1. Tampilkan nomor KTP, nama pengguna, beserta tipe akun yang memiliki limit kurang dari 5 juta
2. Tampilkan nama pemilik akun beserta nama petugasnya bagi setiap transaksi yang dilayani oleh Sinaga Bakianto
3. Tampilkan rata rata peminjaman yang dilakukan oleh pengguna perempuan
4. Tampilkan nama pegawai yang paling banyak melayani transaksi
5. Tampilkan nomor KTP dan nama pengguna yang tidak pernah memilih layanan cicilan 12 bulan, diurutkan berdasarkan pendaftar pertama
6. Tampilkan nomor KTP, nama serta tanggal pendaftaran dari 5 pengguna yang mendaftar pertama kali, diurutkan berdasarkan tanggal pendaftarannya
7. Tampilkan banyaknya transaksi peminjaman dengan tipe cicilan selama 12 bulan
8. Tampilkan nomor KTP, nama, serta banyak transaksi peminjaman yang telah dilakukan oleh pengguna, diurutkan berdasarkan nama pengguna
9. Tampilkan nama, gaji, dan limit akun yang dimiliki oleh 10 pendaftar pertama
10. Tampilkan nama pengguna yang belum pernah melakukan transaksi
11. Tampilkan ID dan tanggal dari data transaksi peminjaman yang dilakukan antara bulan Januari-Juni tahun 2021
12. Tampilkan nama serta gaji pengguna yang memiliki limit pinjaman sebesar 4 juta
13. Tampilkan banyaknya transaksi yang dilakukan berdasarkan tipe peminjamannya
14. Tampilkan id, besar pinjaman, beserta jumlah cicilan yang telah dibayar, diurutkan berdasarkan nomor id transaksi
15. Tampilkan rata-rata usia pengguna yang melakukan transaksi peminjaman
16. Tampilkan nama dan email pegawai yang memiliki email selain @yahoo dan @gmail
17. Tampilkan tanggal transaksi beserta nama pegawai yang melayani akun pada bulan agustus 2020
18. Tampilkan nama dan jumlah uang yang dipinjam oleh masing-masing pengguna
19. Tampilkan nama pengguna dan banyaknya cicilan yang belum dibayar, diurutkan dari banyaknya sisa cicilan yang paling banyak
20. Tampilkan nama pengguna yang telah melewati masa peminjaman. Lengkapi informasi tersebut dengan tanggal jatuh temponya

Ket :

1. Simple Query
2. JOIN
3. Agregasi dan group by
4. join dan order by
5. nested query
6. Tampilkan nomor KTP, nama pengguna dari akun yang memiliki limit kurang dari 5 juta. Lengkapi informasi tersebut dengan nilai limit akunnya

* Query :

select ac.ACC\_KTP\_NUM, ac.acc\_name, at.at\_limit

from account ac, account\_type at

where ac.acc\_at\_id = at.at\_id

AND at.AT\_LIMIT < "5000000";

* Aljabar Relasional :

* Hasil :



1. Tampilkan nama pemilik akun yang transaksi dilayani oleh Sinaga Bakianto

* Query :

select a.acc\_name

from transaction t

inner join employee e on t.t\_e\_ktp\_num = e.e\_ktp\_num

inner join account a on t.t\_acc\_ktp\_num = a.acc\_ktp\_num

where e.e\_name = "Sinaga Bakianto";

* Aljabar Relasional :
* Hasil : 

1. Tampilkan rata rata peminjaman yang dilakukan oleh pengguna perempuan

* Query :

select ROUND (SUM(t.t\_loan) / (COUNT(distinct a.acc\_ktp\_num)), 2)

as "Nilai rata rata"

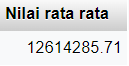
from account a, transaction t

where t.t\_acc\_ktp\_num = a.acc\_ktp\_num

and a.acc\_gender = "P";

* Aljabar Relasional

* Hasil :



1. Tampilkan nama pegawai yang paling banyak melayani transaksi beserta jumlah transaksinya

* Query :

select e.e\_name, COUNT(t.t\_id) as "Jumlah Transaksi"

from employee e

inner join transaction t

on e.e\_ktp\_num = t.t\_e\_ktp\_num

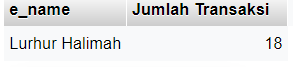
group by e.e\_name

order by COUNT(t.t\_id) DESC

limit 1;

* Aljabar Relasional :

* Hasil :



1. Tampilkan No KTP dan nama pengguna yang hanya memilih layanan cicilan 3 atau 6 bulan, diurutkan berdasarkan pendaftar pertama

* Query :

SELECT ACC\_KTP\_NUM, ACC\_NAME

FROM account

WHERE ACC\_KTP\_NUM NOT IN(

SELECT DISTINCT T\_ACC\_KTP\_NUM

FROM transaction

WHERE T\_LT\_ID IN (

SELECT LT\_ID

FROM loan\_type

WHERE LT\_NAME = '12 BULAN')

)

ORDER BY ACC\_REG\_DATE ASC;

* Aljabar Relasional :



* Hasil :

1. Tampilkan no KTP dan nama pengguna yang tidak mengupload file sesuai dengan persyaratan

* Query :

SELECT ACC\_KTP\_NUM, ACC\_NAME

FROM account

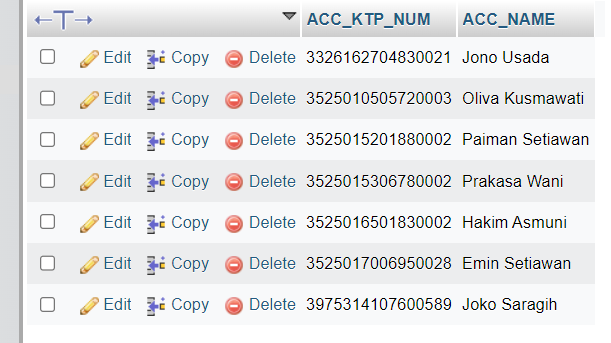
WHERE ACC\_KTP\_SCAN IS NULL

OR ACC\_NPWP\_SCAN IS NULL

OR ACC\_SELFIE IS NULL

OR ACC\_BANK\_NUM\_SCAN IS NULL;

* Aljabar Relasional
* Hasil :



1. Tampilkan no ktp dan nama setiap pengguna yang memilih tipe cicilan selama 12 bulan

* Query :

SELECT ACC\_KTP\_NUM, ACC\_NAME

FROM loan\_type lt

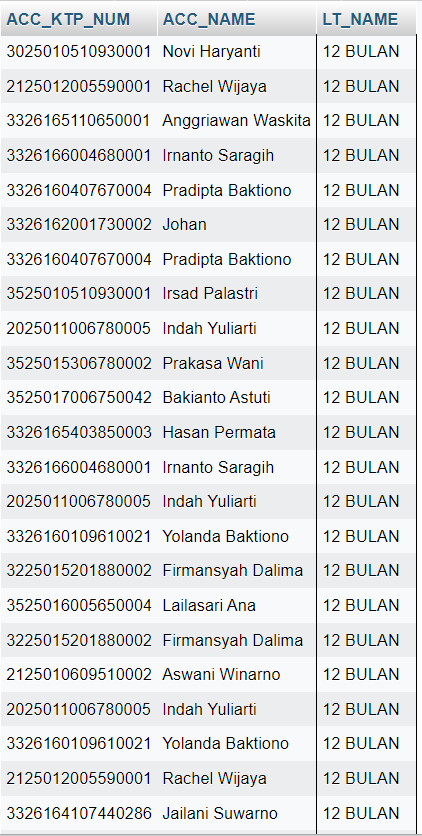
INNER JOIN transaction tr ON tr.T\_LT\_ID = lt.LT\_ID

INNER JOIN account ac ON ac.ACC\_KTP\_NUM = tr.T\_ACC\_KTP\_NUM

WHERE LT\_NAME = '12 BULAN';

* Aljabar Relasional

* Hasil :



1. Tampilkan No KTP, nama, serta banyak transaksi peminjaman yang telah dilakukan oleh pengguna, urutkan berdasarkan jumlah transaksi

* QUERY :

SELECT ACC\_KTP\_NUM,ACC\_NAME, COUNT(T\_ID) AS TOTAL

FROM account ac

LEFT JOIN transaction tr

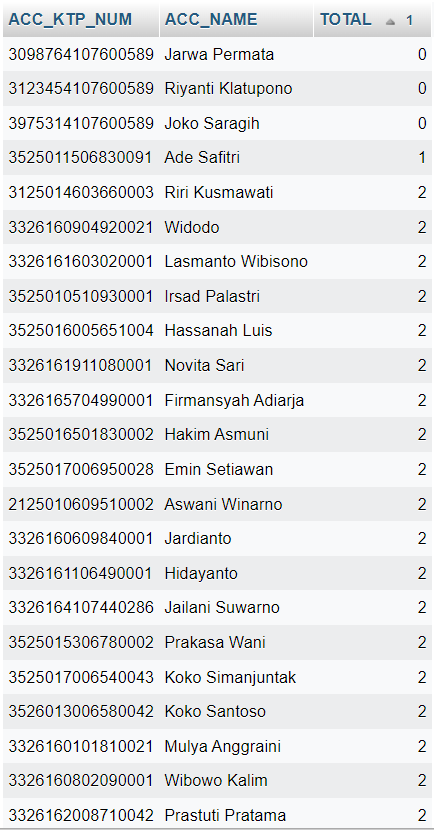
ON ac.ACC\_KTP\_NUM = tr.T\_ACC\_KTP\_NUM

GROUP BY ACC\_KTP\_NUM

ORDER BY TOTAL ASC;

* Aljabar Relasional

* HASIL :



1. Tampilkan NO KTP dan nama pengguna yang telat melakukan pembayaran cicilan (penalti), sertakan juga berapa kali dia terkena penalti dan banyak uang pinalti yang harus dibayarkan

* QUERY:

SELECT ACC\_KTP\_NUM, ACC\_NAME, SUM(T\_PENALTY\_COUNT) AS PENALTI, SUM(T\_PENALTY\_COUNT\*LT\_PENALTY) AS TOTAL\_UANG\_PENALTI

FROM account ac

INNER JOIN transaction tr ON tr.T\_ACC\_KTP\_NUM = ac.ACC\_KTP\_NUM

INNER JOIN loan\_type lt ON lt.LT\_ID = tr.T\_LT\_ID

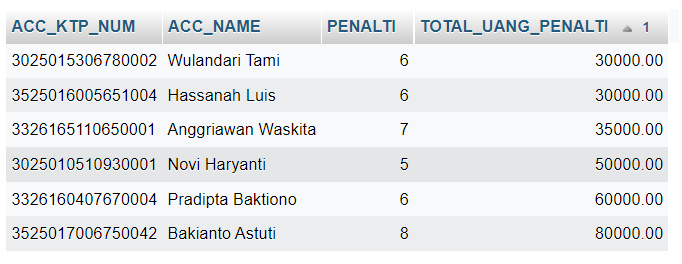
WHERE T\_PENALTY\_COUNT <> 0

GROUP BY ACC\_KTP\_NUM

ORDER BY TOTAL\_PENALTI

* Aljabar relasional

* HASIL :



1. Tampilkan nama pengguna yang belum pernah melakukan transaksi

* QUERY :

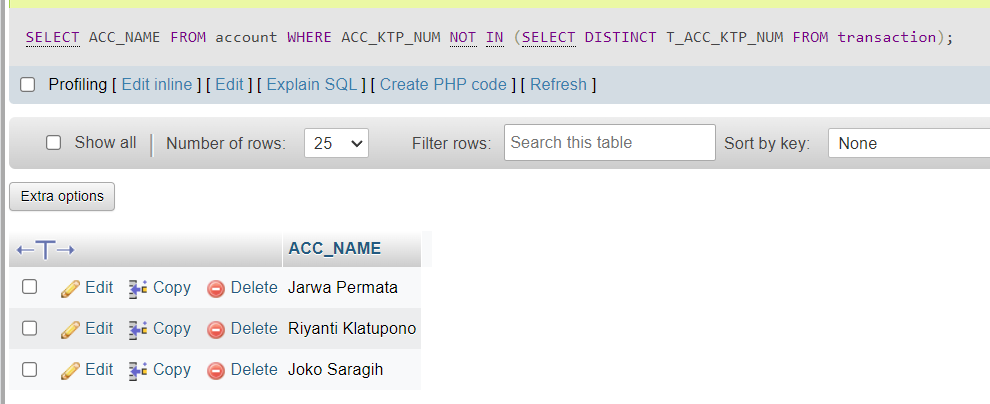
SELECT ACC\_NAME

FROM account

WHERE ACC\_KTP\_NUM NOT IN (

SELECT DISTINCT T\_ACC\_KTP\_NUM

FROM transaction)

* ALJABAR RELASIONAL
* HASIL :

1. Tampilkan ID dan tanggal peminjaman dari data transaksi peminjaman yang dilakukan antara bulan Januari-Juni tahun 2021

* QUERY

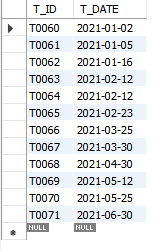
SELECT T\_ID, T\_DATE

FROM Transaction

WHERE MONTH(t\_date) <= 6 AND YEAR(t\_date) = 2021;

* Aljabar Relasional

* HASIL



1. Tampilkan nama pengguna yang telah melewati masa peminjaman. Lengkapi informasi tersebut dengan tanggal jatuh temponya

* QUERY

SELECT ACC\_NAME, T\_DUE\_DATE

FROM account

INNER JOIN transaction

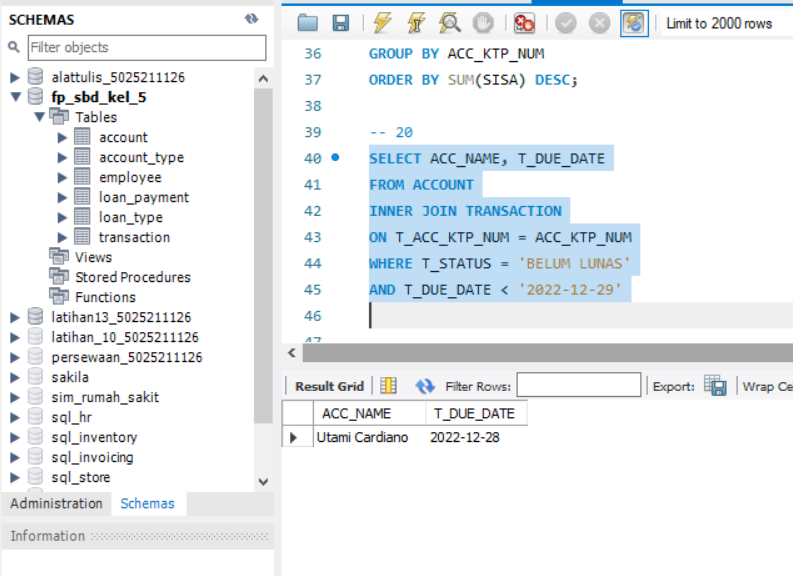
ON T\_ACC\_KTP\_NUM = ACC\_KTP\_NUM

WHERE T\_STATUS = 'BELUM LUNAS'

AND T\_DUE\_DATE < '2022-12-29';

* Aljabar Relasional

* HASIL



1. Tampilkan banyaknya transaksi yang dilakukan berdasarkan tipe peminjamannya yaitu 3 bulan, 6 bulan, 12 bulan

* QUERY

SELECT LT\_NAME,COUNT(T\_LT\_ID) AS Banyak\_Transaksi

FROM transaction,loan\_type

WHERE LT\_ID = T\_LT\_ID

GROUP BY LT\_ID;

* Aljabar Relasional

* HASIL



1. Tampilkan id, besar pinjaman, beserta uang cicilan yang telah dibayar, diurutkan berdasarkan nomor uang cicilan terbesar

* QUERY

SELECT T\_ID, T\_LOAN, SUM(T\_MONTHLY\_PAYMENT) AS TOTAL\_PAYMENT

FROM transaction INNER JOIN loan\_payment

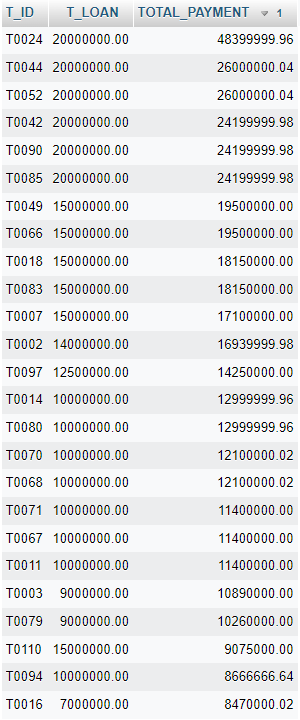
ON T\_ID = LP\_T\_ID

GROUP BY T\_ID

ORDER BY TOTAL\_PAYMENT DESC;

* Aljabar Relasional

* HASIL



1. Tampilkan rata-rata usia pengguna yang melakukan transaksi peminjaman

* QUERY

SELECT ROUND(AVG((TIMESTAMPDIFF(YEAR,ACC\_DATE\_OF\_BIRTH,NOW()))),2) AS "UMUR RATA RATA"

FROM account

WHERE ACC\_KTP\_NUM IN (

SELECT DISTINCT T\_ACC\_KTP\_NUM

FROM transaction );

* Aljabar Relasional

* HASIL



1. Tampilkan nama pengguna yang memiliki rating tidak sempurna, sertakan juga ratingnya lalu urutkan berdasarkan pengguna yang memiliki rating terendah

* QUERY

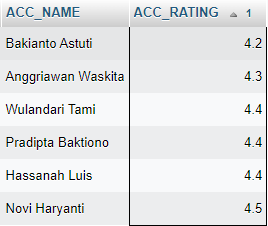
SELECT ACC\_NAME, ACC\_RATING

FROM ACCOUNT

WHERE ACC\_RATING < 5

ORDER BY ACC\_RATING;

* ALJABAR RELASIONAL
* HASIL



1. Tampilkan tanggal transaksi beserta nama pegawai yang melayani pengguna pada bulan agustus 2020

* Query

SELECT T\_DATE, E\_NAME

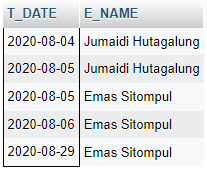
FROM TRANSACTION

INNER JOIN EMPLOYEE

ON T\_E\_KTP\_NUM = E\_KTP\_NUM

WHERE MONTH(T\_DATE) = 8 AND YEAR(T\_DATE) = 2020;

* ALJABAR RELASIONAL
* HASIL



1. Tampilkan banyaknya transaksi setiap bulan, urutkan berdasarkan transaksi paling banyak

* Query

SELECT MONTHNAME(T\_DATE), YEAR(T\_DATE), COUNT(\*)

FROM TRANSACTION

GROUP BY YEAR(T\_DATE), MONTH(T\_DATE)

ORDER BY COUNT(\*) DESC;

* Aljabar Relasional
* HASIL



1. Tampilkan nama dan jumlah uang yang dipinjam oleh masing-masing pengguna, diurutkan dari jumlah peminjaman terbanyak

* QUERY

SELECT ACC\_NAME, SUM(T\_LOAN) AS 'TOTAL PEMINJAMAN'

FROM ACCOUNT

INNER JOIN TRANSACTION

ON ACC\_KTP\_NUM = T\_ACC\_KTP\_NUM

GROUP BY ACC\_KTP\_NUM

ORDER BY SUM(T\_LOAN) DESC;

* ALJABAR RELASIONAL
* HASIL



1. Tampilkan nama pengguna dan banyaknya cicilan yang belum dibayar, diurutkan dari banyaknya sisa cicilan yang paling banyak

* Query

SELECT ACC\_NAME, T\_ID, SUM(SISA)

FROM ACCOUNT,

(

SELECT \*, COUNT(LP\_ID), (LT\_PERIOD - COUNT(LP\_ID)) AS SISA

FROM TRANSACTION

INNER JOIN LOAN\_TYPE

ON T\_LT\_ID = LT\_ID

LEFT JOIN LOAN\_PAYMENT

ON LP\_T\_ID = T\_ID

WHERE T\_STATUS = 'BELUM LUNAS'

GROUP BY T\_ID

ORDER BY SISA

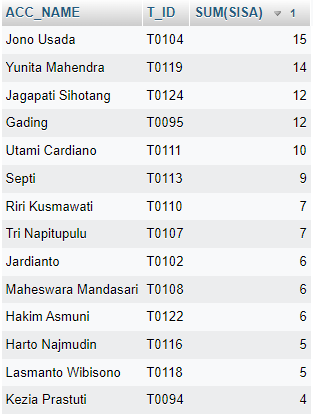
) AS SC

WHERE T\_ACC\_KTP\_NUM = ACC\_KTP\_NUM

GROUP BY ACC\_KTP\_NUM

ORDER BY SUM(SISA) DESC;

* Aljabar Relasional
* HASIL



Ket :

1. Simple Query
2. JOIN
3. Agregasi dan group by
4. join dan order by
5. nested query

Catatan

- Pencatatan denda pinjaman

- Pencatatan kontak darurat dari peminjam

- Perhitungan rating peminjam

- Saat verifikasi, pengguna mengupload dokumen seperti foto selfie, foto/scan KTP, NPWP, scan buku rekening. Aplikasi memiliki daftar jenis dokumen tersebut dan pengguna harus mengunggah dokumen-dokumen tersebut. Apabila belum lengkap, maka sistem bisa tahu dokumen apa yang belum lengkap diupload

SELECT ACC\_KTP\_NUM, ACC\_NAME, SUM(T\_PENALTY\_COUNT), SUM(T\_PENALTY\_COUNT\*LT\_PENALTY) AS TOTAL\_PENALTI

FROM account ac

INNER JOIN transaction tr ON tr.T\_ACC\_KTP\_NUM = ac.ACC\_KTP\_NUM

INNER JOIN loan\_type lt ON lt.LT\_ID = tr.T\_LT\_ID

WHERE T\_PENALTY\_COUNT <> 0

GROUP BY ACC\_KTP\_NUM

ORDER BY TOTAL\_PENALTI