

Credit Card Payments & Rewards

Problem #1:

1. Create a base fact table, following columns are mandatory, include more if you can think of any. Report all in IST for the ease of use for others

- a. User_id
- b. First_name
- c. Last_name
- d. Gender
- e. Email
- f. Signup date
- g. First payment date
- h. First payment status
- i. Last payment date
- j. Last payment status
- k. Total amount paid
- l. First payment amount
- m. Last payment amount

Solution #1: SQL Query for the base fact table:

```
with amountinfo as (
select
    user_id,
    max(case when status = 'completed' then amount else 0 end) as Max_paid_amount,
    min(case when status != 'completed' then 0 else amount end) as Min_paid_amount,
    sum(case when status = 'completed' then amount else 0 end) as Paid_amount_sum
from public.wallet_service_orders
group by user_id
)
select
    distinct ud.id as User_Id,
    ud.first_name as First_Name,
    ud.last_name as Last_Name,
    ud.email as Email,
    ud.gender as Gender,
    ud.created_at as Signup_Date,
    first_value (wsd.created_at) over (partition by ud.id order by wsd.created_at asc rows between unbounded preceding and unbounded
following) as "First Payment Attempt",
    first_value (wsd.status) over (partition by ud.id order by wsd.created_at asc rows between unbounded preceding and unbounded following)
as "First Attempt Status",
    last_value (wsd.created_at) over (partition by ud.id order by wsd.created_at asc rows between unbounded preceding and unbounded
following) as "Last Payment Attempt",
    last_value (wsd.status) over (partition by ud.id order by wsd.created_at asc rows between unbounded preceding and unbounded following)
as "Last Attempt Status",
    amountinfo.Max_paid_amount,
    amountinfo.Min_paid_amount,
    amountinfo.Paid_amount_sum
from
    public.user_schema_users as ud
left join
    public.wallet_service_orders as wsd on ud.id = wsd.user_id
join
    amountinfo on ud.id = amountinfo.user_id
order by
    ud.id;
```

Output table:

Note: “Paid Amount” I have considered Zero in case the status is not completed. i.e. In case of Created/processing/failed, paid amount is Zero.

user_schema_users 1 X

with amountinfo as (select user_id, max(case when status = 'Completed' then paid_amount else 0 end) as paid_amount from user_schema_users group by user_id)

	123 user_id	ABC first_n	ABC last_n	ABC email	ABC ger	ABC signup_date	ABC first payment atter	ABC first attempt sta	ABC last payment atter	ABC last atter	123 max_paid	123 min_paid	123 paid_amount_sum
1	1	Paige	Burnsall	pburnsall10@refi	Female	2018-07-19 07:29:30	2018-01-04 14:50:09.000	created	2018-11-13 14:23:49.000	created	0	0	0
2	2	Maris	Queen	mqueen1@nifty	Female	2017-12-29 14:14:17	2018-02-13 12:34:01.000	processing	2018-07-18 16:07:59.000	processing	0	0	0
3	3	Janina	Matyja	jmatyja2@usgs	Female	2018-10-22 03:42:56	2018-01-15 10:38:26.000	completed	2018-12-10 01:22:22.000	completed	8,046,911	456,241	27,409,104
4	4	Rori	Sirett	rsirett3@fotki.c	Female	2018-09-14 21:23:13	2017-12-24 08:43:00.000	failed	2018-11-21 08:26:51.000	failed	0	0	0
5	5	Edee	Farquarson	efarquarson4@	Female	2018-01-17 06:55:46	2018-02-22 18:17:30.000	created	2018-12-23 03:05:58.000	created	0	0	0
6	6	Gabriele	Caitlin	gcaitlin5@exar	Male	2018-04-18 08:43:55	2018-01-19 11:12:48.000	processing	2018-12-12 05:35:30.000	processing	0	0	0
7	7	Arluene	Cumes	acumes6@ebay	Female	2018-11-15 15:43:30	2018-04-26 03:39:02.000	completed	2018-08-22 14:59:33.000	completed	6,074,464	1,144,480	11,889,918
8	8	Andy	Lillecrop	alillecrop7@go	Male	2018-01-24 14:23:01	2018-06-12 04:40:45.000	failed	2018-12-23 19:46:22.000	failed	0	0	0
9	9	Lamond	Egdal	legdal8@ehow	Male	2018-04-27 17:03:47	2018-06-05 14:15:48.000	created	2018-10-28 12:44:28.000	created	0	0	0
10	10	Shelton	Garms	sgarms9@topsy	Male	2018-12-07 08:20:49	2018-03-09 03:34:42.000	processing	2018-08-16 00:35:02.000	processing	0	0	0
11	11	Marcus	Rowson	mrowsona@sou	Male	2017-12-31 03:19:29	2018-05-26 04:23:30.000	completed	2018-11-19 04:33:51.000	completed	8,197,143	601,730	29,488,446
12	12	Delila	Burnhill	dburnhillb@ms	Female	2018-09-17 07:27:17	2018-05-03 17:22:15.000	failed	2018-11-04 09:12:47.000	failed	0	0	0
13	13	Tobey	Kynetone	tkynetonec@boo	Female	2018-02-23 16:12:21	2018-02-23 21:04:43.000	created	2018-09-24 01:44:32.000	created	0	0	0
14	14	Nils	Drever	ndreverd@ycon	Male	2018-02-13 16:07:45	2018-01-04 12:19:37.000	processing	2018-07-18 01:38:55.000	processing	0	0	0
15	15	Herby	Parlor	hparlore@earth	Male	2018-02-18 18:22:27	2018-03-15 11:31:01.000	completed	2018-12-23 15:44:30.000	completed	8,553,426	162,551	28,073,204
16	16	Malinda	Lumsdon	mlumsdonf@tir	Female	2018-04-03 10:29:30	2018-02-12 19:26:23.000	failed	2018-08-02 15:59:36.000	failed	0	0	0
17	17	Jamil	Kayzer	jkayzerg@phpb	Male	2018-06-23 21:41:17	2017-12-28 15:32:58.000	created	2018-07-08 09:41:00.000	created	0	0	0
18	18	Evan	Pye	epyeh@cnn.con	Male	2018-08-05 11:41:23	2018-01-27 10:57:40.000	processing	2018-09-20 16:34:06.000	processing	0	0	0
19	19	Vite	de Clerq	vdeclerqi@apar	Male	2018-08-05 20:02:17	2018-01-14 09:38:33.000	completed	2018-10-12 17:16:39.000	completed	9,201,090	3,364,099	34,927,243
20	20	Dollie	Kedie	dkediej@tripod	Female	2018-02-12 11:28:29	2018-01-08 23:19:31.000	failed	2018-09-08 09:55:06.000	failed	0	0	0
21	21	Merrick	Armstrong	marmstrongk@	Male	2018-02-25 07:52:20	2018-03-10 07:24:10.000	created	2018-11-30 09:55:50.000	created	0	0	0
22	22	Tracy	Denty	tdentyi@un.org	Male	2018-03-14 00:50:01	2018-02-10 21:43:08.000	processing	2018-11-09 15:24:56.000	processing	0	0	0
23	23	Anthiathia	Skuse	askusem@biglo	Female	2018-11-04 19:32:04	2018-02-14 23:29:31.000	completed	2018-11-07 20:04:40.000	completed	8,856,692	1,550,374	18,237,044
24	24	Goldy	Ferrara	gferraran@virgi	Female	2018-12-14 03:19:27	2018-02-08 20:47:46.000	failed	2018-08-23 23:46:15.000	failed	0	0	0
25	25	Priscilla	Drees	pdreeso@plala	Female	2018-05-15 19:23:30	2018-04-03 13:20:46.000	created	2018-10-15 15:50:11.000	created	0	0	0
26	26	Erina	Demeza	edemezap@car	Female	2018-05-25 08:49:02	2018-01-01 13:18:11.000	processing	2018-11-27 11:05:37.000	processing	0	0	0
27	27	Cindy	Chattock	cchattockq@pc	Female	2018-06-17 16:41:21	2018-03-05 00:47:11.000	completed	2018-10-24 12:43:27.000	completed	8,671,467	1,274,070	47,559,322
28	28	Lydon	Rollingson	lrollingsonr@pa	Male	2018-01-12 23:27:33	2018-05-29 18:17:57.000	failed	2018-12-15 21:58:55.000	failed	0	0	0
29	29	Delainey	Sievewright	dsievewrights@	Male	2018-11-03 17:31:17	2018-01-04 17:40:58.000	created	2018-11-13 01:16:16.000	created	0	0	0
30	30	Katerine	Aldus	kaldust@shop-f	Female	2017-12-31 03:04:23	2018-02-03 12:59:53.000	processing	2018-10-26 19:48:17.000	processing	0	0	0

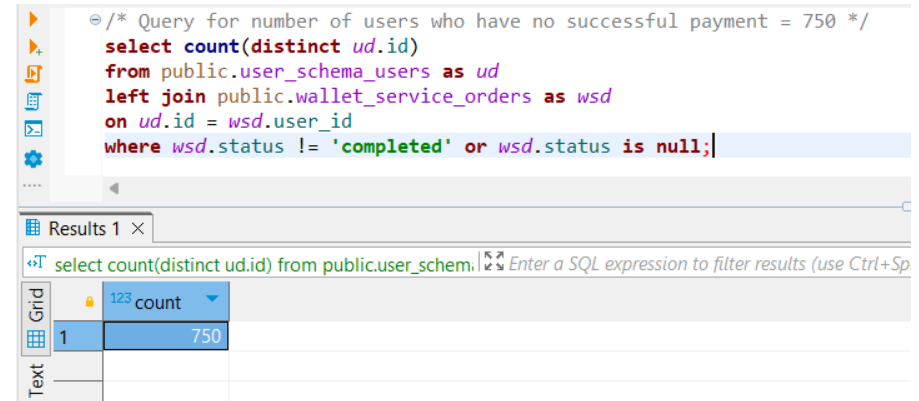
Refresh Save Cancel Export data 200 200+ 200 row(s) fetched - 4s (0.047s fetch), on 2024-05-30 at 13:58:43

Problem #2. How many users have *never* paid on AwardWallet platform?

Solution #2:

Note: I have assumed that those users who don't have "completed" as status have not Paid.

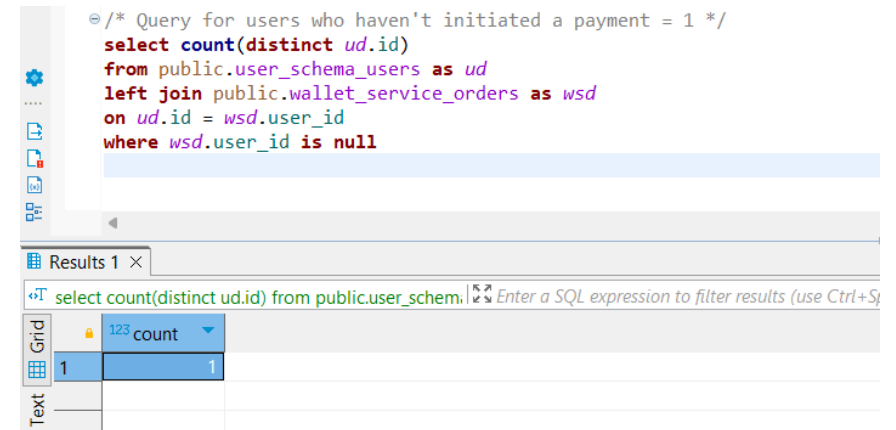
```
/* Number of users who have no successful payment = 750 */
select count(distinct ud.id)
from public.user_schema_users as ud
left join public.wallet_service_orders as wsd
on ud.id = wsd.user_id
where wsd.status != 'completed' or wsd.status is null;
```



```
/* Query for number of users who have no successful payment = 750 */
select count(distinct ud.id)
from public.user_schema_users as ud
left join public.wallet_service_orders as wsd
on ud.id = wsd.user_id
where wsd.status != 'completed' or wsd.status is null;
```

count
750

```
/* Number of users who haven't initiated a payment = 1 */
select count(distinct ud.id)
from public.user_schema_users as ud
left join public.wallet_service_orders as wsd
on ud.id = wsd.user_id
where wsd.user_id is null
```



```
/* Query for users who haven't initiated a payment = 1 */
select count(distinct ud.id)
from public.user_schema_users as ud
left join public.wallet_service_orders as wsd
on ud.id = wsd.user_id
where wsd.user_id is null
```

count
1

Problem #3. How many users have *never* claimed a reward on AwardWallet platform?

Solution #3:

```
/* Number of users who haven't initiated rewards claim = 400 */
select count(distinct ud.id)
from public.user_schema_users as ud
left join public.reward_service_reward_usages as rsd
on ud.id = rsd.user_id
where rsd.user_id is null
```

The screenshot shows a SQL IDE with the following query in the editor:

```
/* Number of users who haven't initiated rewards claim = 400 */
select count(distinct ud.id)
from public.user_schema_users as ud
left join public.reward_service_reward_usages as rsd
on ud.id = rsd.user_id
where rsd.user_id is null
```

Below the editor, the 'Results 1' tab is active, displaying a table with one row and one column:

count
400

```
/* Number of users who haven't claimed reward = 466*/
select
    (count(distinct ud.id) - (select count(distinct user_id)
                             from public.reward_service_reward_usages
                             where status = 'completed'))
    as "Reward not claimed"
from public.user_schema_users as ud
left join public.reward_service_reward_usages as rsd
on ud.id = rsd.user_id
```

The screenshot shows a SQL IDE with the following query in the editor:

```
/* Number of users who haven't claimed reward = 466*/
select
    (count(distinct ud.id) - (select count(distinct user_id)
                             from public.reward_service_reward_usages
                             where status = 'completed'))
    as "Reward not claimed"
from public.user_schema_users as ud
left join public.reward_service_reward_usages as rsd
on ud.id = rsd.user_id
```

Below the editor, the 'Results 1' tab is active, displaying a table with one row and one column:

reward not claimed
466

Problem #4: Write a query to find the cumulative amount paid till each date on the AwardWallet platform. Avoid using window functions.

Solution:



```
with neededtable as (
    select sum(amount) as amount1, created_at
    from public.wallet_service_orders
    where status = 'completed'
    group by created_at
    order by created_at
)
select t1.created_at as Date_of_payment, sum(t2.amount1)
from neededtable as t1 join neededtable as t2
on t1.created_at >= t2.created_at
group by Date_of_payment
order by Date_of_payment
```

wallet_service_orders 1 ×			
with neededtable as (select sum(amount) as amou Enter a SQL exp			
	date_of_payment	sum	
1	2017-12-24 00:12:02.000	15,607,955	
2	2017-12-24 10:33:28.000	21,963,266	
3	2017-12-24 18:16:08.000	22,651,705	
4	2017-12-25 10:38:10.000	24,652,692	
5	2017-12-26 04:38:51.000	31,927,291	
6	2017-12-26 23:48:53.000	40,254,835	
7	2017-12-27 02:48:57.000	45,188,634	
8	2017-12-27 13:24:44.000	46,656,734	
9	2017-12-27 17:04:17.000	47,093,265	
10	2017-12-27 18:33:32.000	55,062,590	
11	2017-12-27 20:53:35.000	57,949,685	
12	2017-12-28 01:30:29.000	67,193,799	
13	2017-12-29 01:55:55.000	74,687,680	
14	2017-12-29 05:53:28.000	83,785,200	
15	2017-12-29 05:58:55.000	92,362,983	
16	2017-12-29 10:33:46.000	98,028,419	
17	2017-12-29 23:56:35.000	107,223,243	
18	2017-12-30 03:39:10.000	110,791,150	
19	2017-12-30 04:43:55.000	113,456,610	
20	2017-12-30 12:02:16.000	115,714,585	
21	2017-12-30 16:10:09.000	124,793,666	
22	2017-12-31 01:25:21.000	131,762,971	
23	2017-12-31 03:50:21.000	141,428,439	
24	2017-12-31 08:31:07.000	143,712,007	
25	2017-12-31 21:23:40.000	144,965,186	
26	2018-01-01 10:20:15.000	149,800,026	
27	2018-01-01 16:41:35.000	156,433,782	
28	2018-01-02 21:02:16.000	164,188,408	
29	2018-01-02 22:52:16.000	168,175,623	
30	2018-01-03 03:03:47.000	172,700,416	
31	2018-01-03 14:10:49.000	173,610,955	

Problem #5:

DAU

```
with dt as (  
    select user_id, to_date(created_at, 'YYYY/MM/DD') as onlydate  
    from public.reward_service_reward_usages  
)  
select count(user_id) as DAU, onlydate  
from dt  
group by onlydate  
order by onlydate
```

	 123 dau ▼	 onlydate ▼
1	15	2017-12-24
2	12	2017-12-25
3	9	2017-12-26
4	19	2017-12-27
5	8	2017-12-28
6	16	2017-12-29
7	15	2017-12-30
8	13	2017-12-31
9	16	2018-01-01
10	18	2018-01-02
11	10	2018-01-03
12	17	2018-01-04
13	16	2018-01-05
14	13	2018-01-06
15	10	2018-01-07
16	14	2018-01-08
17	11	2018-01-09
18	17	2018-01-10
19	9	2018-01-11
20	11	2018-01-12
21	10	2018-01-13
22	8	2018-01-14
23	17	2018-01-15
24	13	2018-01-16
25	8	2018-01-17
26	16	2018-01-18
27	7	2018-01-19
28	13	2018-01-20
29	15	2018-01-21
30	20	2018-01-22
31	9	2018-01-23