



# Mini Lex

검색어가 아니라 상황으로 찾는 법령 어시스턴트

## 개발자설명서 v0.2

lawdata-wellsa 7종 인제스트와 Q&A 검색

## 아키텍처

```
Next.js -> FastAPI -> PostgreSQL/pgvector
      |
      Celery
      |
lawdata-wellsa
```

## 파이프라인

scrape -> fetch\_body -> convert -> ingest

- 공개 원천 수집
- 본문 확보
- Markdown 정규화
- LawVersion/LawVersionArticle 적재

## 7종 service

| kind           | service                   |
|----------------|---------------------------|
| law            | law_service.py            |
| regulate       | legalize_kr_service.py    |
| precedent      | precedent_service.py      |
| interpretation | interpretation_service.py |
| constitution   | constitution_service.py   |
| localrule      | localrule_service.py      |
| treaty         | treaty_service.py         |

## 통합 인제스트

```
from app.services.lawdata_ingest import ingest_all  
ingest_all(db, only=None, dry_run=False)
```

Celery task: `workers.pipeline_tasks.ingest_lawdata_all`

## LAWDATA\_ROOT

```
LAWDATA_ROOT=/app/data/lawdata-wellsa  
LAW_API_OC=wellsa-ai
```

운영과 로컬 경로 차이를 env 하나로 흡수한다.

## DB schema

장문 판례와 자치법규를 위해 본문은 **TEXT**.

```
ALTER TABLE law_version_articles  
  ALTER COLUMN content TYPE TEXT;
```

## Q&A 검색

- 키워드 추출
- 동의어 확장
- DB 후보 검색
- 시민 시나리오 preferred law boost
- source\_articles/source\_counts 반환

## 임베딩 모델

| 모델                      | 상태    |
|-------------------------|-------|
| nomic-embed-text-v2-moe | 현행    |
| bge-m3                  | 듀얼 후보 |
| qwen3-embedding-0.6b    | 벤치 후보 |
| SHA256 fallback         | 장애 폴백 |

## 회귀 시나리오

1. 딥페이크 삭제
2. 임금체불
3. 부당해고
4. 개인정보 유출
5. 스토킹

## 운영 smoke

```
curl /api/v1/health  
curl /api/v1/lawdata-stats  
curl -X POST /api/v1/chat/demo-query
```