### 테이블 생성

새 로그프레소 테이블을 생성합니다.

#### HTTP 요청

POST /api/database/tables

cURL 예시

curl -H "Authorization: Bearer <API\_KEY>" \ -d "table=demo" \ -d "type=v3p" \ -d "primary\_configs={""compression"":""snappy"", ""layout"":""columnar""}" \ -X POST https://HOSTNAME/api/database/tables

요청 매개변수

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 키 | 필수 | 타입 | 설명 | 비고 |
| table | O | 문자열 | 테이블명 |  |
| type | O | 문자열 | 스토리지 엔진 | 예: v3p |
| base\_path | X | 문자열 | 데이터 파티션 경로 |  |
| primary\_configs | X | 문자열 키/값 | 스토리지 엔진 설정 |  |
| replica\_configs | X | 문자열 키/값 | 복제 엔진 설정 |  |
| metadata | X | 문자열 키/값 | 메타데이터 |  |

* primary\_configs: v3p 스토리지 엔진의 경우 아래의 설정을 지원합니다.
* compression: deflate 혹은 snappy
* layout: row 혹은 columnar. 대부분의 경우 columnar를 권장합니다.
* crypto: 암호화 프로파일 이름
* metadata: 아래의 키/값 쌍을 주로 사용합니다.
* logparser: table 명령어에 기본 적용할 파서 식별자를 지정합니다.
* app\_id: 어떤 앱에 속한 테이블인지 기술합니다.

#### 정상 응답

{}

#### 오류 응답

권한이 없는 경우

{ "error\_code": "security-violation", "error\_msg": "logdb management is not allowed to user"}

테이블 이름이 허용되지 않는 경우

{ "error\_code": "invalid-argument", "error\_msg": "'table' must begin with a letter and may contain alphanumeric and underscore characters: a\*"}

테이블이 이미 존재하는 경우

{ "error\_code": "table-already-exists", "error\_msg": null}

스토리지 엔진을 잘못 지정한 경우

{ "error\_code": "general-error", "error\_msg": "java.lang.UnsupportedOperationException, not supported engine: v1"}