코드 정리

진행 상태 완료

잔존율/이탈률

```
with data_1st as (
    SELECT event_name, count(*) as event_count
  FROM `second-hand-test.analytics_440602171.events_*`
  where event_date between '20240518' and '20240520'
  group by event_name
  order by event_name
)

SELECT
    event_name,
    event_count,
    ((LAG(event_count,1) OVER (ORDER BY event_count DESC, event_
    event_count/FIRST_VALUE(event_count) OVER (ORDER BY event_c
from data_1st
where event_name LIKE '%Enter%' OR event_name LIKE 'page_view'
order by event_count desc, event_name asc
```

퍼널 전환율/이탈률

`a_impression: 노출 (page_view)`

`b_lead: 유입 (시작버튼 누른 사용자)`

`c_engagement: 응답 (Q8까지 도달한 사용자)`

`d_reach: 공유 (결과페이지까지 본 사용자)`

```
WITH event_counts AS (
    SELECT
        count(distinct case when event_name = "page_view" then u
        count(distinct case when event_name = "Enter_Q1" then us
        count(distinct case when event_name LIKE "q7_a%" then us
        count(distinct case when event_name = "Enter_ResultPage'
    FROM `second-hand-test.analytics_440602171.events_*`
    WHERE event_date BETWEEN '20240518' AND '20240520'
), funnel_data as (
  SELECT 'a_impression' as event_name, impression as event_count
  UNION ALL
  SELECT 'b_lead', lead_ FROM event_counts
  UNION ALL
  SELECT 'c_engagement', engagement FROM event_counts
  UNION ALL
  SELECT 'd_reach', reach FROM event_counts
)

SELECT
  event_name,
  event_count,
  ROUND((event_count / LAG(event_count) OVER (ORDER BY event_nam
  100 - ROUND((event_count / LAG(event_count) OVER (ORDER BY eve
FROM funnel_data
ORDER BY event_name
```

퍼널 분석 차트

```
json_data = [
    {"event_name": "a_impression", "event_count": 55},
    {"event_name": "b_lead", "event_count": 46},
    {"event_name": "c_engagement", "event_count": 32},
    {"event_name": "d_reach", "event_count": 27}
]
```

```
df = pd.DataFrame(json_data)

fig = px.funnel(df, y='event_name', x='event_count')
fig.show()
```

ABtest t-검정 (초반 작업 - 이후 수정함)

```
import numpy as np
from scipy.stats import ttest_ind

# 데이터
n_A, x_A = 63, 43
n_B, x_B = 48, 36

# 전환율
p_A = x_A / n_A
p_B = x_B / n_B

# 샘플 데이터를 생성합니다.
# 전환된 사용자는 1, 전환되지 않은 사용자는 0으로 표시
data_A = np.concatenate([np.ones(x_A), np.zeros(n_A - x_A)])
data_B = np.concatenate([np.ones(x_B), np.zeros(n_B - x_B)])

# t-검정 수행
t_stat, p_value = ttest_ind(data_A, data_B)

print(f"t_stat: {round(t_stat, 3)}")
print(f"p-value: {round(p_value, 3)}")

# 결과 해석
alpha = 0.05
if p_value < alpha:
    print("귀무가설 기각. 시작 버튼의 위치 변경이 전환율에 유의미한 영향을
```

```
else:
    print("귀무가설 채택. 시작 버튼의 위치 변경이 전환율에 유의미한 영향을
```