

API 설계

API 설계							필터	정렬	...	새로 만들기			
As 서비스	🕒 접근 주체	🕒 메소드(L)	🕒 체크	📄 설명(Description)	📄 통합 자원 식별자(L)	📄 DTO	📄 Response Data Type	📄 Response Data	📄 추가 설명	🕒 FE 담당자	🕒 BE 담당자	+	...
회원 서비스	비회원	POST	🟢 Done	회원가입	/members/signup	@RequestBody String email String password	boolean	{ "result": true, "resultCode": 200, "resultMessage": "회원가입 성공" }			박시형		
회원 서비스	비회원	GET	🟢 Done	아이디 중복검사	/members/duplicate	@RequestParam String email	boolean	{ "result": true, "resultCode": 200, "resultMessage": "사용할 수 없는 아이디입니다." }			박시형		
회원 서비스	비회원	POST	🟢 Done	로그인	/members/login	@RequestBody String email String password	"memberId": Long, "email": String, "password": String, "notionToken": String, "roles": Array<Object>, "token": String	{ "result": { "memberId": 1, "email": "test1234@naver.com", "password": " [bcrypt]52a8103Mhndop0YuVnR495nUDKt9Q u4sqTZuhy1D74RdUJCxGYN5QbC", "notionToken": "token", "roles": [{ "name": "ROLE_USER" }], "token": " ["access_token": " eyJhbGciOiJIUzI1NiIsInR5cyIzadWiOiJ0ZDZlODZlODM0MTcN C3lnIjE6VjIpbjYyVW11IjoLa9MRV9VUUV5ln 1dCjpwXWQIQE2ODE2Of2MjUslmV4C0BMTY 4MTcwMjYyNjX0e4929yl- A9ha7mZgSHxa8aBzwcCxtmni6OMM_6h8r", "refresh_token": "iasodfmoaskdfmusikask" } }, "resultCode": 200, "resultMessage": "로그인 성공" }			박시형		
회원 서비스	회원	POST	🟢 Done	로그아웃	/members/logout	@RequestParam Long memberId @RequestBody String access_token String refresh_token	"memberId": Long "access_token": String "refresh_token": String	{ "result": true, "resultCode": 200, "resultMessage": "로그아웃 성공" }			박시형		
회원 서비스	회원	GET	🟢 Done	Access 토큰 재발급	/members/refresh	@RequestBody String access_token String refresh_token	"access_token": String "refresh_token": String	{ "result": { "access_token": " eyJhbGciOiJIUzI1NiIsInR5cyIzadWiOiJ0ZDZlODZlODM0MTcN C3lnIjE6VjIpbjYyVW11IjoLa9MRV9VUUV5ln 1dCjpwXWQIQE2ODE2Of2MjUslmV4C0BMTY 4MTcwMjYyNjX0e4929yl- A9ha7mZgSHxa8aBzwcCxtmni6OMM_6h8r", "refresh_token": "iasodfmoaskdfmusikask" } }, "resultCode": 200, "resultMessage": "refresh 토큰이 만료되지 않아 access 토큰이 재발급 되었습니다." }			박시형		
인증 서비스	회원	POST	🟢 Done	Tistory 토큰 발급	/oauth/tistory	@RequestBody String code	"access_token": string	{ "result": "aaa", "resultCode": 200, "resultMessage": "정상적으로 카카오 액세스 토큰 받기 성공" }			임지현		
인증 서비스	회원	POST	🟢 Done	Github 토큰 발급	/oauth/git	@RequestBody String code	"access_token": string	{ "result": "aaa", "resultCode": 200, "resultMessage": "정상적으로 깃허브 액세스 토큰 받기 성공" }	1. 접근 주체가 회원 인 API 요청은 모두 HTTP 요청 헤더에 (Authorization : Access-Token)을 포함시켜야한다.		이광용		
인증 서비스	회원	POST	🟢 Done	Notion 토큰 발급	/oauth/notion	@RequestBody String code	"access_token": string	{ "result": "aaa", "resultCode": 200, "resultMessage": "정상적으로 노션 액세스 토큰 받기 성공" }			이광용		
발행 서비스	회원	POST	🟢 Done	Tistory 발행 및 수정	/contents/post	{ { "tistoryId": "10", "type": "html", "blogName": "appleai", "requestLink": " https://www.notion.so/e631cd8ffcaad41c8b0 a0b10468e239bd", "visibility": "3", "categoryName": "1142468", "tagList": "최종, 발행, 성공, 테스트", "status": "발행요청" } }	"tistoryId": long, "type": string "blogName": string "requestLink": string "visibility": int 0 (비공개) 3 (공개) "categoryName": string "tagList": string "status": string "발행요청" "수정요청" "발행 완료" "발행실패" "수정실패"	{ "result": { "headers": [], "body": [{ "requestIndex": 1, "resultCode": 200, "resultMessage": "[발행완료] Tistory 게시물 발 행이 정상적으로 완료되었습니다.<2>" }], "statusCode": "OK", "statusCodeValue": 200 }, "resultCode": 200, "resultMessage": "Nogani Tistory POST API가 정 상적으로 실행되었습니다." }			강주형		
발행 서비스	회원	POST	🟡 Not start	Tistory 발행 예약	/contents/reservation	@PathVariable final String platform (ex: Github, Tistory, Velog) @RequestBody TimeDto int minute int hour int day int month int week int year			TimeDto 형식 1. 분(Minutes) • 값 범위 : 0 ~ 59 2. 시(Hours) • 값 범위 : 0 ~ 23 3. 일(Day of month) • 값 범위 : 1 ~ 31 4. 월(Month) • 값 범위 : 1 ~ 12 또는 JAN ~ DEC 5. 주(Day of week) • 값 범위 : 0 ~ 6 또는 SUN ~ SAT 6. 년(Year) • 값 범위 : 성립 또는 1970 ~ 2099 위 구성요소를 전달받아 Cron 형식으로 자주 예약어한다.				
발행 서비스	회원	POST	🟢 Done	Tistory 발행 이력 조회(무한스크롤)	/contents/tistory	@RequestBody // filter, word, status와 같은 조건이 존재하는 경우 { "lastTistoryId": -1, "pageSize": 1, "filter": "string", "word": "string", "statusList": ["string"] } // 조건 없이 단순 조회인 경우 { "lastTistoryId": -1, "pageSize": 5, "filter": "", "word": "", "statusList": [] }	"tistoryId": Number, "postId": Number, "blogName": String, "title": String, "categoryName": Number, "visibility": Number, "status": String, "requestLink": String, "responseLink": String, "tagList": String	{ "result": [{ "tistoryId": 1, "blogName": "개발자들을 위한 백문몰이", "title": "당신을 위한 프로젝트", "categoryName": "발고리용 언박싱", "visibility": 3, "status": "발행요청", "requestLink": " https://mjmeon94.notion.site/08e08d36ac354 4a7ab21f141232b311b?v=92376629a4d74c356d47228fbd9edcce", "responseLink": " https://mjmeon94.notion.site/08e08d36ac354 4a7ab21f141232b311b?v=92376629a4d74c356d47228fbd9edcce", "tagList": "당신을 위한 프로젝트", "modifiedDate": "2023-03-20", }, { "tistoryId": 2, "blogName": "Python 후기", "title": "자세하게 정리한 언박싱", "categoryName": "개발자들을 위한 놀이리 추 진", "visibility": 3, "status": "발행완료", "requestLink": " https://mjmeon94.notion.site/08e08d36ac354 4a7ab21f141232b311b?v=92376629a4d74c356d47228fbd9edcce", }] }	1. lastTistoryId가 -1 인 경우에만 BlogNameList, CategoryList가 반환된다.		임지현		

