☑️ DATABASE 설계

# **데이터베이스 테이블 구조**

## **1. 사용자 관련 테이블**

### **Department Table (부서 정보)**

* **department\_id** (PK) - 부서 ID
* **department\_name** - 부서명

### **UserAccount Table (사용자 계정 정보 / 안바뀌는거)**

* **user\_id** (PK) - 사용자 ID
* **user\_name** - 사용자 이름
* **department\_id** (FK) - 부서 ID
* **password** - 비밀번호 (VARBINARY)

### **UserBasic Table (사용자 기본 정보 / 바뀌는거)**

* **user\_id** (PK/FK) - 사용자 ID
* **email** - 이메일 주소
* **gender** - 성별
* **age** - 연령대
* **card\_company** - 결제 카드사
* **card\_num** - 카드 번호

### **PayHistory Table (결제 히스토리 정보)**

* **payment\_id (PK)** - 결제번호
* **user\_id (FK)** - 사용자 ID
* **payment\_timestamp** - 결제일시
* **payment\_money** - 결제금액

### **UserBody Table (사용자 신체 정보)**

* **user\_id** (PK/FK) - 사용자 ID
* **weight** - 체중
* **body\_fat** - 체지방률
* **muscle** - 골격근량
* **bmi** - BMI
* **update\_timestamp** - 정보 업데이트 일시

### **UserHealth Table (사용자 건강 정보)**

* **user\_id** (PK/FK) - 사용자 ID
* **activity\_level** - 활동 수준(4단계) ENUM
* **health\_issue** - 질병 및 건강문제
* **meal\_goal** - 식단 목표
* **weight\_goal** - 체중 목표
* ***plus\_minus\_food*** *- 선호, 비선호 음식 (JSON {key:food, value:plus/minus})*
* **meal\_type** - 선호하는 식단 유형(6개)
* **allergy\_name** - 알러지명

**2. 구독 관련 테이블**

### **MealSubscription Table (식단 구독 정보)**

* **subs\_id (PK) -** 구독 ID
* **user\_id (FK)** - 사용자 ID
* **subs\_timestamp** - 구독일(향후 월, 주차, 요일 계산에 필요)
* **subs\_time** - 조식, 중식, 석식 구분 (breakfast, lunch, dinner)
* **subs\_status** - 구독 상태 (BOOLEAN)

**MealUserFeedback Tabel (식단 후기 정보)**

* **subs\_id (PK/FK)** - 구독 ID
* **meal\_score** - 식단 사용자 평점
* **meal\_feedback** - 식단 후기
* **favorite** - 즐겨찾기 여부 (BOOLEAN)

### **MealDetail Table (식단 구성 음식)**

* **subs\_id (PK/FK)** - 구독 ID
* **food\_id (PK/FK)** - 음식 ID

## **3. 푸드백과 관련 테이블**

### **Food Table (음식 데이터)**

* **food\_id (PK)** - 음식 ID
* **food\_name** - 음식 이름
* **calories** - 칼로리 정보
* **carbs** - 탄수화물
* **protein** - 단백질
* **fat** - 지방
* **sugars** - 당류
* **fiber** - 식이섬유
* **sodium** - 나트륨

### **Ingredients Table (구성 재료 데이터)**

* **food\_id (PK/FK)** - 음식 ID
* **ingredient (PK)** - 구성 재료

### **FoodKeyword Table (검색 키워드)**

* **food\_id (PK/FK) -** 음식 ID
* **search\_keyword (PK)** - 검색 키워드

### **FoodSearchHistory Table (음식 검색 기록)**

* **search\_id (PK)** - 검색 기록 ID
* **user\_id (FK)** - 사용자 ID
* **search\_name** - 검색어

## **4. 챗봇 및 AI 추천 관련 테이블**

### **ChatHistory Table (챗봇 대화 기록)**

* **chat\_id (PK)** - 대화 ID
* **chat\_topic** - 대화 주제
* **user\_id (FK)** - 사용자 ID
* **create\_timestamp** - 대화 생성 시간
* **chat\_score** - 사용자 평점
* **sentiment** - 감정 분석 (positive, neutral, negative)

### **ChatMessages Table (챗봇 메세지)**

* **chat\_message\_id (PK)** - 메세지 ID
* **chat\_id (FK)** - 대화 ID
* **sender** - 입력자 (User, AI)
* **message** - 입력 텍스트

탭 5

### **Food Table (음식 데이터)**

* **food\_id (PK)** - 음식 ID
* **food\_name** )- 음식 이름
* **category\_id (FK)** - 대분류 ID
* **subcategory\_id (FK)** - 중분류 ID
* **food\_weight** - 음식 중량(g)

### **FoodCategory Table (음식 대분류 데이터)**

* **category\_id (PK)** - 대분류 ID
* **category\_name** - 대분류명(밥, 메인반찬, 서브반찬, 서브반찬2, 국, 김치)

### **FoodSubcategory Table (음식 소분류 데이터)**

* **subcategory\_id (PK)** - 소분류 ID
* **category\_id (FK)** - 대분류 ID
* **subcategory\_name** - 소분류명 (생채·무침류, 국 및 탕류, 볶음류, 밥류, 조림류, 튀김류, 구이류, 찌개 및 전골류, 나물·숙채류, 전·적 및 부침류, 찜류, 면류, 죽 및 스프류, 빵 및 과자류, 만두류, 김치류, 장아찌·절임류, 젓갈류)

### **Nutrition Table (음식 영양성분 데이터 / 100g 기준)**

* **food\_id (PK, FK)** - 음식 ID
* **calories** - 열량(kcal)
* **carbs** - 탄수화물(g)
* **protein** - 단백질(g)
* **fat** - 지방(g)
* **sugars** - 당류(g)
* **fiber** - 식이섬유(g)
* **sodium** - 나트륨(mg)

☑️ CREATE TABLE

# **CREATE TABLE**

**-- 1. 사용자 관련 테이블**

**-- 부서 정보 테이블**

CREATE TABLE Department (

department\_id INT PRIMARY KEY AUTO\_INCREMENT,

department\_name VARCHAR(255) NOT NULL

);

**-- 사용자 계정 정보 테이블**

CREATE TABLE UserAccount (

user\_id INT PRIMARY KEY AUTO\_INCREMENT,

user\_name VARCHAR(255) NOT NULL,

department\_id INT,

password VARBINARY(255) NOT NULL,

FOREIGN KEY (department\_id) REFERENCES Department(department\_id)

);

**-- 사용자 기본 정보 테이블**

CREATE TABLE UserBasic (

user\_id INT PRIMARY KEY,

email VARCHAR(255) UNIQUE NOT NULL,

gender ENUM('M', 'F', 'Other') NOT NULL,

age INT NOT NULL,

card\_company VARCHAR(255),

card\_num VARCHAR(255),

FOREIGN KEY (user\_id) REFERENCES UserAccount(user\_id)

);

**-- 결제 히스토리 테이블**

CREATE TABLE PayHistory (

payment\_id INT PRIMARY KEY AUTO\_INCREMENT,

user\_id INT,

payment\_timestamp DATETIME NOT NULL,

payment\_money DECIMAL(10,2) NOT NULL,

FOREIGN KEY (user\_id) REFERENCES UserAccount(user\_id)

);

**-- 사용자 신체 정보 테이블**

CREATE TABLE UserBody (

user\_id INT PRIMARY KEY,

weight DECIMAL(5,2),

body\_fat DECIMAL(5,2),

muscle DECIMAL(5,2),

bmi DECIMAL(5,2),

update\_timestamp DATETIME NOT NULL,

FOREIGN KEY (user\_id) REFERENCES UserAccount(user\_id)

);

**-- 사용자 건강 정보 테이블**

CREATE TABLE UserHealth (

user\_id INT PRIMARY KEY,

activity\_level ENUM('1', '2', '3', '4') NOT NULL,

health\_issue TEXT,

meal\_goal TEXT,

weight\_goal DECIMAL(5,2),

plus\_minus\_food JSON,

meal\_type ENUM('A', 'B', 'C', 'D', 'F', 'E'),

allergy\_name TEXT,

FOREIGN KEY (user\_id) REFERENCES UserAccount(user\_id)

);

**-- 2. 구독 관련 테이블**

**-- 식단 구독 정보 테이블**

CREATE TABLE MealSubscription (

subs\_id INT PRIMARY KEY AUTO\_INCREMENT,

user\_id INT,

subs\_timestamp DATETIME NOT NULL,

subs\_time ENUM('breakfast', 'lunch', 'dinner') NOT NULL,

subs\_status BOOLEAN NOT NULL,

FOREIGN KEY (user\_id) REFERENCES UserAccount(user\_id)

);

**-- 식단 후기 테이블**

CREATE TABLE MealUserFeedback (

subs\_id INT PRIMARY KEY,

meal\_score INT CHECK (meal\_score BETWEEN 1 AND 5),

meal\_feedback TEXT,

favorite BOOLEAN,

FOREIGN KEY (subs\_id) REFERENCES MealSubscription(subs\_id)

);

**-- 식단 구성 음식 테이블**

CREATE TABLE MealDetail (

subs\_id INT,

food\_id INT,

PRIMARY KEY (subs\_id, food\_id),

FOREIGN KEY (subs\_id) REFERENCES MealSubscription(subs\_id),

FOREIGN KEY (food\_id) REFERENCES Food(food\_id)

);

**-- 3. 푸드백과 관련 테이블**

**-- 음식 대분류 테이블**

CREATE TABLE FoodCategory (

category\_id INT PRIMARY KEY AUTO\_INCREMENT,

category\_name VARCHAR(255) NOT NULL

);

**-- 음식 소분류 테이블**

CREATE TABLE FoodSubcategory (

subcategory\_id INT PRIMARY KEY AUTO\_INCREMENT,

category\_id INT,

subcategory\_name VARCHAR(255) NOT NULL,

FOREIGN KEY (category\_id) REFERENCES FoodCategory(category\_id)

);

**-- 음식 테이블**

CREATE TABLE Food (

food\_id INT PRIMARY KEY AUTO\_INCREMENT,

food\_name VARCHAR(255) NOT NULL,

category\_id INT,

subcategory\_id INT,

food\_weight DECIMAL(6,2),

FOREIGN KEY (category\_id) REFERENCES FoodCategory(category\_id),

FOREIGN KEY (subcategory\_id) REFERENCES FoodSubcategory(subcategory\_id)

);

**-- 음식 영양성분 테이블 (100g 기준)**

CREATE TABLE Nutrition (

food\_id INT PRIMARY KEY,

calories DECIMAL(6,2),

carbs DECIMAL(6,2),

protein DECIMAL(6,2),

fat DECIMAL(6,2),

sugars DECIMAL(6,2),

fiber DECIMAL(6,2),

sodium DECIMAL(6,2),

FOREIGN KEY (food\_id) REFERENCES Food(food\_id)

);

**-- 음식 구성 재료 테이블**

CREATE TABLE Ingredients (

food\_id INT,

ingredient VARCHAR(255),

PRIMARY KEY (food\_id, ingredient),

FOREIGN KEY (food\_id) REFERENCES Food(food\_id)

);

**-- 음식 검색 키워드 테이블**

CREATE TABLE FoodKeyword (

food\_id INT,

search\_keyword VARCHAR(255),

PRIMARY KEY (food\_id, search\_keyword),

FOREIGN KEY (food\_id) REFERENCES Food(food\_id)

);

**-- 음식 검색 기록 테이블**

CREATE TABLE FoodSearchHistory (

search\_id INT PRIMARY KEY AUTO\_INCREMENT,

user\_id INT,

search\_name VARCHAR(255) NOT NULL,

FOREIGN KEY (user\_id) REFERENCES UserAccount(user\_id)

);

**-- 4. 챗봇 및 AI 추천 관련 테이블**

**-- 챗봇 대화 기록 테이블**

CREATE TABLE ChatHistory (

chat\_id INT PRIMARY KEY AUTO\_INCREMENT,

chat\_topic VARCHAR(255) NOT NULL,

user\_id INT,

create\_timestamp DATETIME NOT NULL,

chat\_score INT CHECK (chat\_score BETWEEN 1 AND 5),

sentiment ENUM('positive', 'neutral', 'negative') NOT NULL,

FOREIGN KEY (user\_id) REFERENCES UserAccount(user\_id)

);

**-- 챗봇 메시지 테이블**

CREATE TABLE ChatMessages (

chat\_message\_id INT PRIMARY KEY AUTO\_INCREMENT,

chat\_id INT,

sender ENUM('User', 'AI') NOT NULL,

message TEXT NOT NULL,

FOREIGN KEY (chat\_id) REFERENCES ChatHistory(chat\_id)

);

☑️ TABLE 정의서

# **TABLE 정의서**

| 구분 | 테이블명 | 설명 |
| --- | --- | --- |
| 1. 사용자 관련 테이블 | Department Table | 부서 정보 |
| **UserAccount Table** | **사용자 계정 정보** |
| UserBasic Table | 사용자 기본 정보 |
| PayHistory Table | 결제 히스토리 정보 |
| UserBody Table | 사용자 신체 정보 |
| UserHealth Table | 사용자 건강 정보 |
| 2. 식단 구독 관련 테이블 | MealSubscription Table | 식단 구독 정보 |
| MealUserFeedback Tabel | 식단 후기 정보 |
| MealDetail Table | 식단 구성 음식 |
| 3. 푸드백과 관련 테이블 | Food Table | 음식 데이터 |
| Ingredients Table | 구성 재료 데이터 |
| FoodKeyword Table | 검색 키워드 |
| FoodSearchHistory Table | 음식 검색 기록 |
| 4. 챗봇 관련 테이블 | ChatHistory Table | 챗봇 대화 기록 |
| ChatMessages Table | 챗봇 메세지 |

## 

## **1. 사용자 관련 테이블**

TABLE NAME : **Department Table (부서 정보)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| department\_id | INT | PRIMARY KEY AUTO\_INCREMENT | 부서 ID | 1  2 |
| department\_name | VARCHAR(255) | NOT NULL | 부서명 | 영업팀  개발팀 |

TABLE NAME : **UserAccount Table (사용자 계정 정보)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| user\_id | INT | PRIMARY KEY  AUTO\_INCREMENT | 사용자 ID | 1  2 |
| user\_name | VARCHAR(255) | NOT NULL | 사용자 이름 | 홍길동  김철수 |
| department\_id | INT | FOREIGN KEY(Department(department\_id)) | 부서 ID | 1  2 |
| password | VARBINARY(255) | NOT NULL | 비밀번호 | \*\*\*\*\*\*\*  \*\*\*\*\*\*\* |

TABLE NAME : **UserBasic Table (사용자 기본 정보)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| user\_id | INT | PRIMARY KEY  FOREIGN KEY(UserAccount(user\_id)) | 사용자 ID | 1  2 |
| email | VARCHAR(255) | UNIQUE, NOT NULL | 이메일 주소 | yun23@example.com  dodo36@example.com |
| gender | ENUM(‘M’, ‘F’, ‘Other’) | NOT NULL | 성별 | M  F |
| age | INT | NOT NULL | 연령대 | 30  25 |
| card\_company | VARCHAR(255) |  | 결제 카드사 | Visa  MasterCard |
| card\_num | VARCHAR(255) |  | 카드 번호 | 1223-3224-1233-4443  3444-3445-3432-2344 |

TABLE NAME : **PayHistory Table (결제 히스토리 정보)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| payment\_id | INT | PRIMARY KEY  AUTO\_INCREMENT | 결제 번호 | 1  2 |
| user\_id | INT | FOREIGN KEY(UserAccount(user\_id)) | 사용자 ID | 1  2 |
| payment\_timestamp | DATETIME | NOT NULL | 결제 일시 | 2025-03-01 12:30:00  2025-03-02 18:45:00 |
| payment\_money | DECIMAL(10, 2) | NOT NULL | 결제 금액 | 25000  35000 |

TABLE NAME : **UserBody Table (사용자 신체 정보)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| user\_id | INT | PRIMARY KEY  FOREIGN KEY(UserAccount(user\_id)) | 사용자 ID | 1  2 |
| weight | DECIMAL(5, 2) |  | 체중 | 70.5  80.2 |
| body\_fat | DECIMAL(5, 2) |  | 체지방률 | 20.3  18.5 |
| muscle | DECIMAL(5, 2) |  | 골격근량 | 32.5  35.0 |
| bmi | DECIMAL(5, 2) |  | BMI | 22.5  24.0 |
| update\_timestamp | DATETIME | NOT NULL | 정보 업데이트 일시 | 2025-03-01 10:00:00  2025-03-05 12:32:00 |

TABLE NAME : **UserHealth Table (사용자 건강 정보)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| user\_id | INT | PRIMARY KEY  FOREIGN KEY(UserAccount(user\_id)) | 사용자 ID | 1  2 |
| activity\_level | INT | NOT NULL | 활동수준 (4단계) | 3  2 |
| health\_issue | TEXT |  | 질병 및 건강문제 | 고혈압  없음 |
| meal\_goal | TEXT |  | 식단 목표 | 근육 증가  체중 감량 |
| weight\_goal | DECIMAL(5, 2) |  | 체중 목표 | 75.0  70.0 |
| plus\_minus\_food | JSON |  | 선호, 비선호 음식 | {‘chicken’:’plus’, ‘pinuts’:’minus’}  {‘beef’:’plus’, ‘diary’:’minus’} |
| meal\_type | ENUM(‘A’, ‘B’, ‘C’, ‘D’, ‘E’, ‘F’) |  | 선호하는 식단 유형 | A  C |
| allergy\_name | TEXT |  | 알러지명 | 복숭아  콩 |

**2. 식단 구독 관련 테이블**

TABLE NAME : **MealSubscription Table (식단 구독 정보)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| subs\_id | INT | PRIMARY KEY  AUTO\_INCREMENT | 구독 ID | 1  2 |
| user\_id | INT | FOREIGN KEY(UserAccount(user\_id)) | 사용자 ID | 1  2 |
| subs\_timestamp | DATETIME | NOT NULL | 구독 일시 | 2025-03-01 10:00:00  2025-03-05 12:32:00 |
| subs\_time | ENUM(‘breakfast’, ‘lunch’, ‘dinner’) | NOT NULL | 조식, 중식, 석식 구분 | breakfast  lunch |
| subs\_status | BOOLEAN | NOT NULL | 구독 상태 | 1  1 |

TABLE NAME : **MealUserFeedback Tabel (식단 후기 정보)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| subs\_id | INT | PRIMARY KEY  FOREIGN KEY(MealSubscription(subs\_id)) | 구독 ID | 1  2 |
| meal\_score | INT | CHECK(meal\_score BETWEEN 1 AND 5) | 식단 사용자 평점 | 5  3 |
| meal\_feedback | TEXT |  | 식단 후기 | 아주 맛있었어요!  조금 짰어요. |
| favorite | BOOLEAN |  | 즐겨찾기 여부 | 1  0 |

TABLE NAME : **MealDetail Table (식단 구성 음식)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| subs\_id | INT | PRIMARY KEY  FOREIGN KEY(MealSubscription(subs\_id)) | 구독 ID | 1  2 |
| food\_id | INT | PRIMARY KEY  FOREIGN KEY(Food(food\_id)) | 음식 ID | 1  2 |

## 

## 

## **3. 푸드백과 관련 테이블**

TABLE NAME : **Food Table (음식 데이터)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| food\_id | INT | PRIMARY KEY  AUTO\_INCREMENT | 음식 ID | 1  2 |
| food\_name | VARCHAR(255) | NOT NULL | 음식 이름 | 닭가슴살 샐러드  치킨 버거 |
| category\_id | INT | FOREIGN KEY(FoodCategory(category\_id)) | 대분류 ID | 1  2 |
| subcategory\_id | INT | FOREIGN KEY(FoodSubcategory(subcategory\_id)) | 소분류 ID | 1  2 |
| food\_weight | DECIMAL(6, 2) |  | 음식 중량 (g) | 490  350 |

TABLE NAME : **FoodCatecory Table (음식 대분류 데이터)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| category\_id | INT | PRIMARY KEY  AUTO\_INCREMENT | 대분류 ID | 1  2 |
| category\_name | VARCHAR(255) | NOT NULL | 대분류명 | 메인 반찬  서브 반찬 |

TABLE NAME : **FoodSubcategory Table (음식 소분류 데이터)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| subcategory\_id | INT | PRIMARY KEY  AUTO\_INCREMENT | 소분류 ID | 1  2 |
| category\_id | INT | FOREIGN KEY(FoodCategory(category\_id)) | 대분류 ID | 1  2 |
| subcategory\_name | VARCHAR(255) | NOT NULL | 소분류 이름 | 볶음류  조림류 |

TABLE NAME : **Nutrition Table (음식 영양성분 데이터 / 100g 기준)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| food\_id | INT | PRIMARY KEY, FOREIGN KEY(Food(food\_id)) | 음식 ID | 1  2 |
| calories | DECIMAL(6, 2) |  | 열량(kcal) | 250  350 |
| carbs | DECIMAL(6, 2) |  | 탄수화물(g) | 10  40 |
| protein | DECIMAL(6, 2) |  | 단백질(g) | 30  20 |
| fat | DECIMAL(6, 2) |  | 지방(g) | 5  10 |
| sugars | DECIMAL(6, 2) |  | 당류(g) | 3.7  7.15 |
| fiber | DECIMAL(6, 2) |  | 식이섬유(g) | 3.2  6.5 |
| sodium | DECIMAL(6, 2) |  | 나트륨(mg) | 180  570 |

TABLE NAME : **Ingredients Table (구성 재료 데이터)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| food\_id | INT | PRIMARY KEY  FOREIGN KEY(Food(food\_id)) | 음식 ID | 1  2 |
| ingredient | VARCHAR(255) | PRIMARY KEY | 구성 재료 | 닭가슴살  양상추 |

TABLE NAME : **FoodKeyword Table (검색 키워드)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| food\_id | INT | PRIMARY KEY  FOREIGN KEY(Food(food\_id)) | 음식 ID | 1  2 |
| search\_keyword | VARCHAR(255) | PRIMARY KEY | 검색 키워드 | 닭  계란 |

TABLE NAME : **FoodSearchHistory Table (음식 검색 기록)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| search\_id | INT | PRIMARY KEY  AUTO\_INCREMENT | 검색 기록 ID | 1  2 |
| user\_id | INT | FOREIGN KEY(UserAccount(user\_id)) | 사용자 ID | 1  2 |
| search\_name | VARCHAR(255) | NOT NULL | 검색어 | 찜닭  고사리 |

## 

## **4. 챗봇 및 AI 추천 관련 테이블**

TABLE NAME : **ChatHistory Table (챗봇 대화 기록)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| chat\_id | INT | PRIMARY KEY  AUTO\_INCREMENT | 대화 ID | 1  2 |
| chat\_topic | VARCHAR(255) | NOT NULL | 대화 주제 | 식단 추천  건강 상담 |
| user\_id | INT | FOREIGN KEY(UserAccount(user\_id)) | 사용자 ID | 1  2 |
| create\_timestamp | DATETIME | NOT NULL | 대화 생성 시간 | 2025-03-01 10:00:00  2025-03-05 12:32:00 |
| chat\_score | INT | CHECK (chat\_score BETWEEN 1 AND 5) | 사용자 평점 | 5  4 |
| sentiment | ENUM(‘positive’, ‘neutral’, ‘negative’) | NOT NULL | 감정 분석 | positive  meutral |

TABLE NAME : **ChatMessages Table (챗봇 메세지)**

| column | data type | 제약조건 | 설명 | 예시 |
| --- | --- | --- | --- | --- |
| chat\_message\_id | INT | PRIMARY KEY  AUTO\_INCREMENT | 메세지 ID | 1  2 |
| chat\_id | INT | FOREIGN KEY(ChatHistory(chat\_id)) | 대화 ID | 1  2 |
| sender | ENUM  (‘User’, ‘AI’) | NOT NULL | 입력자  (User, AI) | User  AI |
| message | TEXT | NOT NULL | 입력 텍스트 | - 나 땅콩 알러지 있는데 괜찮을까?  - 다이어트를 목표로 한 식단을 추천 드릴게요! |

☑️ ERD

