

DM_user_window - 설계노트 (Version v1 scope)

1, 목적(Why)

유저 가입일(signup_date) 기준 180일 윈도우 내 **LTV outcome(매출/주문/AOV/구매여부/첫구매일)**를 1-row로 표준화한다.

downstream 분석에서 Consistency / Activation이 **장기 성과(180d revenue, orders)**를 어떻게 설명하는지의 **종속변수(Outcome DM)**로 사용한다.

2, Grain

1 user_id = 1 row

기준일은 signup_date이며, 주문/매출 지표는 signup_date 기준 0~179일(180d) 윈도우에서 계산된다.

3, Input tables & Join key

- **users** - 기준 테이블(signup_date/유저 속성), key - **user_id**
- **orders** - 180일 내 주문(구매) 이벤트, key - **user_id** (*order_id*는 주문 단위)
- **order_items** - 주문 매출(revenue = line_amount 합) 계산용, key - **order_id**
- 참고(스캔 최적화): 적용 window가 (0~180)일이므로 signup_date 기준 **bounds(min_signup, max_signup)**를 계산해 orders를 $\text{min_signup} \leq \text{date} < \text{max_signup} + 180$ 일로 먼저 필터한 뒤, 최종 집계에서 유저별 window 조건을 다시 적용한다.

4, Partition / Clustering

- **PARTITION BY** : signup_date 코호트/기간 분석이 핵심("가입 월별" slice)
- **CLUSTER BY** : user_id 후속 DM/분석에서 user join이 대부분

5, Window 정의

- 기준일: signup_date
- 전처리 (global filter, 스캔 절감 목적)

- $\text{min_signup} \leq \text{order_date} < \text{max_signup} + 180\text{일}$ 범위로 **orders**를 먼저 필터
- 유저별 window(최종 집계 기준)
 - 0–180d: $\text{signup_date} \leq \text{order_date} < \text{signup_date} + 180\text{일}$ (LTV outcome 계산 윈도우)
- 경계 규칙: 시작일 포함, 종료일 미포함

6, Main Features and 계산 로직

A. 주문/매출 outcome (180d)

- **orders_180d**: 180일 내 distinct order_id count
 - Logic: COUNT(DISTINCT order_id) within user window
- **revenue_180d**: 180일 내 매출 합
 - Logic: order_items에서 SUM(line_amount)로 order_id 단위 order_revenue 생성 후, user window 내 주문에 조인하여 합산
- **has_purchase_180d**: 180일 내 구매 여부
 - Logic: orders_180d > 0
- **first_order_date_180d**: 180일 내 첫 구매일
 - Logic: MIN(order_date)
- **aov_180d**: 평균 주문금액(AOV)
 - Logic: revenue_180d / orders_180d (SAFE_DIVIDE + NULLIF로 0 방지)

Scope note: SQL에는 promo 관련 컬럼/지표가 있으나, 이번 버전에서는 데이터 생성 품질 이슈로 **promo** 스코프 제외(문서/스토리에서 사용하지 않음). 이후 코드 정리 시 주석 처리/제거 예정.

7. Sanity checks

- PK 유일성: row 수 = distinct user_id 수
- 값 범위: orders_180d ≥ 0, revenue_180d ≥ 0, aov_180d ≥ 0 (NULL 가능)
- 논리 정합성:
 - has_purchase_180d == (orders_180d > 0)
 - orders_180d = 0이면 first_order_date_180d는 NULL이어야 함
- 산식 정합성(샘플 유저 점검):

- $aov_180d \approx revenue_180d / orders_180d$
- 매출 조인 중복 점검:
 - order_rev(order_id 단위)에서 order_id가 유일하게 1행인지 확인(중복이면 revenue 과대계상 위험)

8. DM이 꼭 필요한지

필요 - 프로젝트의 핵심 종속변수(**180d LTV outcome**)를 유저 단위로 표준화해, **Consistency/Activation**과의 관계 분석에 재사용 가능하다.