

# 외환 계좌 데이터베이스 만들기

## 1) DB 명세서 (Database Specification)

### 1.1 목적/범위

- 목적: 사용자 정보와 사용자별 보유 외환액(통화별 잔액)을 저장/조회하는 실습용 DB.
- 범위: 사용자 등록, 통화(ISO 4217) 마스터 관리, 사용자×통화 보유 잔액 관리.

### 1.2 개체(Entity)

- `users` : 사용자 기본 정보
- `currencies` : 통화 마스터 (ISO 코드, 표시명, 소수 자리)
- `fx_holdings` : 사용자별 통화 잔액(1명:다통화)

## 1.3 설계 원칙

- 정규화 최소화: 통화 속성은 마스터 테이블로 분리, 보유는 교차(유저×통화)로 1행.
- 무결성 보장: 외래키(FK)로 사용자·통화 존재성 보장, `(user_id, currency_code)` 유니크.
- 삭제 규칙: 사용자 삭제 시 해당 사용자의 보유(`fx_holdings`) 자동 삭제(CASCADE).

## 1.4 명명 규칙(요약)

- 테이블: 소문자 복수형(`users`, `currencies`, `fx_holdings`)
- 기본키: 단수명`_id`(`user_id`, `holding_id`)
- 외래키: 테이블단수`_id`(`user_id`), `currency_code`
- 수치: 금액은 `DECIMAL(24, 8)` 고정 소수

## 2) 테이블 명세서 (Table Specifications)

### 2.1 users

사용자 기본 정보

컬럼명	타입	NULL	기본값	제약	설명
user_id	BIGINT UNSIGNED	NO	AUTO_INCREMENT	PK	사용자 ID
email	VARCHAR(255)	NO		UNIQUE	로그인/식별 이메일
full_name	VARCHAR(100)	NO			사용자 이름
status	TINYINT	NO	1		활성/비활성 1/0
created_at	DATETIME	NO	CURRENT_TIMESTAMP		생성 시각

인덱스

- PK(user\_id), UNIQUE(email)

## 2.2 currencies

통화 마스터 (ISO 4217)

컬럼명	타입	NULL	기본값	제약	설명
currency_code	CHAR(3)	NO		PK	ISO 4217 코드(예: KRW, USD)
currency_name	VARCHAR(50)	NO			통화명
minor_unit	TINYINT UNSIGNED	NO			소수 자리수(USD=2, KRW/JPY=0 등)

인덱스

- PK(currency\_code)

## 2.3 fx\_holdings

사용자별 보유 외환(유저×통화 유니크)

컬럼명	타입	NULL	기본값	제약	설명
holding_id	BIGINT UNSIGNED	NO	AUTO_INCREMENT	PK	보유 레코드 ID
user_id	BIGINT UNSIGNED	NO		FK → users.user_id ON DELETE CASCADE	사용자
currency_code	CHAR(3)	NO		FK → currencies.currency_code	통화
balance	DECIMAL(24,8)	NO	0		통화 잔액
updated_at	DATETIME	NO	CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP		갱신 시각

인덱스/제약

- PK(holding\_id)
- UNIQUE(user\_id, currency\_code) — 사용자·통화 1행만 허용
- FK(user\_id) → users(user\_id) **ON DELETE CASCADE**
- FK(currency\_code) → currencies(currency\_code)

### 3) 관계(ER) 명세서 (Relationship Specification)

#### 3.1 카디널리티

- **users (1) — (N) fx\_holdings**  
1명의 사용자가 여러 통화 보유 가능. (동일 통화 중복 금지: UNIQUE(user\_id, currency\_code))
- **currencies (1) — (N) fx\_holdings**  
1개의 통화가 여러 사용자의 보유에서 참조됨.

#### 3.2 참조 무결성/삭제 규칙

- **fx\_holdings.user\_id → users.user\_id : ON DELETE CASCADE**  
사용자를 삭제하면 그 사용자의 보유 행이 함께 삭제됩니다.
- **fx\_holdings.currency\_code → currencies.currency\_code : 기본(Restrict)**  
통화 코드를 삭제하려면 먼저 해당 통화를 참조하는 보유 데이터를 정리해야 합니다.

#### 3.3 비즈니스 규칙

- 같은 사용자·같은 통화 조합은 1행만 존재(UNIQUE).
- 금액은 통화 단위 그대로 저장(환산가치 아님).
- 통화 소수 자리수는 **currencies.minor\_unit**로 표현(표시/검증에 활용 가능).

## 4) 입력 데이터

### 통화(currencies)

currency_code	currency_name	minor_unit
KRW	Korean Won	0
USD	US Dollar	2
JPY	Japanese Yen	0
EUR	Euro	2
GBP	Pound Sterling	2
AUD	Australian Dollar	2

## 사용자(users)

user_id	email	full_name	status
1	alice@example.com	Alice Kim	active
2	bob@example.com	Bob Lee	active
3	carol@example.com	Carol Park	active
4	dave@example.com	Dave Choi	active
5	erin@example.com	Erin Jung	inactive
6	frank@example.com	Frank Yoo	active
7	grace@example.com	Grace Han	active
8	heidi@example.com	Heidi Lim	active

## 보유 외환(fx\_holdings) — 총 19행

user_id	currency_code	balance
1	KRW	1,200,000.00000000
1	USD	950.25000000
1	EUR	210.00000000
2	KRW	560,000.00000000
2	JPY	75,000.00000000
3	USD	1,250.75000000
3	GBP	80.00000000

4	KRW	330,000.00000000
4	USD	420.00000000
4	JPY	12,000.00000000
5	EUR	500.00000000
5	AUD	600.00000000
6	KRW	980,000.00000000
6	USD	110.00000000
6	EUR	45.50000000

7	JPY	150,000.00000000
7	GBP	30.00000000
8	KRW	250,000.00000000
8	AUD	120.00000000

## 5) 실습문제

1. **fx\_holdings, users** 테이블에서 전체 사용자 보유 외환을 조회하세요.  
(컬럼: `user_id, full_name, currency_code, balance` / `user_id, currency_code` 오름차순)
2. **users, fx\_holdings** 테이블에서 이메일이 `alice@example.com`인 사용자의 보유 외환을 조회하세요.  
(컬럼: `currency_code, balance` / `currency_code` 오름차순)
3. **fx\_holdings** 테이블에서 통화별 총 보유 잔액을 조회하세요.  
(컬럼: `currency_code, total_balance` / `currency_code` 오름차순)
4. **users, fx\_holdings** 테이블에서 사용자별 보유 요약(보유 통화 수, 잔액 합계)을 조회하세요.  
(컬럼: `user_id, full_name, currency_cnt, sum_raw` / `user_id` 오름차순)
5. **fx\_holdings, users** 테이블에서 원화(`KRW`) 보유 상위 5명의 이름과 잔액을 조회하세요.  
(`balance` 내림차순, 상위 5명)

6. **fx\_holdings, users** 테이블에서 달러(USD)를 보유한 사용자와 해당 잔액을 조회하세요.  
(컬럼: `full_name, balance / balance` 내림차순)
7. **users, fx\_holdings** 테이블에서 보유 통화가 **2개** 이상인 사용자의 ID, 이름, 보유 통화  
개수를 조회하세요.  
(개수 내림차순, 같은 개수면 `full_name` 오름차순)
8. **users, fx\_holdings** 테이블에서 상태가 `inactive`인 사용자의 보유 내역을 조회하세요.  
(컬럼: `full_name, currency_code, balance / full_name` 오름차순)
9. **fx\_holdings, users, currencies** 테이블에서 이메일이 `alice@example.com`인  
사용자의 보유 외환과 통화 명문명을 함께 조회하세요.  
(컬럼: `currency_code, currency_name, balance / currency_code` 오름차순)
10. **fx\_holdings** 테이블에서 통화별 보유 사용자 수와 총 잔액을 함께 조회하세요.  
(컬럼: `currency_code, holder_count, total_balance / currency_code` 오름차순)