

Queries on payeasy wallet

QUERY 1. Retrieve passbook of an user.

```
SELECT user_id , person_fname || ' ' || person_lname as name , trans_id ,
trans_description ,trans_timestamp FROM
(SELECT as4.* , person_id FROM
(SELECT as3.* , user_id FROM
(
SELECT wallet_id , trans_id , trans_descrip || ': ' || pay_type || ' ' ||
amount as trans_description ,trans_timestamp from payment NATURAL JOIN
transactions
Union
SELECT wallet_id , trans_id , trans_descrip || ': ' || comp_name || 'Plan: '
|| plan_desc|| 'Ph_no: ' || ph_no || 'Amount: ' || plan_amount as
trans_description,trans_
Union
SELECT from_wallet as wallet_id , trans_id , trans_descrip || ': ' ||
'Amount: ' || amount || ' To_Wallet: ' || to_wallet as
trans_description,trans_
Union
SELECT to_wallet as wallet_id , trans_id , trans_descrip || ': ' || 'Amount:
' || amount || ' From_Wallet: ' || from_wallet as trans_description,trans_
)as as3 NATURAL JOIN Wallet) as as4
NATURAL JOIN Users)as as5
NATURAL JOIN Person where user_id = 3;
```

QUERY 2. Retrieve complaint details which are not yet resolved.

```
SELECT * FROM COMPLAINT WHERE comp_status = 'NOT SOLVED';
```

QUERY 3. Retrieve customer details having maximum transactions .

```
SELECT user_id , person_id , person_fname || ' ' || person_lname as name ,
no_of_trans FROM
(SELECT user_id , wallet_id , person_id ,no_of_trans FROM
(
SELECT wallet_id , count(trans_id) as no_of_trans FROM
(
SELECT wallet_id , trans_id from payment
Union
SELECT wallet_id , trans_id from recharge
Union
SELECT from_wallet as wallet_id , trans_id from transfer
Union
SELECT to_wallet as wallet_id , trans_id from transfer
)as as1 group by wallet_id order by no_of_trans desc ) as as2
JOIN USERS ON as2.wallet_id = users.user_id)as as3
NATURAL JOIN Person;
```

QUERY 4. Retrieve customers on the basis of region (In which they live).

```
SELECT  person_id , person_fname || ' ' || person_lname
as name ,city FROM PERSON order by city ;
```

QUERY 5. Retrieve details of customers who didn't make any transaction.

```
SELECT user_id , person_id , person_fname || ' ' || person_lname as name
FROM
```

```
(
SELECT user_id , person_id FROM USERS

EXCEPT

SELECT user_id , person_id FROM
(
(SELECT wallet_id , count(trans_id) as no_of_trans FROM
(
SELECT wallet_id , trans_id from payment
Union
SELECT wallet_id , trans_id from recharge
Union
SELECT from_wallet as wallet_id , trans_id from transfer
Union
SELECT to_wallet as wallet_id , trans_id from transfer
)as as1 group by wallet_id order by no_of_trans desc
) as as2
JOIN
USERS ON as2.wallet_id = users.user_id )
as as3
) as as4 Natural join person;
```

QUERY 6. Retrieve most popular recharge plan.

```
SELECT * from (select comp_id , plan_id ,
count(recharge_id)as freq from Recharge Natural Join
Recharge_plan group by (comp_id,plan_id)) as re NATURAL
JOIN Telecom_company order by freq desc limit 1;
```

QUERY 7. Number of users who are teenagers and senior citizens.

```
SELECT * FROM PERSON WHERE (DATEDIFF(YEAR, PERSON_DOB,
GETDATE()))>60 OR DATEDIFF(YEAR, PERSON_DOB,
GETDATE()))<18);
```

QUERY 8. Retrieve KYC details of an user.

```
select user_id , person_fname || ' ' || person_lname as
user_name , kyc_verf_status , aadhar_card_no, pan_no
from users NATURAL JOIN person where user_id = 2;
```

QUERY 9. List out customers who have verified accounts.

```
SELECT user_id , person_fname || ' ' || person_lname as name from USERS  
NATURAL JOIN PERSON where KYC_verf_status = 'Verified';
```

QUERY 10. List of users who have made transaction with the given user

```
SELECT user_id , person_fname || ' ' || person_lname as name FROM  
(SELECT user_id , person_id FROM (SELECT user_id from transfer as t JOIN  
Wallet as w ON w.wallet_id=t.to_wallet where t.from_wallet=3  
Union  
SELECT user_id from transfer as t JOIN Wallet as w ON  
w.wallet_id=t.from_wallet where t.to_wallet=3) as t1  
NATURAL JOIN USERS) as t2  
NATURAL JOIN PERSON;
```