

부산 응급실 혼잡도 예측 데이터베이스 ERD 설계

테이블 구조

1. hospitals (병원 정보 - 정적 데이터)

목적: 부산 응급실 기본 정보 저장

컬럼명	데이터 타입	제약 조건	설명
id	VARCHAR(10)	PRIMARY KEY	병원 고유 ID (예: ER001)
name	VARCHAR(100)	NOT NULL	병원명
address	VARCHAR(200)	NOT NULL	도로명주소
phone	VARCHAR(20)	NOT NULL	대표 전화번호
emergency_phone	VARCHAR(20)		응급실 직통번호
latitude	DECIMAL(10, 8)	NOT NULL	위도
longitude	DECIMAL(11, 8)	NOT NULL	경도
district	VARCHAR(20)	NOT NULL	구 (예: 서구, 해운대구)
emergency_level	VARCHAR(50)	NOT NULL	응급실 등급
hospital_type	VARCHAR(50)	NOT NULL	병원 유형
beds_total	INT		총 병상 수
has_ct	BOOLEAN	DEFAULT FALSE	CT 보유 여부
has_mri	BOOLEAN	DEFAULT FALSE	MRI 보유 여부
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	생성일시
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	수정일시

인덱스:

- PRIMARY KEY: id
- INDEX: district (구별 검색용)
- INDEX: emergency_level (등급별 검색용)

2. emergency_predictions (혼잡도 예측 - 동적 데이터, 나중에 구현)

목적: 시간대별 응급실 혼잡도 예측 정보

컬럼명	데이터 타입	제약 조건	설명
id	INT	PRIMARY KEY AUTO_INCREMENT	예측 레코드 ID
hospital_id	VARCHAR(10)	FOREIGN KEY	병원 ID (hospitals.id 참조)
prediction_datetime	DATETIME	NOT NULL	예측 시점
predicted_wait_time	INT		예상 대기시간 (분)
congestion_level	ENUM('여유', '보통', '혼잡')	NOT NULL	혼잡도 상태
congestion_color	ENUM('green', 'yellow', 'red')	NOT NULL	혼잡도 색상
predicted_patients	INT		예상 대기 환자 수
confidence_score	DECIMAL(5, 2)		예측 신뢰도 (0-100)
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	예측 생성일시

인덱스:

- PRIMARY KEY: id
- FOREIGN KEY: hospital_id → hospitals(id)
- INDEX: (hospital_id, prediction_datetime) (병원별 시간 조회용)

3. historical_data (과거 데이터 - 머신러닝 학습용, 나중에 구현)

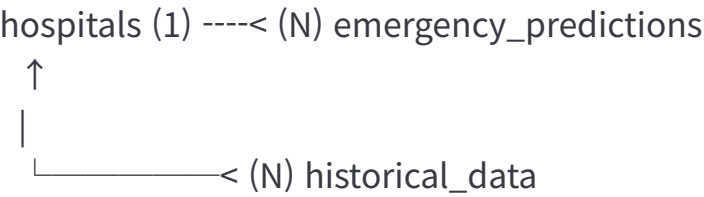
목적: 머신러닝 모델 학습을 위한 시계열 데이터

컬럼명	데이터 타입	제약 조건	설명
id	INT	PRIMARY KEY AUTO_INCREMENT	레코드 ID
hospital_id	VARCHAR(10)	FOREIGN KEY	병원 ID
record_datetime	DATETIME	NOT NULL	기록 시점
actual_wait_time	INT		실제 대기시간 (분)
patient_count	INT		대기 환자 수
available_beds	INT		가용 병상 수
day_of_week	TINYINT		요일 (0=월요일)
hour_of_day	TINYINT		시간 (0-23)
is_holiday	BOOLEAN	DEFAULT FALSE	공휴일 여부
weather_condition	VARCHAR(20)		날씨 상태
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	생성일시

인덱스:

- PRIMARY KEY: id
- FOREIGN KEY: hospital_id → hospitals(id)
- INDEX: (hospital_id, record_datetime) (병원별 시계열 조회용)
- INDEX: (day_of_week, hour_of_day) (패턴 분석용)

테이블 관계 (Relationships)




- hospitals ↔ emergency_predictions: 1:N 관계**

- 하나의 병원은 여러 시간대의 예측 데이터를 가짐
 - **hospitals ↔ historical_data**: 1:N 관계
 - 하나의 병원은 여러 과거 기록을 가짐
-

현재 단계에서 구현할 테이블

Phase 1 (현재):

-  **hospitals** 테이블만 구현
- CSV 데이터 import
- 기본 CRUD API 개발

Phase 2 (2주차):

- **emergency_predictions** 테이블 추가
- 머신러닝 시뮬레이션 데이터 생성

Phase 3 (선택사항):

- **historical_data** 테이블 추가
 - 시계열 데이터 누적
-

Draw.io에서 ERD 그리는 방법

1단계: Draw.io 시작

1. <https://app.diagrams.net/> 접속
2. "Create New Diagram" 클릭

3. "Blank Diagram" 선택

2단계: 테이블 추가

- 1. 왼쪽 패널에서 "Entity Relation" 카테고리 선택
- 2. "Table" 아이콘을 캔버스에 드래그
- 3. 테이블 이름 변경: **hospitals**

3단계: 컬럼 추가

테이블을 더블클릭하여 편집 모드:



hospitals

PK id: VARCHAR(10)

name: VARCHAR(100)

address: VARCHAR(200)

phone: VARCHAR(20)

emergency_phone: VARCHAR(20)

latitude: DECIMAL(10,8)

longitude: DECIMAL(11,8)

district: VARCHAR(20)

emergency_level: VARCHAR(50)

hospital_type: VARCHAR(50)

beds_total: INT

has_ct: BOOLEAN

has_mri: BOOLEAN

created_at: TIMESTAMP

updated_at: TIMESTAMP

4단계: 나머지 테이블 추가 (나중에)

- emergency_predictions
- historical_data

5단계: 관계선 연결 (나중에)

- 테이블 간 선을 그어서 FOREIGN KEY 관계 표시
-

다음 단계

ERD 완성 후:

- 1. MySQL 데이터베이스 생성
- 2. CREATE TABLE SQL 문 작성
- 3. CSV 데이터 import
- 4. Spring Boot Entity 클래스 생성