Number, "kindName": String, "kindImage": String, "tag": [{ "tagId": Number, "tagName": String, "tagStatus": Boolean },] }, "articles": [{ "articleId": Number, "articleImage": String, "userName": String, "likeCount": Number }] }	
✓		flower	컬렉션 (태그) 추가 / 삭제	POST	/flower/{subjectId}	{ "kindId": Number, "tagId": Number, "tagStatus": Boolean }	{ "status": 200, "message": "" }	
✓		flower	오늘의 꽃	GET	/flower/daily		{ "status": 200, "message": "", "dailyFlowerName": String, "dailyFlowerLanguage": String, "dailyFlowerContent": String, "dailyFlowerImage": String }	
✓		flower	꽃 검색	GET	/flower/search/{searchText}	{ "searchText": String }	{ "status": 200, "message": "", "flowers": [{ "subjectId": Number, "subjectName": String, "kindId": Number, "kindName": String, "kindImage": String }] }	
□	꽃 추천	command	편지 추천 - Spring	POST	/command/letter	{ "content": String }	{ "status": 200, "message": "편지 기반 꽃 추천 성공", "flower": { "subject": String, "subjectLanguage": String }	
□		command	편지 추천 - Django					
□		command	상황 추천 - Spring	GET	/command/situation		{ "status": 200, "message": "상황 기반 꽃 추천 성공", "flowers": [{ "tag": { "tagId": Number, "tagName": String }, "subjectId": Number, "kindId": Number, "kindImage": String }] }	
□		command	상황 추천 - Django					
□		command	게시글 추천 - Spring	GET	/command/article		{ "status": 200, "message": "게시글 기반 꽃 추천 성공", "articles": [{ "articleId": Number, "articleImage": String, "userName": String, "likeCount": Number }] }	
□		command	게시글 추천 - Django					