



BASIS DATA

SELECT JOIN

TIM AJAR BASIS DATA JTI-POLINEMA

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.
- Di dalam sebuah relationship, primary key memiliki peran penting untuk mengaitkan entitas. Selain itu, primary key juga digunakan untuk mendefinisikan batasan keterhubungan.

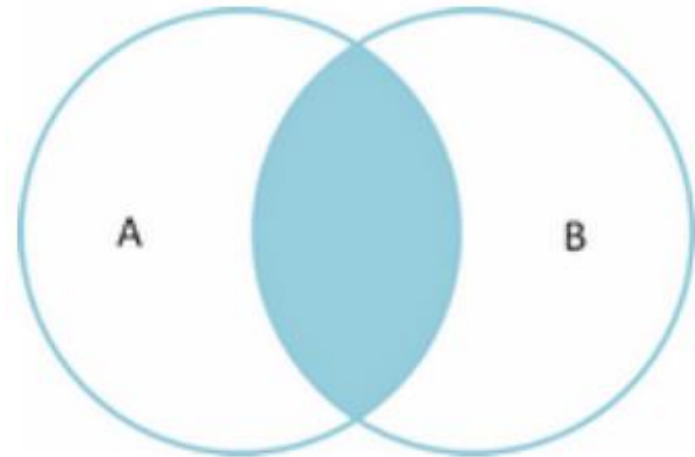
JOIN



- **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 table.
- Join dapat diklasifikasikan ke dalam dua jenis: **inner** dan **outer**.

INNER JOIN

- **Inner join** pada dasarnya adalah menemukan persimpangan (intersection) antara dua buah tabel.
- **Cara 1**
SELECT A1, A2, ..., An
FROM r1
INNER JOIN r2
ON r1.join_key = r2.join_key
- **Cara 2**
SELECT A1, A2, ..., An
FROM r1, r2
WHERE r1.key = r2.key

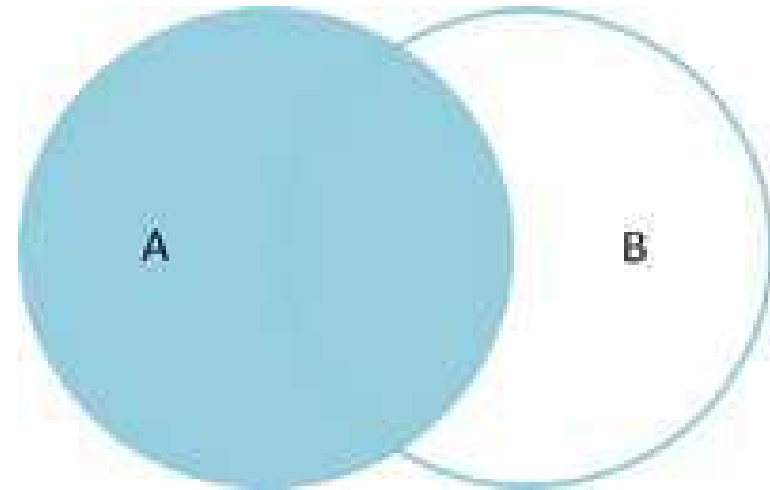


OUTER JOIN

Left outer join (atau left join) mengembalikan semua nilai dari tabel kiri ditambah dengan nilai dari tabel kanan yang sesuai (atau NULL jika tidak ada nilai yang sesuai).

Syntax

```
SELECT A1, A2, ..., An  
FROM r1  
LEFT OUTER JOIN r2  
ON r1.join_key = r2.join_key
```





LEFT JOIN

- informasi yang ditampilkan adalah semua data dari tabel sebelah kiri statement left join beserta pasangannya dari tabel sebelah kanan. Meskipun ada data dari sebelah kiri tidak memiliki pasangan, tetap akan ditampilkan dengan pasangannya berupa NULL
- Contoh
SELECT M.NIM, M.nama,
K.IdKelas, K.Nilai
FROM Mahasiswa M left join KRS K
On M.NIM = K.NIM
WHERE Nama like '%a%'

OUTER JOIN

Right outer join (atau right join) pada dasarnya sama seperti left join, namun dalam bentuk terbalik—kanan dan kiri.

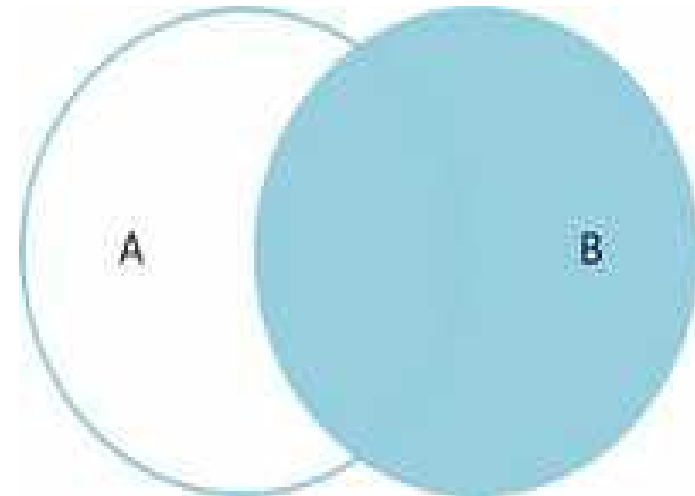
- **syntax**

```
SELECT A1, A2, ..., An
```

```
FROM r1
```

```
RIGHT OUTER JOIN r2
```

```
ON r1.join_key = r2.join_key
```





RIGHT JOIN

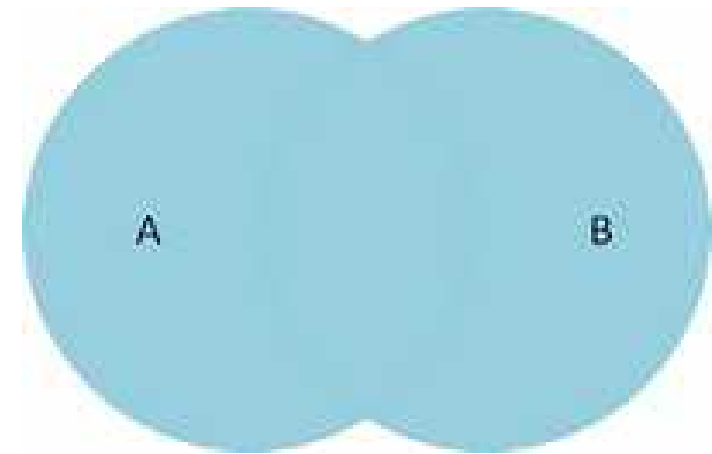
- informasi yang ditampilkan adalah semua data dari tabel sebelah kanan statement right join beserta pasangannya dari tabel sebelah kiri. Meskipun ada data dari sebelah kanan statement right join tidak memiliki pasangan, data tersebut tetap ditampilkan dengan diberi pasangan NULL

- Contoh

```
SELECT M.NIM, M.nama,  
       K.IdKelas, K.Nilai  
FROM Mahasiswa M Right join KRS K  
On M.NIM = K.NIM  
WHERE Nama like '%a%'
```


OUTER JOIN

- **Full outer join (atau full join)** pada hakekatnya merupakan kombinasi dari left dan right join.
- **syntax**
SELECT A1, A2, ..., An
FROM r1
FULL OUTER JOIN r2
ON r1.join_key = r2.join_key





FULL JOIN

- informasi yang ditampilkan adalah dari kedua tabel baik kiri maupun kanan statement full join dengan pasangannya. Bagi data yang tidak memiliki pasangan akan diberi pasangan dengan NULL

- Contoh

```
SELECT M.NIM, M.nama,  
       K.IdKelas, K.Nilai  
FROM Mahasiswa M Full join KRS K  
On M.NIM = K.NIM  
WHERE Nama like '%a%'
```



SYNTAX

SELECT

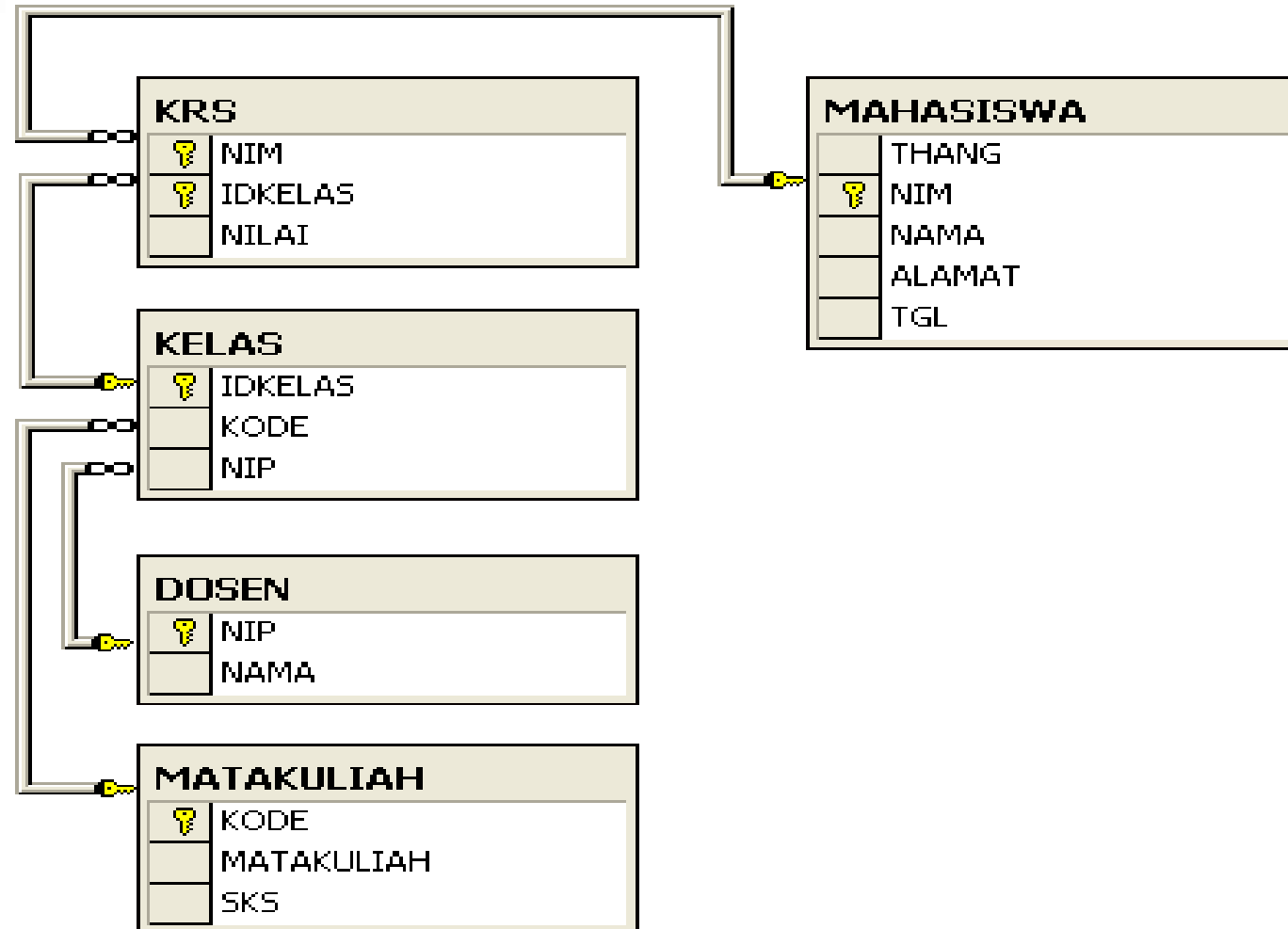
Tabel1.Kolom1, Tabel1.Kolom2, ..., Tabel1.Kolomn,
Tabel2.Kolom1, Tabel2.Kolom2, ..., Tabel2.Kolomn,
...

Tabeln.Kolom1, Tabeln.Kolom2, ..., Tabeln.Kolomn

FROM Tabel1, Tabel2,..., Tabeln

[WHERE kondisi];

RELASI ANTAR TABEL





PENGGUNAAN ALIAS

```
SELECT alias1.kolom, alias2.kolom  
FROM tabel1 alias1, tabel2 alias2  
WHERE kondisi
```

Contoh:

```
SELECT M.NIM, M.nama,  
       K.IdKelas, K.Nilai  
FROM Mahasiswa M, KRS K
```



CONTOH SYNTAX

1. JOIN atau INNER JOIN

Menggabungkan dua tabel dimana diantara dua tabel datanya bersesuaian.

2. LEFT JOIN atau LEFT OUTER JOIN

Menggabungkan dua tabel dimana diantara dua tabel datanya bersesuaian dan juga semua record pada tabel sebelah kiri.

3. RIGHT JOIN atau RIGHT OUTER JOIN

Menggabungkan dua tabel dimana diantara dua tabel datanya bersesuaian dan juga semua record pada tabel sebelah kanan.



```
SELECT Nilai.NIM, NAMA_MHS, KD_MK, MID  
FROM Nilai INNER JOIN Mahasiswa  
ON Nilai.NIM = Mahasiswa.NIM
```

Hasil :

NIM	NAMA_MHS	KD_MK	MID
10296832	Nurhayati	KK021	60
10296126	Astuti	KD132	70
31296500	Budi	KK021	55
41296525	Prananigrum	KU122	90
21196353	Quraish	KU122	75
50095487	Pipit	KD132	80



```
SELECT Mahasiswa.NIM, NAMA_MHS, KD_MK, MID  
FROM Mahasiswa LEFT OUTER JOIN Nilai  
ON Nilai.NIM = Mahasiswa.NIM
```

Hasil :

NIM	NAMA_MHS	KD_MK	MID
10296832	Nurhayati	KK021	60
10296126	Astuti	KD132	70
31296500	Budi	KK021	55
41296525	Prananigrum	KU122	90
21196353	Quraish	KU122	75
50095487	Pipit	KD132	80
10296001	Fintri	-	-
21198002	Julizar	-	-



```
SELECT Mahasiswa.NIM, NAMA_MHS, KD_MK, MID
FROM Nilai RIGHT OUTER JOIN Mahasiswa
ON Nilai.NIM = Mahasiswa.NIM
```

Hasil :

NIM	NAMA_MHS	KD_MK	MID
10296832	Nurhayati	KK021	60
10296126	Astuti	KD132	70
31296500	Budi	KK021	55
41296525	Prananigrum	KU122	90
21196353	Quraish	KU122	75
50095487	Pipit	KD132	80
10296001	Fintri	-	-
21198002	Julizar	-	-

NO_DEP	NAMA	LOKASI
1	KEUANGAN	MALANG
2	PEMASARAN	MALANG
3	PRODUKSI	MALANG
4	TENAGA KERJA	MALANG
5	IT	MALANG

Gambar 1.1 departemen

NO_PRO	NAMA	LOKASI	NO_DEP
1	DESAIN GAMBAR	MALANG	5
2	PEMASARAN BARANG	MALANG	3
3	JARINGAN KOMPUTER	SURABAYA	4

Gambar 1.2 proyek

NO_KER	NO_KTP	NO_PRO	LAMA_JAM
1	20104260001	1	36
2	20104340002	1	36
3	20104290001	2	60
4	20104320002	2	60
5	20104270002	3	42

Gambar 1.3 kerja

TUGAS



NO_KTP	NAMA	JENIS_KEL	ALAMAT	TGL_LAHIR	GAJI	NO_DEP
20104260001	DODO	1	MALANG	12-DEC-87	2000000	1
20104270002	NINI	2	SURABAYA	13-DEC-89	2000000	2
20104280002	NORMAN	1	MALANG	01-JAN-79	2500000	2
20104290001	LALA	2	MALANG	11-MAR-90	1750000	5
20104300002	DINDA	2	KEDIRI	15-FEB-80	2000000	3
20104310001	SONI	1	TULUNGAGUNG	29-OCT-70	3000000	4
20104320002	NONO	1	SURABAYA	17-SEP-86	2600000	5
20104330001	DITA	2	MALANG	01-MAR-74	2000000	3
20104340002	HARI	1	SURABAYA	01-NOV-69	2900000	5
20104350002	NINO	1	MALANG	10-NOV-72	2000000	1

Gambar 1.4 pegawai

NO_TANG	NO_KTP	NAMA	JENIS_KEL	TGL_LAHIR	HUBUNGAN
1	20104350002	HERA	2	12-NOV-72	ISTRI
2	20104350002	RAHEL	2	17-AUG-92	ANAK
3	20104350002	ROKI	1	12-NOV-98	ANAK
4	20104310001	SHINTA	2	20-MAY-72	ISTRI
5	20104290001	DINI	2	01-JUL-90	ANAK

Gambar 1.5 tanggungan

TUGAS



1. Buat query dengan menggunakan konsep JOIN untuk menyelesaikan soal dan menghasilkan output berikut : [**Database Perusahaan**]

- a) Tampilkan nama departemen yang mempunyai pegawai termuda, sertakan nama pegawai dan tanggal lahirnya.

DEPARTEMEN	PEGAWAI	TGL_LAHIR
IT	LALA	11-03-1990

- b) Tampilkan nama departemen dan jumlah pegawai yang bekerja pada masing-masing departemen tersebut.



NAMA	JML_PEGAWAI
IT	3
KELANGAN	2
PRODUKSI	2
PEMASARAN	2
TENAGA KERJA	1

- c) Tampilkan semua informasi mengenai proyek dan di departemen mana proyek tersebut dikerjakan.

NO_PRO	NAMA	LOKASI	NO_DEP	NO_DEP	NAMA	LOKASI
1	DESAIN GAMBAR	MALANG	5	5	IT	MALANG
2	PEMASARAN BARANG	MALANG	3	3	PRODUKSI	MALANG
3	JARINGAN KOMPUTER	SURABAYA	4	4	TENAGA KERJA	MALANG



TERIMAKASIH

REFERENSI



- Dwi Puspitasari, S.Kom, "**Buku Ajar Dasar Basis Data**", *Program Studi Manajemen Informatika Politeknik Negeri Malang*, 2012.
- Fathansyah, "**Basisdata Revisi Kedua**", Bandung: Informatika, 2015.