REST API v1 (21.09.09)

	Category	Summary	HTTP Method	URI	Request Header	Params	Request Body	Request Description	Success Code	Response Header	Response Entity	Response Description
ex)	로그인	로그인	POST	/auth/login	X	X	userId : String password : String		200	Token	X	
	로그인	로그인	POST	/auth/login			{ userId : String, password : String, }		200	Token	X	
	회원가입	회원가입	POST	/auth/user			{ userId : String, nickname: String, password : String }		201			
		아이디 중복체크	GET	/auth/user/valDuplicated			{ val : String('id'), content : String }		200		{ result : Boolean }	1.ex { result : True }
		닉네임 중복체크	GET	/auth/user/valDuplicated			{ val : String('nickname'), content : String }		200		{ result : Boolean }	1.ex { result : True }
		유저 직업 리스트 반환	GET	/auth/user/job					200		{ contents : { [String, String, ...] } }	1. ex { contents : { ['의사','소방관','학생 ...] } }
		유저 mbti 저장	POST	/auth/user/profile/mbti	Token		{ result : String }		200			
		유저 tendency 검사를 위한 책 리스트 반환	GET	/auth/user/profile/tendency					200		{ contents: [{ id: INT, title : String, author : String, story: String, img : String }, {}, {}...] }	1. ex { contents: [{ id: 도서 테이블 pk, title : 도서 제목, author : 도서 작가, story: 도서 줄거리, img : 도서 표지 이미지 URL }, {}, {}...] }
		유저 tendency 저장	POST	/auth/user/profile/tendency	Token		{ bookIds : [{ id : INT, check : INT, }, {}, {} ...] }	1. check == 0 : 보고 싶어요 2. check == 1 : 그저 그래요 3. check == 2 : 보기 싫어요	200			
	마이페이지	정보 표시	GET	/auth/user/profile	Token				200		{ content: { id : INT, userId : String, nickname: String, mbti : String, tendency : String, job : String, age : INT, gender : String } }	
		책 히스토리	GET	/auth/user/profile/history					200		미정	
		추천 마인드맵 (마이페이지, 중복)	GET	/book/recommend/mypage	Token				200		{ userRecommend : [{ userRecommendCnt : INT id : INT, title : String, author : String, story : String, img : String, genre : [String, String, ..], topic : [String, String ...] }, {}, {} ...], bookRecommend : [{ userRecommendCnt : INT, id : INT, title : String, author : String, story : String, img : String, genre : [String, String, ...], topic : [String, String, ...] }, {}, {} ...] }	1. userRecommend[userRecommendCnt] - 도서 기반 추천 결과가 어떤 도서를 기준으로 나왔는지 알려주기 위한 것 유저 기반 추천 결과 A, B, C .... 가 있다면 도서 기반 추천 결과는 A 도서와 유사한 장르의 도서가 된다. userRecommendCnt 는 INT 로 0~10 사이로 정해진다. 2. ex { userRecommend : [{ userRecommendCnt : 2 id : 도서 테이블 pk, title : 도서 제목, author : 작가 이름, story : 줄거리 ..., img : 책 표지 이미지 URL, genre : [추리, 한국소설, ..], topic : [미스터리, 살인사건...] }, {}, {} ...], bookRecommend : [{ userRecommendCnt : 2 (다른 결과와 구분되는 key 라고 생각), id : 도서 테이블 pk, title : 도서 제목, author : 작가 이름, story : 줄거리 ..., img : 책 표지 이미지 URL, genre : [추리, 한국소설, ...], topic : [미스터리, 살인사건, ...] }, {}, {} ...] }
	도서 관리	상세 보기	GET	/book/detail/{bookId}					200		{ detail : { id: INT, title: String, author : String, publisher : String, story : String, img : String, genre : [String, String...], topic : [String, String ...] }, reviews : { [id: INT, userId : String, score : INT, content : String], [] ... }, likes : { [id: INT, userId : String], []... }, review_cnt : INT, like_cnt : INT }	1. ex { detail : { id: 도서테이블 pk, title: 도서 제목, author : 도서 작가, publisher : 출판사, story : 줄거리 ..., img : 도서 표지 이미지 URL, genre : [추리, 한국소설...], topic : [미스터리, 살인사건...] }, reviews : [{ id : 리뷰테이블 pk, userId : ssafy5, score : 5, content : '두근두근하네요' }, {} ...], likes : [{ id : 좋아요 테이블 pk, userId : ssafy5 }, {}...], review_cnt : INT (리뷰 숫자), like_cnt : INT (좋아요 숫자) }
		리뷰 작성하기	POST	/book/review/{bookId}	Token		{ score : INT, content : String }		201			
		좋아요	POST	/book/like/{bookId}	Token				201			
	추천 알고리즘 부분은 아직 잘 모르는 분야라서 일단 이렇게 했습니다. 개발하실 때 상황에 맞게 수정하시고 문서 수정 부탁드립니다.											
	추천알고리즘	유저 기반 추천	GET	/book/recommend/user	Token		{ type : String }	1. type == 'like' : 좋아요 기반 추 2. type == 'review' : 리뷰 기반 추	200		{ type : String, contents: [{ id : INT, title : String, author : String, publisher : String, story : String, img : String, genre : [String, String, ...], topic : [String, String, ...], review_cnt : INT, like_cnt : INT }, {}, {} ...] }	1. ex { type : like, contents: [{ id : , title : String, author : String, publisher : String, story : String, img : String, genre : [String, String, ...], topic : [String, String, ...], review_cnt : INT, like_cnt : INT }, {}, {} ...] }
		책(아이템)기반 추천	GET	/book/recommend/book	Token		{ type : String }	1. type == 'genre' : 장르 기반 추 2. type == 'author' : 작가 기반 추	200		{ type : String, contents: [{ id : INT, title : String, author : String, publisher : String, story : String, img : String, genre : [String, String, ...], topic : [String, String, ...], review_cnt : INT, like_cnt : INT }, {}, {} ...] }	1. ex { type : like, contents: [{ id : , title : String, author : String, publisher : String, story : String, img : String, genre : [String, String, ...], topic : [String, String, ...], review_cnt : INT, like_cnt : INT }, {}, {} ...] }

REST API v1 (21.09.09)

Category	Summary	HTTP Method	URI	Request Header	Params	Request Body	Request Description	Success Code	Response Header	Response Entity	Response Description

	추천 마인드맵 (마이페이지, 중복)	GET	/book/recommend/mypage	Token				200		{
										userRecommend : [
										{
										userRecommendCnt : INT
										id : INT,
										title : String,
										author : String,
										story : String,
										img : String,
										genre : [String, String, ..],
										topic : [String, String ...]
										},
										{}, {} ...
										],
										bookRecommend : [
										{
										userRecommendCnt : INT,
										id : INT,
										title : String,
										author : String,
										story : String,
										img : String,
										genre : [String, String, ...],
										topic : [String, String, ...]
										},
										{}, {} ...
										]
										}
											1. userRecommend[userRecommendCnt]
											- 도서 기반 추천 결과가 어떤 도서를 기준으로 나왔는지 알려주기 위한 것
											유저 기반 추천 결과A, B, C.....가 있다면
											도서 기반 추천 결과는 A 도서와 유사한 장르의 도서가 뜬다.
											userRecommendCnt 는 INT 로 0~10 사이로 정해진다.
											
											2. ex
											{
											userRecommend : [
											{
											userRecommendCnt : 2
											id : 도서 테이블 pk,
											title : 도서 제목,
											author : 작가 이름,
											story : 줄거리 ...,
											img : 책 표지 이미지 URL,
											genre : [추리, 한국소설, ..],
											topic : [미스터리, 살인사건...]
											},
											{}, {} ...
											],
											bookRecommend : [
											{
											userRecommendCnt : 2 (다른 결과와 구분되는 key 라고 생각),
											id : 도서 테이블 pk,
											title : 도서 제목,
											author : 작가 이름,
											story : 줄거리 ...,
											img : 책 표지 이미지 URL,
											genre : [추리, 한국소설, ...],
											topic : [미스터리, 살인사건, ...]
											},
											{}, {} ...
											]
											}

게시판											