

Rhinceros Ltd BI Data Analyst Assessment

jpminiguez001@gmail.com [Cambiar de cuenta](#)



El nombre y la foto asociados a tu cuenta de Google se registrarán cuando subas archivos y envíes este formulario. Solo el correo que introduzcas forma parte de tu respuesta.

SQL Assessment

This exercise is to assess your understanding of SQL which you would use in your daily tasks. You're free to use any version of SQL that you're most comfortable with. The questions are based on the 2 tables in the images below.

TRANSACTIONS				
Date	TransID	UserID	Amount	TransactionType
16/09/2020	1	3245435	100	Deposit
16/09/2020	2	4323465	20	Deposit
16/09/2020	3	3245435	50	Deposit
15/09/2020	4	6542133	30	Withdraw
15/09/2020	5	2178366	300	Withdraw
15/09/2020	6	4323465	10	Deposit
15/09/2020	7	8453156	80	Deposit
...				



USERS				
UserID	RegistrationDate	City	Age	DateModified
3245435	27/06/2020	Berlin	23	27/06/2020
4323465	18/06/2020	London	54	18/06/2020
6542133	30/08/2020	Berlin	31	30/08/2020
2178366	27/09/2020	Munich	20	27/09/2020
4323465	16/09/2020	London	55	16/09/2020
8453156	18/06/2020	London	33	01/08/2020
3245435	27/06/2020	Munich	23	01/08/2020
...				

Write a query which whenever you run it will return all the Users who made a deposit in the last 30 days from the day you run it.

```
select distinct
UserID
from Transactions
where
datediff(day, tr.date, now()) <= 30 and
tr.transactionType = 'Deposit'
```

Write a query which sums transaction amount by date and user. There should be a column that sums deposits, another which sums withdrawals. Withdrawals should also be negative.

```
select
UserID,
Date,
sum(case when transactionType = 'Deposit' then amount else 0 end) as S_Deposit,
```



```
-sum(case when transactionType='Withdraw' then amount else 0 end) as S_Withdraw  
from Transactions  
group by  
UserID,  
Date
```

Write query that sums and counts deposits per user registration date

--We deduplicate the UserID in the table USERS with USERS_V2

with USERS_V2 as (select distinct userID, registrationDate from USERS)

```
select  
id. registrationDate,  
count(tr.*),  
sum(tr.amount)  
from  
USERS_V2 id  
left join  
Transactions tr  
on id.UserID = tr.UserID  
where  
tr.transactionType = 'deposit'
```

Write a query that shows all the users and their latest information.

--We deduplicate the UserID in the table USERS with USERS_V2 (same than previous one)

--display all the users (they will appear with NULL in case of no transaction)

with USERS_V2 as (select distinct userID from USERS)

```
select  
id.UserID  
Max(tr.date)  
from  
USERS_V2 id  
left join  
Transactions tr  
on id.UserID = tr.UserID
```



What are some checks you would perform to ensure data quality after running a query or inserting data into a table?

- UserID in USERS table should not have NULLS or Duplicates (primary key UserID). We avoid the duplication you have in the USERS table
- the UserID for new transaction should exist in the User table UserID field (foreign key)
- All the dates should be date
- One check in the age field (<100 years) or at least one warning msg.
- The amount in the transaction table should be positive.
- The transaction type should be foreign key to another table with the closed list for the type of transactions

Página 2 de 4

Atrás

Siguiente

Borrar formulario

Nunca envíes contraseñas a través de Formularios de Google.

Este formulario se creó en Epiceros Development AB. [Notificar uso inadecuado](#)

Google Formularios

