

LATIHAN MODUL 4
PRAKTIKUM BASIS DATA
JOIN PART I

Oleh:

Nama : Fitra Ilyasa
Nim : 120140048
Kelas : Basis Data RB



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK ELEKTRO INFORMATIKA DAN SISTEM FISIS
INSTITUT TEKNOLOGI SUMATERA
LAMPUNG SELATAN
2022

BAB I

TEORI DASAR

Relationship

Relationship adalah suatu hubungan antara beberapa entitas. Konsep ini sangat penting sekali di dalam basis data, di mana memungkinkan entitas-entitas untuk saling berhubungan satu sama lain. Didalam sebuah *relationship*, *primary key* memiliki peran penting untuk mengaitkan entitas. Selain itu, *primary key* juga digunakan untuk mendefinisikan batasan keterhubungan.

Join

Join merupakan salah satu konstruksi dasar dari SQL dan basis data. *Join* dapat didefinisikan sebagai kombinasi record dari dua atau lebih tabel di dalam basis data relasional dan menghasilkan sebuah tabel (*temporary*) baru yang disebut sebagai *joined tabel*. *Join* digunakan untuk menghubungkan data yang diambil dari tabel-tabel melalui sebuah kolom yang menghubungkan mereka.

Ada beberapa Jenis JOIN yang dapat digunakan seperti CROSS JOIN, INNER JOIN, STRAIGHT JOIN, OUTER JOIN (RIGHT dan LEFT JOIN), NATURAL JOIN.

a) Cross Join

Cross join merupakan operasi *join* yang paling dasar. Menghasilkan kombinasi semua baris yang terdapat pada tabel-tabel yang digabungkan, baik yang memiliki pasangan maupun yang tidak. *Join* jenis ini juga disebut dengan istilah *cartesian product*.

Bila menggunakan *cross join*, maka hasil dari *cross join* akan menciptakan hasil yang didasarkan pada semua kemungkinan kombinasi baris dalam kedua set data. Dengan demikian, jumlah baris yang dikembalikan adalah $N \times M$, dimana N adalah jumlah baris dalam kumpulan data A dan M jumlah baris dalam kumpulan data B. Jelas, jumlah baris dalam *cross join* dapat menjadi sampah.

Atau :

```
SELECT <nama_kolom> FROM <tabel_1>,<tabel_2>;
```

```
SELECT <nama_kolom> FROM <tabel_1> CROSS JOIN <tabel_2>
```

b) Inner Join

Inner join Mengembalikan baris-baris dari dua tabel atau lebih yang memenuhi syarat. Umumnya penggunaanya didasarkan pada kunci tamu.

Inner join pada dasarnya adalah menemukan persimpangan (*intersection*) antara dua buah tabel.

Jika menggunakan *inner join* antara tabel A dan B maka hasilnya hanya akan menampilkan data yang memiliki pasangan saja atau yang memiliki kesamaan saja.

```
SELECT <nama_kolom> FROM <tabel_1> INNER JOIN <tabel_2> ON 1.NamaKolom = 2.NamaKolom;
```

atau

```
SELECT <nama_kolom> FROM <tabel_1> INNER JOIN <tabel_2>  
WHERE tabel_1.kolom_kunci = tabel_2.kolom_kunci;
```

c) Natural Join

Operasi INNER JOIN yang dilakukan pada semua atribut yang sama dari kedua tabel (nama dan domainnya). Pada tabel hasil, atribut-atribut yang sama hanya akan muncul sekali. Dalam menggabungkan tabel, operasi NATURAL JOIN tidak memerlukan penyebutan atribut penghubung. Penggabungan tabel dilakukan otomatis pada atribut yang sama (nama dan domainnya).

```
SELECT <nama_kolom> FROM <tabel_1> NATURAL JOIN <tabel_2>;
```

BAB II

PEMBAHASAN & ANALISIS

1. Buat dan Tambahkan tabel “nasabah” pada Database Bank_Itera dan isi nilai pada setiap kolom seperti berikut.

Command Line : CREATE TABLE Nasabah (Id_Nasabah int(10), no_rekening varchar(20), Nama_Nasabah varchar(50), Alamat varchar(60), No_Telepon varchar(20), PRIMARY KEY(Id_nasabah)) ENGINE = InnoDB;

Command Line : INSERT INTO Nasabah value (1, "145678", "Puspa Sari", "Sei Batang Kuis", "081376809122"), (2, "256132", "Sinta Jojo", "Iskandar Muda", "089987654311"), (3, "247182", "Bunga Putri", "Jalan Martapura", "081249987011"), (4, "350067", "Grace Sebayang", "Gang Keluarga", "085276812453"), (5, "114115", "Christine Talia", "Sei Kambah", "082266778866"), (6, "100111", "Rimayka Natalia", "Perumahan Puri K", "081234567890");

Command Line : SELECT * FROM Nasabah;

```
XAMPP for Windows - mysql -u root -p
MariaDB [(none)]> create database bank_itera;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]> use bank_itera;
Database changed
MariaDB [bank_itera]> CREATE TABLE Nasabah (Id_Nasabah int(10), no_rekening varchar(20), Nama_Nasabah varchar(50), Alamat varchar(60), No_Telepon varchar(20), PRIMARY KEY(Id_nasabah)) ENGINE = InnoDB;
Query OK, 0 rows affected (0.276 sec)

MariaDB [bank_itera]> INSERT INTO Nasabah value (1, "145678", "Puspa Sari", "Sei Batang Kuis", "081376809122"), (2, "256132", "Sinta Jojo", "Iskandar Muda", "089987654311"), (3, "247182", "Bunga Putri", "Jalan Martapura", "081249987011"), (4, "350067", "Grace Sebayang", "Gang Keluarga", "085276812453"), (5, "114115", "Christine Talia", "Sei Kambah", "082266778866"), (6, "100111", "Rimayka Natalia", "Perumahan Puri K", "081234567890");
Query OK, 6 rows affected (0.080 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [bank_itera]> SELECT * FROM Nasabah;
+-----+-----+-----+-----+-----+
| Id_Nasabah | no_rekening | Nama_Nasabah | Alamat | No_Telepon |
+-----+-----+-----+-----+-----+
| 1 | 145678 | Puspa Sari | Sei Batang Kuis | 081376809122 |
| 2 | 256132 | Sinta Jojo | Iskandar Muda | 089987654311 |
| 3 | 247182 | Bunga Putri | Jalan Martapura | 081249987011 |
| 4 | 350067 | Grace Sebayang | Gang Keluarga | 085276812453 |
| 5 | 114115 | Christine Talia | Sei Kambah | 082266778866 |
| 6 | 100111 | Rimayka Natalia | Perumahan Puri K | 081234567890 |
+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)
```

2. Masukkan data berikut ke dalam tabel “rekening”

Command Line : CREATE TABLE Rekening (no_rekening varchar(20), Kode_cabang varchar(20), Pin varchar(20), Saldo int(10), PRIMARY KEY(no_rekening)) ENGINE = InnoDB;

Command Line : INSERT INTO Rekening value ("145678", "SGS02", "908723", 1000000), ("256132", "KKM70", "340987", 230000), ("247182", "JKT48", "091298", 4000000), ("350067", "RGS02", "230987", 5500000), ("114115", "SGS01", "203010", 500000), ("100111", "KKM69", "213671", 998500);

Command Line : SELECT * FROM Rekening;

```

XAMPP for Windows - mysql -u root -p
MariaDB [bank_itera]> CREATE TABLE Rekening (no_rekening varchar(20), Kode_cabang varchar(20), Pin varchar(20), Saldo int(10), PRIMARY
KEY(no_rekening)) ENGINE = InnoDB;
Query OK, 0 rows affected (0.441 sec)

MariaDB [bank_itera]> INSERT INTO Rekening value ("145678", "SGS02", "908723", 1000000), ("256132", "KKM70", "340987", 230000), ("2471
82", "JKT48", "091298", 4000000), ("350067", "RGS02", "230987", 5500000), ("114115", "SGS01", "203010", 500000), ("100111", "KKM69", "
213671", 998500);
Query OK, 6 rows affected (0.072 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [bank_itera]> SELECT * FROM Rekening;
+-----+-----+-----+-----+
| no_rekening | Kode_cabang | Pin   | Saldo |
+-----+-----+-----+-----+
| 100111      | KKM69       | 213671 | 998500 |
| 114115      | SGS01       | 203010 | 500000 |
| 145678      | SGS02       | 908723 | 1000000 |
| 247182      | JKT48       | 091298 | 4000000 |
| 256132      | KKM70       | 340987 | 230000 |
| 350067      | RGS02       | 230987 | 5500000 |
+-----+-----+-----+-----+
6 rows in set (0.001 sec)

```

3. Masukkan data berikut pada tabel “transaksi”

Command Line : CREATE TABLE Transaksi (No_transaksi int(10), No_rekening varchar(20), Jenis_transaksi varchar(20), Tgl_transaksi varchar(30), Jumlah int(20), PRIMARY KEY (No_transaksi)) ENGINE = InnoDB;

Command Line : INSERT INTO Transaksi value (11, "100111", "ATM", "2022-06-09 08:00:45", 100000), (12, "114115", "ATM", "2022-06-09 11:10:12", 200000), (13, "114115", "DEBET", "2022-06-09 13:00:30", 50000), (14, "256132", "DEBET", "2022-06-15 11:20:00", 150000), (15, "145678", "ATM", "2022-06-18 23:11:22", 250000), (16, "145678", "DEBET", "2022-06-18 15:30:08", 300000), (17, "350067", "ATM", "2022-06-18 10:02:45", 100000);

Command Line : SELECT * FROM Transaksi;

```

XAMPP for Windows - mysql -u root -p
MariaDB [bank_itera]> CREATE TABLE Transaksi (No_transaksi int(10), No_rekening varchar(20), Jenis_transaksi varchar(20), Tgl_transaksi
i varchar(30), Jumlah int(20), PRIMARY KEY (No_transaksi)) ENGINE = InnoDB;
Query OK, 0 rows affected (2.001 sec)

MariaDB [bank_itera]> INSERT INTO Transaksi value (11, "100111", "ATM", "2022-06-09 08:00:45", 100000), (12, "114115", "ATM", "2022-06
-09 11:10:12", 200000), (13, "114115", "DEBET", "2022-06-09 13:00:30", 50000), (14, "256132", "DEBET", "2022-06-15 11:20:00", 150000),
(15, "145678", "ATM", "2022-06-18 23:11:22", 250000), (16, "145678", "DEBET", "2022-06-18 15:30:08", 300000), (17, "350067", "ATM", "
2022-06-18 10:02:45", 100000);
Query OK, 7 rows affected (0.055 sec)
Records: 7 Duplicates: 0 Warnings: 0

MariaDB [bank_itera]> SELECT * FROM Transaksi;
+-----+-----+-----+-----+-----+
| No_transaksi | No_rekening | Jenis_transaksi | Tgl_transaksi | Jumlah |
+-----+-----+-----+-----+-----+
| 11 | 100111 | ATM | 2022-06-09 08:00:45 | 100000 |
| 12 | 114115 | ATM | 2022-06-09 11:10:12 | 200000 |
| 13 | 114115 | DEBET | 2022-06-09 13:00:30 | 50000 |
| 14 | 256132 | DEBET | 2022-06-15 11:20:00 | 150000 |
| 15 | 145678 | ATM | 2022-06-18 23:11:22 | 250000 |
| 16 | 145678 | DEBET | 2022-06-18 15:30:08 | 300000 |
| 17 | 350067 | ATM | 2022-06-18 10:02:45 | 100000 |
+-----+-----+-----+-----+-----+
7 rows in set (0.001 sec)

```

Command Line : ALTER TABLE Transaksi ADD FOREIGN KEY(no_rekening) REFERENCES rekening(no_rekening) ON DELETE RESTRICT ON UPDATE CASCADE;

Command Line : DESC Transaksi;

Command Line : ALTER TABLE Nasabah ADD FOREIGN KEY(no_rekening) REFERENCES rekening(no_rekening) ON DELETE RESTRICT ON UPDATE CASCADE;

Command Line : DESC Nasabah;

```

XAMPP for Windows - mysql -u root -p
MariaDB [bank_itera]> ALTER TABLE Transaksi ADD FOREIGN KEY(no_rekening) REFERENCES rekening(no_rekening) ON DELETE RESTRICT ON UPDATE
CASCADE;
Query OK, 7 rows affected (0.822 sec)
Records: 7 Duplicates: 0 Warnings: 0

MariaDB [bank_itera]> DESC Transaksi;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| No_transaksi | int(10) | NO | PRI | NULL | |
| No_rekening | varchar(20) | YES | MUL | NULL | |
| Jenis_transaksi | varchar(20) | YES | | NULL | |
| Tgl_transaksi | varchar(30) | YES | | NULL | |
| Jumlah | int(20) | YES | | NULL | |
+-----+-----+-----+-----+-----+
5 rows in set (0.048 sec)

MariaDB [bank_itera]> ALTER TABLE Nasabah ADD FOREIGN KEY(no_rekening) REFERENCES rekening(no_rekening) ON DELETE RESTRICT ON UPDATE C
ASCADE;
Query OK, 6 rows affected (0.760 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [bank_itera]> DESC Nasabah;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| Id_Nasabah | int(10) | NO | PRI | NULL | |
| no_rekening | varchar(20) | YES | MUL | NULL | |
| Nama_Nasabah | varchar(50) | YES | | NULL | |
| Alamat | varchar(60) | YES | | NULL | |
| No_Telepon | varchar(20) | YES | | NULL | |
+-----+-----+-----+-----+-----+

```

4. Dengan menggunakan cross join, tampilkan data dari nasabah dan rekening yang sesuai (sama).

Command Line : SELECT * FROM rekening CROSS JOIN nasabah;

```
MariaDB [bank_itera]> SELECT * FROM rekening CROSS JOIN nasabah;
```

no_rekening	Kode_cabang	Pin	Saldo	Id_Nasabah	no_rekening	Nama_Nasabah	Alamat	No_Telepon
100111	KKM69	213671	998500	1	145678	Puspa Sari	Sei Batang Kuis	081376809122
114115	SGS01	203010	500000	1	145678	Puspa Sari	Sei Batang Kuis	081376809122
145678	SGS02	908723	1000000	1	145678	Puspa Sari	Sei Batang Kuis	081376809122
247182	JKT48	091298	4000000	1	145678	Puspa Sari	Sei Batang Kuis	081376809122
256132	KKM70	340987	230000	1	145678	Puspa Sari	Sei Batang Kuis	081376809122
350067	RGS02	230987	5500000	1	145678	Puspa Sari	Sei Batang Kuis	081376809122
100111	KKM69	213671	998500	2	256132	Sinta Jojo	Iskandar Muda	089987654311
114115	SGS01	203010	500000	2	256132	Sinta Jojo	Iskandar Muda	089987654311
145678	SGS02	908723	1000000	2	256132	Sinta Jojo	Iskandar Muda	089987654311
247182	JKT48	091298	4000000	2	256132	Sinta Jojo	Iskandar Muda	089987654311
256132	KKM70	340987	230000	2	256132	Sinta Jojo	Iskandar Muda	089987654311
350067	RGS02	230987	5500000	2	256132	Sinta Jojo	Iskandar Muda	089987654311
100111	KKM69	213671	998500	3	247182	Bunga Putri	Jalan Martapura	081249987011
114115	SGS01	203010	500000	3	247182	Bunga Putri	Jalan Martapura	081249987011
145678	SGS02	908723	1000000	3	247182	Bunga Putri	Jalan Martapura	081249987011
247182	JKT48	091298	4000000	3	247182	Bunga Putri	Jalan Martapura	081249987011
256132	KKM70	340987	230000	3	247182	Bunga Putri	Jalan Martapura	081249987011
350067	RGS02	230987	5500000	3	247182	Bunga Putri	Jalan Martapura	081249987011
100111	KKM69	213671	998500	4	350067	Grace Sebayang	Gang Keluarga	085276812453
114115	SGS01	203010	500000	4	350067	Grace Sebayang	Gang Keluarga	085276812453
145678	SGS02	908723	1000000	4	350067	Grace Sebayang	Gang Keluarga	085276812453
247182	JKT48	091298	4000000	4	350067	Grace Sebayang	Gang Keluarga	085276812453
256132	KKM70	340987	230000	4	350067	Grace Sebayang	Gang Keluarga	085276812453
350067	RGS02	230987	5500000	4	350067	Grace Sebayang	Gang Keluarga	085276812453

100111	KKM69	213671	998500	5	114115	Christine Talia	Sei Kambah	082266778866
114115	SGS01	203010	500000	5	114115	Christine Talia	Sei Kambah	082266778866
145678	SGS02	908723	1000000	5	114115	Christine Talia	Sei Kambah	082266778866
247182	JKT48	091298	4000000	5	114115	Christine Talia	Sei Kambah	082266778866
256132	KKM70	340987	230000	5	114115	Christine Talia	Sei Kambah	082266778866
350067	RGS02	230987	5500000	5	114115	Christine Talia	Sei Kambah	082266778866
100111	KKM69	213671	998500	6	100111	Rimayka Natalia	Perumahan Puri K	081234567890
114115	SGS01	203010	500000	6	100111	Rimayka Natalia	Perumahan Puri K	081234567890
145678	SGS02	908723	1000000	6	100111	Rimayka Natalia	Perumahan Puri K	081234567890
247182	JKT48	091298	4000000	6	100111	Rimayka Natalia	Perumahan Puri K	081234567890
256132	KKM70	340987	230000	6	100111	Rimayka Natalia	Perumahan Puri K	081234567890
350067	RGS02	230987	5500000	6	100111	Rimayka Natalia	Perumahan Puri K	081234567890

36 rows in set (0.035 sec)

5. Dengan menggunakan Inner join, tampilkan data dari rekening dan transaksi yang sesuai(sama).

Command Line : SELECT * FROM rekening INNER JOIN transaksi ON rekening.No_rekening = transaksi.No_rekening;

```
MariaDB [bank_itera]> SELECT * FROM rekening INNER JOIN transaksi ON rekening.No_rekening = transaksi.No_rekening;
```

no_rekening	Kode_cabang	Pin	Saldo	No_transaksi	no_rekening	Jenis_transaksi	Tgl_transaksi	Jumlah
100111	KKM69	213671	998500	11	100111	ATM	2022-06-09 08:00:45	100000
114115	SGS01	203010	500000	12	114115	ATM	2022-06-09 11:10:12	200000
114115	SGS01	203010	500000	13	114115	DEBET	2022-06-09 13:00:30	50000
256132	KKM70	340987	230000	14	256132	DEBET	2022-06-15 11:20:00	150000
145678	SGS02	908723	1000000	15	145678	ATM	2022-06-18 23:11:22	250000
145678	SGS02	908723	1000000	16	145678	DEBET	2022-06-18 15:30:08	300000
350067	RGS02	230987	5500000	17	350067	ATM	2022-06-18 10:02:45	100000

7 rows in set (0.096 sec)

6. Dengan menggunakan natural Join, tampilkan nasabah dan rekening.

Command Line : SELECT * FROM nasabah NATURAL JOIN rekening;

XAMPP for Windows - mysql - u root - p

MariaDB [bank_itera]> SELECT * FROM nasabah NATURAL JOIN rekening;

no_rekening	Id_Nasabah	Nama_Nasabah	Alamat	No_Telepon	Kode_cabang	Pin	Saldo
145678	1	Puspa Sari	Sei Batang Kuis	081376809122	SGS02	908723	1000000
256132	2	Sinta Jojo	Iskandar Muda	089987654311	KKM70	340987	230000
247182	3	Bunga Putri	Jalan Martapura	081249987011	JKT48	091298	4000000
350067	4	Grace Sebayang	Gang Keluarga	085276812453	RGS02	230987	5500000
114115	5	Christine Talia	Sei Kambah	082266778866	SGS01	203010	500000
100111	6	Rimayka Natalia	Perumahan Puri K	081234567890	KKM69	213671	998500

6 rows in set (0.001 sec)

7. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data rekening dan transaksi ketika jumlah transaksi > 100000.

Command Line : SELECT * FROM rekening NATURAL JOIN transaksi WHERE jumlah > 100000;

MariaDB [bank_itera]> SELECT * FROM rekening NATURAL JOIN transaksi WHERE jumlah > 100000;

no_rekening	Kode_cabang	Pin	Saldo	No_transaksi	Jenis_transaksi	Tgl_transaksi	Jumlah
114115	SGS01	203010	500000	12	ATM	2022-06-09 11:10:12	200000
256132	KKM70	340987	230000	14	DEBET	2022-06-15 11:20:00	150000
145678	SGS02	908723	1000000	15	ATM	2022-06-18 23:11:22	250000
145678	SGS02	908723	1000000	16	DEBET	2022-06-18 15:30:08	300000

4 rows in set (0.033 sec)

8. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data nasabah dan rekening ketika saldo rekening>1500000.

Command Line : SELECT * FROM nasabah NATURAL JOIN rekening WHERE saldo > 1500000;

MariaDB [bank_itera]> SELECT * FROM nasabah NATURAL JOIN rekening WHERE saldo > 1500000;

no_rekening	Id_Nasabah	Nama_Nasabah	Alamat	No_Telepon	Kode_cabang	Pin	Saldo
247182	3	Bunga Putri	Jalan Martapura	081249987011	JKT48	091298	4000000
350067	4	Grace Sebayang	Gang Keluarga	085276812453	RGS02	230987	5500000

2 rows in set (0.001 sec)

9. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data rekening dan nasabah ketika kode cabang “KKM” dan saldo<1000000.

Command Line : SELECT * FROM rekening NATURAL JOIN nasabah WHERE kode_cabang LIKE "KKM%" AND saldo < 1000000;

XAMPP for Windows - mysql - u root - p

MariaDB [bank_itera]> SELECT * FROM rekening NATURAL JOIN nasabah WHERE kode_cabang LIKE "KKM%" AND saldo < 1000000;

no_rekening	Kode_cabang	Pin	Saldo	Id_Nasabah	Nama_Nasabah	Alamat	No_Telepon
256132	KKM70	340987	230000	2	Sinta Jojo	Iskandar Muda	089987654311
100111	KKM69	213671	998500	6	Rimayka Natalia	Perumahan Puri K	081234567890

2 rows in set (0.024 sec)

10. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan nama nasabah, alamat, nomor telepon, nomor rekening dan kode cabang ketika alamat nasabah “Se”.

Command Line : SELECT nama_nasabah, alamat, no_telepon, No_rekening FROM nasabah NATURAL JOIN rekening WHERE alamat LIKE "Se%";

```
MariaDB [bank_itera]> SELECT nama_nasabah, alamat, no_telepon, No_rekening FROM nasabah NATURAL JOIN rekening WHERE alamat LIKE "Se%";
```

nama_nasabah	alamat	no_telepon	No_rekening
Puspa Sari	Sei Batang Kuis	081376809122	145678
Christine Talia	Sei Kambah	082266778866	114115

2 rows in set (0.001 sec)

11. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan nomor rekening, saldo, jumlah transaksi, tanggal transaksi ketika nomor rekening sama dengan 145678.

Command Line : SELECT No_rekening, saldo, Jumlah, Tgl_transaksi FROM rekening NATURAL JOIN transaksi WHERE No_rekening="145678";

```
MariaDB [bank_itera]> SELECT No_rekening, saldo, Jumlah, Tgl_transaksi FROM rekening NATURAL JOIN transaksi WHERE No_rekening="145678";
```

No_rekening	saldo	Jumlah	Tgl_transaksi
145678	1000000	250000	2022-06-18 23:11:22
145678	1000000	300000	2022-06-18 15:30:08

2 rows in set (0.024 sec)

12. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan data rekening dan transaksi ketika nomor rekening “1” dan jenis transaksi menggunakan “DEBET”.

Command Line : SELECT * FROM rekening NATURAL JOIN transaksi WHERE No_rekening LIKE "1%" AND Jenis_transaksi="DEBET";

```
MariaDB [bank_itera]> SELECT * FROM rekening NATURAL JOIN transaksi WHERE No_rekening LIKE "1%" AND Jenis_transaksi="DEBET";
```

no_rekening	Kode_cabang	Pin	Saldo	No_transaksi	Jenis_transaksi	Tgl_transaksi	Jumlah
114115	SGS01	203010	500000	13	DEBET	2022-06-09 13:00:30	50000
145678	SGS02	908723	1000000	16	DEBET	2022-06-18 15:30:08	300000

2 rows in set (0.082 sec)

13. Dengan menggunakan salah satu dari Jenis Join diatas, tampilkan nama nasabah, alamat, nomor telepon, nomor rekening dan kode cabang ketika nomor rekening kode cabang “SGS” dan nomor telepon “0812”.

Command Line : SELECT nama_nasabah, alamat, no_telepon, No_rekening, Kode_cabang FROM rekening NATURAL JOIN nasabah WHERE Kode_cabang LIKE "SGS%" OR no_telepon LIKE "0812%";

```
MariaDB [bank_itera]> SELECT nama_nasabah, alamat, no_telepon, No_rekening, Kode_cabang FROM rekening NATURAL JOIN nasabah WHERE Kode_cabang LIKE "SGS%" OR no_telepon LIKE "0812%";
```

nama_nasabah	alamat	no_telepon	No_rekening	Kode_cabang
Puspa Sari	Sei Batang Kuis	081376809122	145678	SGS02
Bunga Putri	Jalan Martapura	081249987011	247182	JKT48
Christine Talia	Sei Kambah	082266778866	114115	SGS01
Rimayka Natalia	Perumahan Puri K	081234567890	100111	KKM69

4 rows in set (0.001 sec)

BAB III

KESIMPULAN

Kesimpulannya, *Relationship* adalah suatu hubungan antara beberapa entitas. *Join* merupakan salah satu konstruksi dasar dari SQL dan basis data seperti Cross Join, Natural Join, Inner Join dan Outer Join.

DAFTAR PUSTAKA

Modul 4 Praktikum Basis Data

