




API 명세서

Short cuts [[<페이지 이름> : DTO (페이지 불러오기)

1. DTO 표에서 페이지 정보 작성
2. Request, Response에서 [[<페이지 이름> 을 통하여 작성한 DTO 불러오기

API 명세서

Status	Aa 기능	HTTP method	URI	설명	Request	Response
Not started	<u>회원</u>					
Not started	<u>Untitled</u>	POST	/api/user/login	로그인		
Not started	<u>Untitled</u>	POST	/api/user/sign-up	회원가입	 <u>signupRequestDto</u>	
Not started	<u>Untitled</u>	GET	/api/user/nickname	닉네임 중복 확인		
Not started	<u>Untitled</u>	PUT	/api/user/nickname	닉네임 수정		
Not started	<u>Untitled</u>	POST	/api/user/write/{idol-id}	특정 아이돌에 응원 한마디 - 아이돌에 응원한마디 추가 - 유저에 Datetime 추가(마이페이지에서 사용) - 하루에 1번 가능		

Status	Aa 기능	☑ HTTP method	≡ URI	≡ 설명	≡ Request	≡ Response
Not started	<u>Untitled</u>	GET	/api/user/comment	관심 아이돌 별 응원을 남긴 Datetime 이 담긴 리스트		
Not started	<u>Untitled</u>					
Not started	<u>아이돌</u>					
Not started	<u>Untitled</u>	GET	/api/idol/list	아이돌 리스트 - 회원 가입 시 아이돌 목록 표시 - RDB와 NoSQL 둘 다 넣자		
Not started	<u>Untitled</u>	GET	/api/idol/{idol-id}	아이돌 상세정보 - 사진, 이름, sns - 관심 아이돌 여부		 <u>idolDetailResponseDto</u>
Not started	<u>Untitled</u>	GET	/api/idol/{idol-id}/pos-neg	금부정지수(일별 or 시간별) 리스트 - 추후 시간이 남으면 실시간으로 전환		
Not started	<u>Untitled</u>	GET	/api/idol/{idol-id}/comment	응원 한 마디 리스트 - 30개 정도		
Not started	<u>Untitled</u>	GET	/api/idol/{idol-id}/keyword	키워드 정보 리스트		
Not started	<u>Untitled</u>					
Not started	<u>관심 아이돌</u>					




Status	Aa 기능	Ⓜ HTTP method	≡ URI	≡ 설명	≡ Request	≡ Response
Not started	<u>Untitled</u>	GET	/api/interest/{nickname}/list	메인페이지 > 나의 관심 아이돌 리스트 - 남의꺼 훑쳐보는거 처리		
Not started	<u>Untitled</u>	GET	/api/interest/{nickname}/{idol-id}	나의 관심 아이돌 상세 정보 - 아이돌 별 사진, 이름, 긍정지수, 키워드 리스트		
Not started	<u>Untitled</u>	DELETE	/api/interest/{nickname}/{idol-id}	관심 아이돌 삭제		
Not started	<u>Untitled</u>					
Not started	<u>Peak</u>					
Not started	<u>Untitled</u>	GET	/api/peak/top	실시간 랭킹 8위까지 리스트		 <u>idolInfoResponseDto</u>
Not started	<u>Untitled</u>	GET	/api/peak/{idol-id}	특정 아이돌의 시간 별 랭킹 차트 - 랭킹 페이지 (30 분마다) - Response body에 변동도 줘야 함(캐싱도)		
Not started	<u>Untitled</u>	GET	/api/peak/weekly/{idol-id}	특정 아이돌의 주간 랭킹 차트 - 차트 페이지, 상세 페이지(1주일, 일별)		
Not started	<u>Untitled</u>	GET	/api/peak/current/{idol-id}	특정 아이돌의 현재 랭킹 순위 - 아이돌 상세 페이지		

Status	Aa 기능	☑ HTTP method	≡ URI	≡ 설명	≡ Request	≡ Response
Not started	<u>Untitled</u>					
Not started	<u>Youtube</u>					
Not started	<u>Untitled</u>	GET	/api/youtube	트렌딩 유튜브 리스트		
Not started	<u>Untitled</u>	GET	/api/youtube/{idol-id}	특정 아이돌의 트렌딩 유튜브 리스트		
Not started	<u>Untitled</u>					
Not started	<u>News</u>					
Not started	<u>Untitled</u>	GET	/api/news	트렌딩 뉴스 리스트		
Not started	<u>Untitled</u>	GET	/api/news/{idol-id}	특정 아이돌의 트렌딩 뉴스 리스트		
Not started	<u>Untitled</u>	GET	/api/news/{idol-id}/{keyword}	특정 아이돌+키워드의 트렌딩 뉴스 리스트		
Not started	<u>Untitled</u>					
Not started	<u>개인 사용 데이터</u>					
Not started	<u>Untitled</u>	GET	/api/personal/{user-id}/{idol-id}	특정 아이돌의 평균덕력과 나의 순위 특정 아이들에 대한 나의 덕력 - 덕력: 클릭, 응원글 수, 방문시간 등으로 계산한 점수		

Status	Aa 기능	⌵ HTTP method	≡ URI	≡ 설명	≡ Request	≡ Response
Not started	<u>Untitled</u>	GET	/api/personal/click/{user-id}/{idol-id}	유저가 특정 아이돌을 클릭한 횟수와 모든 유저의 평균 클릭 횟수 반환		
Not started	<u>Untitled</u>	GET	/api/personal/comment/{user-id}/{idol-id}	유저가 특정 아이돌에 남긴 응원 갯수와 모든 유저의 평균 응원 갯수 반환		
Not started	<u>Untitled</u>	GET	/api/personal/stay/{user-id}/{idol-id}	유저가 특정 아이돌을 방문한 시간과 모든 유저의 평균 시간 반환		
Not started	<u>Untitled</u>					

▼ DTO TABLE

DTO(Data Transfer Object)

Aa 이름	⌵ 태그
 <u>signupRequestDto</u>	API Request
 <u>idolInfoResponseDto</u>	API Response
 <u>idolDetailResponseDto</u>	API Response

▼ 컨닝 답변

- 한쪽에 보관했다고 해서 그 뒤에 가공될 데이터도 거기에만 있어야 될 필요는 없을 것 같다
- 인증 시스템과 권한이 섞인 느낌?
 - 인증 및 인가 파트만 잘 해놓으면 따로 있어도 상관 없음

- 용도가 나뉨

- RDB
 - 정제되고 깔끔하고 확실한 정보를 차곡차곡 쌓아놓는 성격
 - 유실되면 안되고 백업 자주하는 중요한 데이터
 - 고비용 & 안전
 - 속도에 크게 저하가 없는 한
 - 엄청 빈번하게 가져올 거 아님
 - 적당히 정제된 것들을 넣자
 - 중요한 데이터는 고비용의 RDB에 있어야 되는거 아닐까 라는 생각

- NoSQL
 - 막 쌓아놓고 쓰는 성격
 - 스키마 제약없이 빠르게
 - 원본 그대로 쓸 거 아님
 - 자주 쓰고 그러는 느낌은 아님
 - 마구마구 적재하고 나중에 추출할 것들은 NoSQL을 쓰자

- 결론은 분리되어야 맞다고 생각하신다