없는 회차·중복 ID·URL 누락을 DB에서 막아봤습니다

지피터스 24기 · 자료 정리 2026-10-14 / 기존 실습 근거 2026-10-13 / 단독 재현 코드는 10-14 실행 확인

SQLite가 거절한 세 입력

저장 규칙을 DB 제약으로 옮겨 실제 거절 여부를 확인한 실습입니다. 원고 품질 같은 판단은 별도로 두고, 참조 관계·중복 ID·게시 URL의 최소 조건을 SQLite 제약으로 확인했습니다.

시도	기대하는 엔진 응답
없는 E99에 원고 연결	FOREIGN KEY constraint failed
URL 없이 published 상태 저장	CHECK constraint failed
E02 ID 중복 저장	UNIQUE constraint failed: episode.id

직접 확인

`python3 cto-replay.py`를 실행합니다. 별도 파일 설치 없이 Python 표준 sqlite3의 메모리 DB를 사용합니다. 세 결과 모두 rejected=true인지 확인합니다.

published_url의 LIKE 조건은 https:// 접두어만 봅니다. 실제 URL 유효성·게시 반영·읽기 권한을 보장하지 않습니다.

10월 14일 단독 재현 출력


```
[
  {
    "case": "foreign_key",
    "error": "FOREIGN KEY constraint failed",
    "rejected": true
  },
  {
    "case": "published_without_url",
    "error": "CHECK constraint failed: state!='published' OR published_url LIKE 'https://%'",
    "rejected": true
  },
  {
    "case": "duplicate_episode",
    "error": "UNIQUE constraint failed: episode.id",
    "rejected": true
  }
]
```

단독 재현 코드: cto-replay.py

긴 줄은 지면에 맞춰 자동 줄바꿈됩니다. 코드 복사 시 줄바꿈·들여쓰기를 확인하세요.

```
"""Three rejected writes in an isolated SQLite DB."""
import sqlite3, json
SCHEMA = """PRAGMA foreign_keys=ON;
CREATE TABLE IF NOT EXISTS episode(id TEXT PRIMARY KEY,
    title TEXT NOT NULL,
    question TEXT NOT NULL);
CREATE TABLE IF NOT EXISTS revision(id TEXT PRIMARY KEY,
    episode_id TEXT NOT NULL REFERENCES episode(id),
    label TEXT NOT NULL,
    source TEXT NOT NULL);
CREATE TABLE IF NOT EXISTS action(episode_id TEXT PRIMARY KEY REFERENCES episode(id),
    state TEXT NOT NULL CHECK(state IN ('draft',
    'review',
    'approved',
    'published')),
    next_action TEXT NOT NULL,
    evidence TEXT NOT NULL DEFAULT '',
    published_url TEXT NOT NULL DEFAULT '',
    CHECK(state!='published' OR published_url LIKE 'https://%'));
"""
c = sqlite3.connect(':memory:')
c.executescript(SCHEMA)
c.execute('INSERT INTO episode VALUES(?,?,?)',
          ('E02','Public sample','Does the sum match?'))
cases = [
 ('foreign_key', 'INSERT INTO revision VALUES(?,?,?,?)',
  ('R-missing','E99','sample','source.md')),
 ('published_without_url', 'INSERT INTO action VALUES(?,?,?,?,?)',
  ('E02','published','check','','')),
 ('duplicate_episode', 'INSERT INTO episode VALUES(?,?,?)',
  ('E02','duplicate','question'))]
results = []
for label, sql, args in cases:
    try:
        c.execute(sql,args)
    except sqlite3.IntegrityError as err:
        results.append({'case':label,'error':str(err),'rejected':True})
    else:
        raise AssertionError(label+' was not rejected')
print(json.dumps(results,ensure_ascii=False,indent=2))
```

기존 임시 DB 검사 5건: 재접속 후 저장 유지, 없는 회차 참조 거절, URL 없는 게시상태 거절, 중복 ID 거절, 제목 변경 후 관계 유지.