

Chapter 07. 프로젝트 분석 심화

프로젝트 분석 심화

🎯 19. 2020년 7월에 우리 신규유저가 하루 안에 결제로 넘어가는 비율이 어떻게 되나요?
그 비율이 어떤지 알고싶고, 결제까지 보통 몇 분 정도가 소요되는지 알고싶어요.

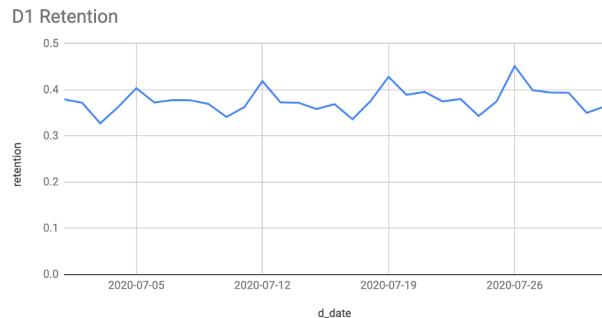
- Paying Conversion within 1 day 이해
- 최초 구매일 찾기
- LEFT JOIN 사용 이유
- TIMEDIFF 사용

```
1 • with rt_tbl as (  
2   select A.*  
3         , B.customer_id as paying_user  
4         , B.purchased_at  
5         , TIME_TO_SEC(TIMEDIFF(B.purchased_at, A.created_at))/3600 as diff_hours  
6   from fastcampus.tbl_customer A  
7   left join (select customer_id  
8              , min(purchased_at) as purchased_at  
9              from fastcampus.tbl_purchase  
10             group by customer_id) B  
11  on A.customer_id = B.customer_id  
12  and B.purchased_at < A.created_at + interval 1 day  
13  
14  where A.created_at >= '2020-07-01'  
15        and A.created_at < '2020-08-01'  
16 )  
17  
18  select round(count(paying_user)/count(customer_id)*100,2)  
19  from rt_tbl  
20  union all  
21  select round(avg(diff_hours),2)  
22  from rt_tbl
```

🎯 20. 우리 서비스는 유저의 재방문율이 높은 서비스인가요? 이를 파악하기 위해 7월 기준 Day1 Retention이 어떤지 구해주시고, 추세를 보기 위해 Daily로 추출해주세요.

- N-day Retention 이해
- SELF JOIN 사용

```
1 • select date_format(A.visited_at, '%Y-%m-%d') as d_date
2       , count(distinct A.customer_id) as active_users
3       , count(distinct B.customer_id) as retained_users
4       , count(distinct B.customer_id)/count(distinct A.customer_id) as retention
5
6 from tbl_visit A
7 left join tbl_visit B
8 on A.customer_id = B.customer_id
9 and date_format(A.visited_at, '%Y-%m-%d') = date_format(B.visited_at - interval 1 day, '%Y-%m-%d')
10
11 where date_format(A.visited_at, '%Y-%m-%d') < '2020-08-01'
12 group by 1
```



🎯 21. 우리 서비스는 신규유저가 많나요? 기존유저가 많나요? 유저들이 가입한 기간별로 그룹지어 고객 분포가 어떤지 알려주세요 DAU 기준으로 부탁드립니다.

- User Age에 따른 DAU 분석
- 마지막 접속일 찾기
- DATEDIFF 사용

```
1 • with tbl_visit_by_joined as
2   (select date_format(A.visited_at - interval 9 hour, '%Y-%m-%d') as d_date
3      , A.customer_id
4      , B.created_at as d_joined
5      , max(A.visited_at) as last_visit
6      , datediff(cast(max(A.visited_at) as date), cast(B.created_at as date)) as date_diff
7   from fastcampus.tbl_visit A
8   left join fastcampus.tbl_customer B
9   on A.customer_id = B.customer_id
10
11  where A.visited_at >= '2020-07-01'
12  and A.visited_at < '2020-08-01'
13  group by 1,2,3
14  )
```

🎯 21. 우리 서비스는 신규유저가 많나요? 기존유저가 많나요? 유저들이 가입한 기간별로 그룹지어 고객 분포가 어떤지 알려주세요 DAU 기준으로 부탁드립니다.

- User Age에 따른 DAU 분석
- 마지막 접속일 찾기
- DATEDIFF 사용

```
16  select A.d_date
17      , case when A.date_diff >= 730 then '2년 이상'
18            when A.date_diff >= 365 then '1년 이상'
19            when A.date_diff >= 183 then '6개월 이상'
20            when A.date_diff >= 91 then '3개월 이상'
21            when A.date_diff >= 30 then '1개월 이상'
22            else '1개월 미만'
23      end as segment
24      , B.all_users
25      , count(A.customer_id) as users
26      , round(count(A.customer_id)/B.all_users,2) as per
27  from tbl_visit_by_joined A
28  left join (select d_date
29              , count(customer_id) as all_users
30              from tbl_visit_by_joined
31             group by 1) B
32  on A.d_date = B.d_date
33  group by 1,2,3
34  order by 1,2,3
```