테이블 정보

```
-- 고객의 접근로그(행동로그) 테이블
CREATE TABLE "AwsDataCatalog"."ecommerce"."accesslog"(
  ts timestamp COMMENT '고객이 방문한 시간',
  product_id int COMMENT '상품정보 테이블의 product_id',
  customer_id int COMMENT '고객정보 테이블의 customer_id',
  order_id int COMMENT '주문정보 테이블의 order_id'
);

-- 상품정보 테이블
CREATE TABLE "rds"."ecommerce"."product" (
  product_id int NOT NULL AUTO_INCREMENT COMMENT '상품정보 테이블의 PK컬럼',
  name varchar(255) NOT NULL COMMENT '상품 이름',
  price int NOT NULL COMMENT '상품 가격',
  PRIMARY KEY (product_id)
);

-- 고객정보 테이블
CREATE TABLE "rds"."ecommerce"."customer" (
  customer_id bigint NOT NULL AUTO_INCREMENT COMMENT '고객정보 테이블의 PK컬럼',
  name varchar(255) DEFAULT NULL COMMENT '고객의 이름',
  PRIMARY KEY (customer_id),
  UNIQUE KEY username (username)
);

-- 주문정보 테이블
CREATE TABLE "rds"."ecommerce"."orders" (
  order_id int NOT NULL AUTO_INCREMENT COMMENT '주문정보 테이블의 PK컬럼',
  order_cnt int NOT NULL COMMENT '주문 수량',
  order_price int NOT NULL COMMENT '주문 가격',
  order_dt timestamp NOT NULL COMMENT '주문일자, timestamp 타입의 컬럼, where 절 작성예씨 order_dt >= CAST('2023-04-01' AS timestamp)',
  customer_id bigint NOT NULL COMMENT '고객정보 테이블의 customer_id',
  product_id int NOT NULL COMMENT '상품 테이블의 product_id',
  PRIMARY KEY (order_id),
  KEY order_prd_id_f3688dba_fk_product_prd_id (product_id),
  KEY order_cust_id_a1158f81_fk_customer_id (customer_id),
  KEY ix_orders_01 (promo_id),
  KEY ix_orders_02 (order_dt),
  CONSTRAINT order_cust_id_a1158f81_fk_customer_id FOREIGN KEY (customer_id) REFERENCES customer (customer_id),
  CONSTRAINT order_prd_id_f3688dba_fk_product_prd_id FOREIGN KEY (product_id) REFERENCES product (product_id)
);
```

SQL 작성 시 참고

### 현재 시간, 현재 timestamp

```
date_add('hour', 9, current_timestamp)
```

### 현재 시간에서 5분 전, 최근 5분간

```
date_add('minute', -5, date_add('hour', 9, current_timestamp))
```

### 현재 시간에서 7일 전, 지난 일주일간

```
date_add('day', -7, current_timestamp + interval '9' hour)
```

어제

```
date_add('day', -1, current_timestamp + interval '9' hour)
```

# SQL 예시

### 질문 : 우리회사의 최근 5분 동안 총주문금액과 총주문수량을 분 단위로 알려줘

SQL :

```
SELECT
   DATE_FORMAT(order_dt, '%Y-%m-%d %H:%i') AS order_minute,
   SUM(order_price) AS total_order_price,
   SUM(order_cnt) AS total_order_cnt
FROM "rds"."ecommerce"."orders" AS orders
WHERE order_dt >= date_add('minute', -5, date_add('hour', 9, current_timestamp))
GROUP BY DATE_FORMAT(order_dt, '%Y-%m-%d %H:%i')
ORDER BY order_minute DESC;
```

### 질문 : 우리회사의 최근 1시간 동안 주문전환율 TOP5를 알려줘

SQL :

```
WITH MinMaxTimes AS (
    SELECT
        MIN(ts) AS min_ts,
        MAX(ts) AS max_ts
    FROM
        "AwsDataCatalog"."ecommerce"."accesslog"
    WHERE ts > date_add('hour', -1, date_add('hour', 9, current_timestamp))
)
SELECT
    p.product_id,
    p.name,
    COALESCE(ra.view_count, 0) AS view_count,
    COALESCE(ro.order_count, 0) AS order_count,
    CASE
        WHEN COALESCE(ra.view_count, 0) > 0
        THEN ROUND(COALESCE(ro.order_count, 0) / CAST(COALESCE(ra.view_count, 0) AS DOUBLE) * 100, 2)
        ELSE 0
    END AS conversion_rate
FROM
    "rds"."ecommerce"."product" p
CROSS JOIN
    MinMaxTimes mmt
LEFT JOIN (
    SELECT
        product_id,
        COUNT(*) AS view_count
    FROM
        "AwsDataCatalog"."ecommerce"."accesslog",
        MinMaxTimes mmt
    WHERE ts BETWEEN mmt.min_ts AND mmt.max_ts
    GROUP BY
        product_id
) ra ON p.product_id = ra.product_id
LEFT JOIN (
    SELECT
        product_id,
        COUNT(*) AS order_count
    FROM
        "rds"."ecommerce"."orders",
        MinMaxTimes mmt
    WHERE
        order_dt BETWEEN mmt.min_ts AND mmt.max_ts
    GROUP BY
        product_id
) ro ON p.product_id = ro.product_id
ORDER BY
    conversion_rate DESC
LIMIT 5;
```

### 질문 : 우리회사의 상품 별 주문 금액 top 5를 알려줘

```
SELECT
    p.name,
    SUM(o.order_price) AS total_order_price
FROM
    "rds"."ecommerce"."orders" o
JOIN
    "rds"."ecommerce"."product" p ON o.product_id = p.product_id
GROUP BY
    p.name
ORDER BY
    total_order_price DESC
LIMIT 5;
```