# 상품 테이블: 상품에 대한 정보를 저장

CREATE TABLE product (

id int NOT NULL AUTO\_INCREMENT, # 상품 ID, 고유 식별자

name varchar(255) NOT NULL, # 상품 이름

img\_path varchar(255) DEFAULT NULL, # 상품 이미지 경로

category varchar(255) DEFAULT NULL, # 상품 카테고리

price int NOT NULL, # 상품 가격

last\_update\_time datetime(6) NOT NULL, # 마지막 업데이트 시간

PRIMARY KEY (id)

) ENGINE=InnoDB AUTO\_INCREMENT=1 DEFAULT CHARSET=utf8;

# 고객 테이블: 고객에 대한 정보를 저장

CREATE TABLE customer (

id bigint NOT NULL AUTO\_INCREMENT, # 고객 ID, 고유 식별자

password varchar(128) NOT NULL, # 고객 비밀번호

last\_login datetime(6) DEFAULT NULL, # 마지막 로그인 시간

is\_superuser tinyint(1) NOT NULL, # 슈퍼유저 여부

username varchar(150) NOT NULL, # 사용자 이름

first\_name varchar(150) NOT NULL, # 이름

last\_name varchar(150) NOT NULL, # 성

email varchar(254) NOT NULL, # 이메일 주소

is\_staff tinyint(1) NOT NULL, # 스태프 여부

is\_active tinyint(1) NOT NULL, # 활성 상태

date\_joined datetime(6) NOT NULL, # 가입 날짜

phone\_number varchar(20) DEFAULT NULL, # 전화번호

age int DEFAULT NULL, # 나이

gender varchar(10) DEFAULT NULL, # 성별

address varchar(200) DEFAULT NULL, # 주소

last\_update\_time datetime(6) NOT NULL, # 마지막 업데이트 시간

name varchar(255) DEFAULT NULL, # 전체 이름

PRIMARY KEY (id),

UNIQUE KEY username (username)

) ENGINE=InnoDB AUTO\_INCREMENT=1 DEFAULT CHARSET=utf

# 주문 테이블: 고객의 주문에 대한 정보를 저장

CREATE TABLE orders (

id int NOT NULL AUTO\_INCREMENT, # 주문 ID, 고유 식별자

promo\_id varchar(255) DEFAULT NULL, # 프로모션 ID

order\_cnt int NOT NULL, # 주문 수량

order\_price int NOT NULL, # 주문 가격

order\_dt varchar(255) NOT NULL, # 주문 날짜

last\_update\_time datetime(6) NOT NULL, # 마지막 업데이트 시간

cust\_id bigint NOT NULL, # 고객 ID, 고객 테이블의 외래키

prd\_id int NOT NULL, # 상품 ID, 상품 테이블의 외래키

PRIMARY KEY (id),

KEY order\_prd\_id\_f3688dba\_fk\_product\_prd\_id (prd\_id),

KEY order\_cust\_id\_a1158f81\_fk\_customer\_id (cust\_id),

KEY ix\_orders\_01 (promo\_id),

KEY ix\_orders\_02 (last\_update\_time),

CONSTRAINT order\_cust\_id\_a1158f81\_fk\_customer\_id FOREIGN KEY (cust\_id) REFERENCES customer (id),

CONSTRAINT order\_prd\_id\_f3688dba\_fk\_product\_prd\_id FOREIGN KEY (prd\_id) REFERENCES product (id)

) ENGINE=InnoDB AUTO\_INCREMENT=1 DEFAULT CHARSET=utf8;

# 참고할 SQL

1. 고객과 그들의 주문 정보 조회

SELECT customer.\*, orders.order\_price, orders.order\_dt

FROM customer

JOIN orders ON customer.id = orders.cust\_id;

2. 특정 카테고리 상품을 주문한 고객 목록

SELECT DISTINCT customer.id, customer.name

FROM customer

JOIN orders ON customer.id = orders.cust\_id

JOIN product ON orders.prd\_id = product.id

WHERE product.category = 'electronics';

3. 주문된 모든 상품의 정보와 주문한 고객의 정보

SELECT product.name, product.price, customer.name AS customer\_name, orders.order\_dt

FROM orders

JOIN product ON orders.prd\_id = product.id

JOIN customer ON orders.cust\_id = customer.id;

4. 가장 많이 팔린 상품의 이름과 판매 수량

SELECT product.name, SUM(orders.order\_cnt) AS total\_sold

FROM orders

JOIN product ON orders.prd\_id = product.id

GROUP BY product.name

ORDER BY total\_sold DESC

LIMIT 1;

5. 각 고객별 최근 주문 상품 정보

SELECT customer.id, customer.name, product.name AS product\_name, MAX(orders.last\_update\_time) AS last\_order\_time

FROM customer

JOIN orders ON customer.id = orders.cust\_id

JOIN product ON orders.prd\_id = product.id

GROUP BY customer.id, product.name;

6. 특정 고객의 모든 주문과 그 주문에 포함된 상품 정보

SELECT customer.name, orders.id AS order\_id, product.name AS product\_name, orders.order\_price

FROM customer

JOIN orders ON customer.id = orders.cust\_id

JOIN product ON orders.prd\_id = product.id

WHERE customer.id = 101;

7. 각 상품별 총 판매액

SELECT product.name, SUM(orders.order\_price \* orders.order\_cnt) AS total\_revenue

FROM product

JOIN orders ON product.id = orders.prd\_id

GROUP BY product.name;

8. 특정 기간 동안 활성 상태인 고객에 의해 주문된 모든 상품

SELECT product.name, orders.order\_dt

FROM orders

JOIN product ON orders.prd\_id = product.id

JOIN customer ON orders.cust\_id = customer.id

WHERE customer.is\_active = 1 AND orders.order\_dt BETWEEN '2022-01-01' AND '2022-12-31';

9. 각 고객별로 가장 많이 주문한 상품

SELECT customer.name, product.name AS favorite\_product, MAX(total\_orders) AS max\_orders

FROM (

SELECT orders.cust\_id, orders.prd\_id, COUNT(\*) AS total\_orders

FROM orders

GROUP BY orders.cust\_id, orders.prd\_id

) AS order\_counts

JOIN customer ON order\_counts.cust\_id = customer.id

JOIN product ON order\_counts.prd\_id = product.id

GROUP BY customer.name

ORDER BY max\_orders DESC;

10. 고객별 최근 1년간의 총 주문 금액

SELECT customer.name, SUM(orders.order\_price) AS total\_spent\_last\_year

FROM orders

JOIN customer ON orders.cust\_id = customer.id

WHERE orders.order\_dt >= DATE\_SUB(NOW(), INTERVAL 1 YEAR)

GROUP BY customer.id;

11. 오늘 주문 금액

select sum(order\_price) as today\_total\_order\_price

from orders

where order\_dt = date\_format(curdate(), '%Y-%m-%d')

order by last\_update\_time desc;

12. 어제 주문 금액

select sum(order\_price) as yesterday\_total\_order\_price

from orders

where order\_dt = date\_format(date\_sub(curdate(), interval 1 day), '%Y-%m-%d');