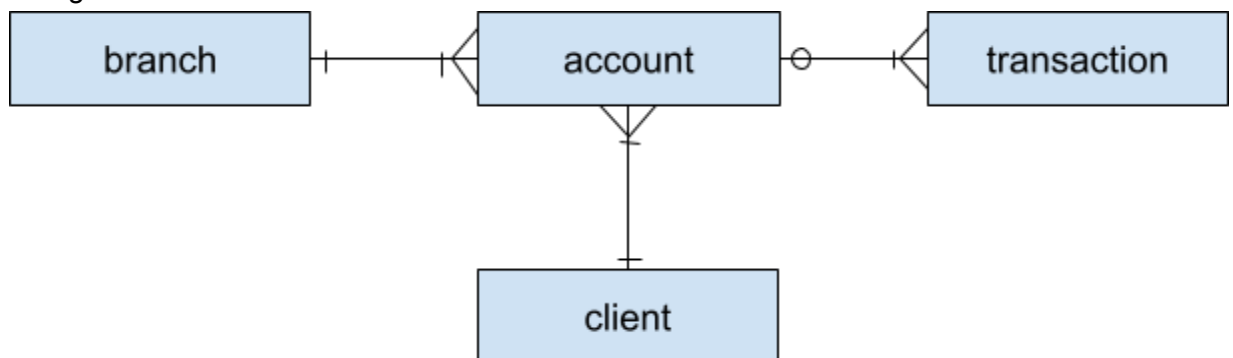


## Tugas 2

- Langkah-langkah yang dapat digunakan pada MySQL untuk mengimpor seluruh file csv ke dalam tabel, yaitu:
  - Misalkan pada MySQL sudah dibuat database dan tabel nya terlebih dahulu (dengan menggunakan CREATE DATABASE dan CREATE TABLE)
  - Jika tabel sudah dibuat maka, pilih tabel yang ingin diimpor datanya, misalkan tabel account. Klik kanan pada tabel pada jendela schemas dan pilih Table Data Import Wizard.
  - Akan ada jendela yang muncul untuk mengarahkan user untuk memilih file csv, yaitu dengan meng-klik Browse. Pilih file account.csv.
  - Jika sudah, klik next, dan akan diberikan dua pilihan yaitu menggunakan tabel yang sudah ada (use existing table) atau membuat tabel baru (create new table), karena sebelumnya sudah membuat tabel maka pilih pilihan pertama.
  - Klik next dan akan muncul tampilan untuk mencocokkan nama kolom yang sudah dibuat sebelumnya dengan nama kolom yang ada di file csv. Jika sudah dirasa cocok, maka cukup klik next saja.
  - Klik next terus hingga data jendela import sudah menunjukkan bahwa data berhasil diimpor.
  - Lakukan hal yang sama untuk seluruh tabel.
- Dengan menganggap semua tabel adalah entitas maka akan dibuat diagram E-R nya sebagai berikut:



Jika dijabarkan hubungan antara empat entitas tersebut maka:

- Branch - Account: setiap akun hanya boleh dibuat pada satu branch, namun satu branch dapat menangani lebih dari satu akun
- Client - Account: klien boleh memiliki lebih dari satu akun, namun setiap akun hanya terasosiasi dengan satu klien
- Transaction - Account: satu akun dapat melakukan banyak transaksi dan setiap transaksi pasti terhubung dengan satu akun, namun akun diperbolehkan untuk tidak melakukan transaksi apapun.

3. Misal tabel A adalah “satu” dan tabel B adalah “banyak”, maka mungkin saja jika tabel A memiliki lebih banyak baris daripada tabel B. Semua tergantung dengan bentuk data dan partisipasinya. Misalkan tabel A adalah tabel customer yang memiliki 200 baris dan tabel B adalah tabel transaksi yang memiliki 100 baris. Satu customer dapat melakukan banyak transaksi, namun satu customer diperbolehkan untuk tidak melakukan transaksi sama sekali. Sehingga hubungan antara kedua tabel tetap satu-ke-banyak, namun partisipasi customer adalah parsial sedangkan partisipasi transaksi adalah total.

4. Perintah SQL untuk memberikan akses yang sesuai kepada rekan saya, yaitu:  
CREATE VIEW transaction\_view AS  
SELECT amount\_transaction, date\_issued, transaction\_type\_id  
FROM transaction;  
Dari perintah SQL tersebut, rekan saya bisa mengakses informasi seperti kapan transaksi dibuat, banyaknya uang yang dikeluarkan untuk transaksi, dan tipe transaksinya apa.

5. Tabel yang berisikan jumlah akun berdasarkan bulan akun dibuat:

bulan_dibuat	jumlah_akun
2023-07	10
2023-08	24
2023-09	5
2023-10	4

Kode SQL:

```
SELECT  
  DATE_FORMAT(created_at, '%Y-%m') AS bulan_dibuat,  
  COUNT(*) AS jumlah_akun  
FROM account  
GROUP BY bulan_dibuat  
ORDER BY bulan_dibuat ASC;
```

6. Tabel yang berisikan urutan nama jenis akun berdasarkan jumlah akun yang terdaftar:

jenis_akun	jumlah_akun
Savings	9
Online Banking	6
Student	5
Health Saving (HSA)	5
Business	4
Custodial	3
Certificate of Deposit (CD)	3
Money Market	3
Checking	3
Trust	2

Kode SQL:

```
SELECT
```

```

a_t.account_type_name AS jenis_akun,
COUNT(*) AS jumlah_akun
FROM account AS a
JOIN account_types AS a_t ON a.account_type_id = a_t.ID
GROUP BY jenis_akun
ORDER BY jumlah_akun DESC;

```

Jika dilihat dari hasil yang diberikan, kebanyakan client memilih membuka akun rekening yang diperuntukkan sebagai tabungan ataupun yang dikhususkan untuk melakukan transaksi online

7. Tabel yang berisikan seluruh informasi cabang dan jumlah akun berdasarkan cabang:

branch_ID	branch_name	location	manager	jumlah_akun
3	Main Branch Munich	Munich	Anna	2
6	Main Branch cologne	Cologne	Fritz	3
4	Main Branch Hamburg	Hamburg	Emilia	4
7	Main Branch Leipzig	Leipzig	Ben	4
1	Main Branch Berlin	Berlin	Maximilian	5
2	Main Branch Stuttgart	Stuttgart	Paul	5
5	Main Branch Frankfurt	Frankfurt	Lukas	5
8	Main Branch Dresden	Dresden	Jonas	5
9	Main Branch Bonn	Bonn	Emma	5
10	Main Branch Chemnitz	Chemnitz	Hans	5

Kode SQL:

```

SELECT
    b.*,
    COUNT(a.ID) AS jumlah_akun
FROM branch as b LEFT JOIN account AS a
ON b.branch_ID = a.branch_ID
GROUP BY b.branch_ID
ORDER BY jumlah_akun ASC;

```

8. Tabel yang berisi nama lengkap klien, total saldo berdasarkan jenis akun, dan nama jenis akunnya:

nama_lengkap	jenis_akun	total_saldo
Alexander Huber	Savings	40
Alexander Huber	Student	1234
BELINDA JOHNSON	Certificate of Deposit (CD)	2000
BELINDA JOHNSON	Online Banking	200
BELINDA JOHNSON	Online Banking	620
BELINDA JOHNSON	Savings	915
DAVID SMITH	Checking	6424
Dominik Richter	Health Saving (HSA)	7899
ELLIS KIRK	Certificate of Deposit (CD)	20
ELLIS KIRK	Checking	880
ELLIS KIRK	Health Saving (HSA)	880

nama_lengkap	jenis_akun	total_saldo
ELLIS KIRK	Money Market	70
ELLIS KIRK	Online Banking	80
ELLIS KIRK	Savings	330
ELLIS KIRK	Savings	880
ELLIS KIRK	Savings	945
ELLIS KIRK	Student	724
ELLIS KIRK	Trust	100
JANET SPEARS	Custodial	545
JANET SPEARS	Health Saving (HSA)	357
JESSICA RALPH	Certificate of Deposit (CD)	444
JESSICA RALPH	Savings	9999
JOELLE BECK	Health Saving (HSA)	546
JOELLE BECK	Money Market	344
JOELLE BECK	Savings	788
JOELLE BECK	Student	926
JOELLE BECK	Student	6463
KEVIN MCCABE	Custodial	80
KEVIN MCCABE	Online Banking	3832
KEVIN MCCABE	Savings	2000
Laura Hoffmann	Business	5419
SALLY MITCHELL	Business	220
SALLY MITCHELL	Business	5000
SALLY MITCHELL	Checking	888
SALLY MITCHELL	Custodial	820
SALLY MITCHELL	Trust	1340
Sarah Schmidt	Online Banking	4567
SCOTTY ARMOUR	Health Saving (HSA)	4555
SCOTTY ARMOUR	Online Banking	781
SCOTTY ARMOUR	Student	256
TIMOTHY HEAD	Business	4567
TIMOTHY HEAD	Money Market	345
TIMOTHY HEAD	Savings	440

Kode SQL:

```
SELECT
  CONCAT(c.first_name, ' ', c.last_name) AS nama_lengkap,
  a_t.account_type_name AS jenis_akun,
  a.balance_account AS total_saldo
FROM client AS c
JOIN account AS a ON c.client_ID = a.client_id
JOIN account_types AS a_t ON a.account_type_id = a_t.ID
GROUP BY nama_lengkap, jenis_akun, total_saldo
ORDER BY nama_lengkap;
```

9. Berikut ini adalah hasil cuplikan yang diberikan oleh SQL

jenis_kelamin	jumlah_transaksi
F	48
M	14

Dari hasil tersebut dapat dilihat bahwa klien berjenis kelamin perempuan lebih banyak melakukan transaksi dibandingkan dengan klien yang berjenis kelamin laki-laki.

Kode SQL:

```
SELECT
  c.sex AS jenis_kelamin,
  COUNT(t.id) AS jumlah_transaksi
FROM transactions AS t
JOIN account AS a ON t.source_account_id = a.id
JOIN client AS c ON a.client_id = c.client_id
GROUP BY c.sex
ORDER BY jumlah_transaksi DESC;
```

10. Berdasarkan jawaban pada nomor 9, yaitu wanita yang melakukan transaksi lebih banyak, jenis transaksi yang paling banyak dilakukan oleh wanita adalah transfer.

jenis_transaksi	jumlah_transaksi
Transfer	27
ATM Deposit	9
ATM Withdrawal	8
Balance Inquiry	4

Kode SQL:

```
SELECT
  t_t.transaction_type AS jenis_transaksi,
  COUNT(t.ID) AS jumlah_transaksi
FROM transactions AS t
JOIN transaction_types AS t_t ON t.transaction_type_id = t_t.ID
JOIN account AS a ON t.source_account_id = a.ID
JOIN client AS c ON a.client_id = c.client_id
WHERE c.sex = 'F'
GROUP BY t_t.transaction_type
ORDER BY jumlah_transaksi DESC;
```