

Modul Praktikum Basis Data

Joins



**TIM ASISTEN BASIS DATA
ILMU KOMPUTER UPI 2014/2015**



JOIN?

Join merupakan perintah untuk menggabungkan dua atau beberapa table dengan syarat salah satu field dari masing-masing table memiliki isi field yang sama sebagai relasi. Apakah yang tidak memiliki relasi bisa dijoinkan? Temukan jawabannya nanti. Perintah **Join** dimulai dengan "SELECT" untuk memilih data field apa aja yang akan ditampilkan. Ada beberapa jenis **Join**, yaitu *Inner Join*, *Left Join*, dan *Right Join*. Sebelum dimulai penjelasannya, ada beberapa table yang harus dibuat. Diantaranya:

Table mhs

Field	Type	Null	Key	Default	Extra
nim	varchar(10)	NO	PRI	NULL	
nama	varchar(25)	NO		NULL	
program_studi	varchar(30)	NO		NULL	

Table dosen

Field	Type	Null	Key	Default	Extra
kode_dosen	varchar(10)	NO	PRI	NULL	
nama_dosen	varchar(30)	NO		NULL	

Table matkul

Field	Type	Null	Key	Default	Extra
kode_matkul	varchar(10)	NO	PRI	NULL	
nama_matkul	varchar(50)	NO		NULL	
sks	int(11)	NO		NULL	
semester	int(11)	NO		NULL	

Table nilai

Field	Type	Null	Key	Default	Extra
kode_nilai	int(11)	NO	PRI	NULL	auto_increment
nim	varchar(10)	NO	MUL	NULL	
kode_dosen	varchar(10)	NO	MUL	NULL	
kode_matkul	varchar(10)	NO	MUL	NULL	
nilai	varchar(2)	YES		NULL	

Setelah semua table dibuat, isilah semua table tersebut dengan data-data berikut:

Table mhs

nim	nama	program_studi
1203141	Ahmad	Ilmu Komputer
1203142	Maulana	Pendidikan Ilmu Komputer
1203143	Aksan	Pendidikan Ilmu Komputer
1203144	Maulana	Ilmu Komputer
1203145	Iip	Ilmu Komputer

Table dosen

kode_dosen	nama_dosen
NP-001	Cucu
NP-002	Cici
NP-003	Caca
NP-004	Danang
NP-005	Somad
NP-006	Iis
NP-007	Wahyu

Table matkul

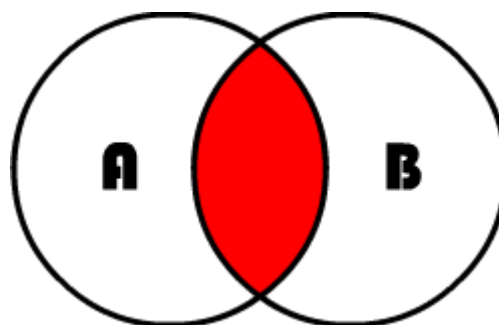
kode_matkul	nama_matkul	sks	semester
M-001	Kalkulus	3	2
M-002	Landasan Pendidikan	2	2
M-003	Algoritma dan Pemrograman I	2	2
M-004	Sistem Digital	2	2
M-005	Basis Data	3	2
M-008	Pengantar Teknologi Informasi	3	1
M-009	Konsep Teknologi	3	2

Table nilai

kode_nilai	nim	kode_dosen	kode_matkul	nilai
1	1203141	NP-001	M-001	A
2	1203141	NP-004	M-004	A
3	1203141	NP-005	M-005	B
4	1203142	NP-002	M-002	A
5	1203142	NP-003	M-003	A
6	1203143	NP-002	M-002	A
7	1203143	NP-003	M-003	B
8	1203144	NP-001	M-001	A
9	1203144	NP-004	M-004	A
10	1203144	NP-005	M-005	B
11	1203145	NP-001	M-001	A
12	1203145	NP-004	M-004	A
13	1203145	NP-005	M-005	A

Inner Join

Inner Join menggabungkan dua table yang hanya memiliki irisan. Dengan kata lain jika tidak ada relasi, maka data tersebut akan tampil dengan data yang ditampilkan tidak memiliki tujuan yang jelas. Ilustrasi gambar sebagai berikut:



Sintaks:

```
SELECT nama_field FROM nama_table1 INNER JOIN nama_table2 ON  
nama_table1.nama_field=nama_table2.nama_field;
```

Bisa juga tidak menggunakan kata inner, sehingga querynya menjadi:

```
SELECT nama_field FROM nama_table1 JOIN nama_table2 ON  
nama_table1.nama_field=nama_table2.nama_field;
```

Contoh:

```
SELECT nilai.kode_nilai, dosen.kode_dosen, dosen.nama_dosen, nilai.nim,  
nilai.kode_matkul, nilai.nilai FROM dosen INNER JOIN nilai ON  
dosen.kode_dosen=nilai.kode_dosen ORDER BY nilai.kode_nilai;
```

Atau

```
SELECT nilai.kode_nilai, dosen.kode_dosen, dosen.nama_dosen, nilai.nim,  
nilai.kode_matkul, nilai.nilai FROM dosen JOIN nilai ON  
dosen.kode_dosen=nilai.kode_dosen ORDER BY nilai.kode_nilai;
```

kode_nilai	kode_dosen	nama_dosen	nim	kode_matkul	nilai
1	NP-001	Cucu	1203141	M-001	A
2	NP-004	Danang	1203141	M-004	A
3	NP-005	Somad	1203141	M-005	B
4	NP-002	Cici	1203142	M-002	A
5	NP-003	Caca	1203142	M-003	A
6	NP-002	Cici	1203143	M-002	A
7	NP-003	Caca	1203143	M-003	B
8	NP-001	Cucu	1203144	M-001	A
9	NP-004	Danang	1203144	M-004	A
10	NP-005	Somad	1203144	M-005	B
11	NP-001	Cucu	1203145	M-001	A
12	NP-004	Danang	1203145	M-004	A
13	NP-005	Somad	1203145	M-005	A

Jika diantara 2 table tersebut tidak memiliki irisan, maka tidak ada yang ditampilkan.

```
SELECT nilai.kode_nilai, dosen.kode_dosen, dosen.nama_dosen, nilai.nim,  
nilai.kode_matkul, nilai.nilai FROM dosen INNER JOIN nilai ON  
dosen.kode_dosen=nilai.kode_matkul ORDER BY nilai.kode_nilai;
```

Empty set (0.00 sec)

Atau

```
SELECT nilai.kode_nilai, dosen.kode_dosen, dosen.nama_dosen, nilai.nim,  
nilai.kode_matkul, nilai.nilai FROM dosen JOIN nilai;
```

kode_nilai	kode_dosen	nama_dosen	nim	kode_matkul	nilai
1	NP-001	Cucu	1203141	M-001	A
1	NP-002	Cici	1203141	M-001	A
1	NP-003	Caca	1203141	M-001	A
1	NP-004	Danang	1203141	M-001	A
1	NP-005	Somad	1203141	M-001	A
1	NP-006	Iis	1203141	M-001	A
1	NP-007	Wahyu	1203141	M-001	A
2	NP-001	Cucu	1203141	M-004	A
2	NP-002	Cici	1203141	M-004	A
2	NP-003	Caca	1203141	M-004	A
2	NP-004	Danang	1203141	M-004	A
2	NP-005	Somad	1203141	M-004	A
2	NP-006	Iis	1203141	M-004	A
2	NP-007	Wahyu	1203141	M-004	A
3	NP-001	Cucu	1203141	M-005	B
3	NP-002	Cici	1203141	M-005	B
3	NP-003	Caca	1203141	M-005	B
3	NP-004	Danang	1203141	M-005	B
3	NP-005	Somad	1203141	M-005	B
3	NP-006	Iis	1203141	M-005	B
3	NP-007	Wahyu	1203141	M-005	B

...

Apakah data yang ditampilkan memiliki tujuan yang jelas? -_-?

LATIHAN PERTAMA!

Coba tampilkan nama dari table mhs, nama_dosen dari table dosen dan nilai dari table nilai dengan menggunakan **INNER JOIN**. Apakah bisa menggabungkan 3 table seperti ini? Jangan dulu discroll kebawah ya! (-_-)

Setelah dicoba, coba samakan dengan yang dibawah ini (^_^):

```
SELECT mhs.nama, dosen.nama_dosen, nilai.nilai FROM mhs INNER JOIN nilai  
INNER JOIN dosen on mhs.nim=nilai.nim AND  
dosen.kode_dosen=nilai.kode_dosen;
```

nama	nama_dosen	nilai
Ahmad	Cucu	A
Ahmad	Danang	A
Ahmad	Somad	B
Maulana	Cici	A
Maulana	Caca	A
Aksan	Cici	A
Aksan	Caca	B
Maulana	Cucu	A
Maulana	Danang	A
Maulana	Somad	B
Iip	Cucu	A
Iip	Danang	A
Iip	Somad	A

LATIHAN KEDUA!

Itu jika 3 table, kalau 4 table? Coba tampilkan nama dari table mhs, nama_dosen dari table dosen, nama_matkul dari table matkul, dan nilai dari table nilai dengan menggunakan inner join. Apakah bisa? (>_<) Jangan dulu discroll kebawah ya!

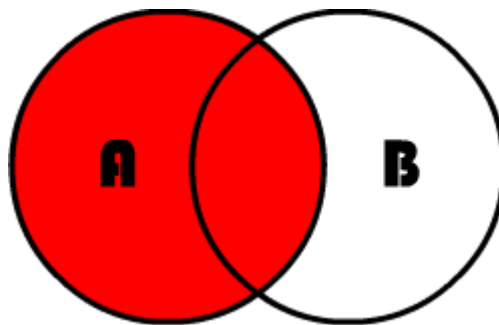
Sintaks:

```
SELECT mhs.nama, dosen.nama_dosen, matkul.nama_matkul, nilai.nilai FROM
mhs INNER JOIN nilai INNER JOIN dosen INNER JOIN matkul ON
mhs.nim=nilai.nim AND dosen.kode_dosen=nilai.kode_dosen AND
matkul.kode_matkul=nilai.kode_matkul;
```

nama	nama_dosen	nama_matkul	nilai
Ahmad	Cucu	Kalkulus	A
Ahmad	Danang	Sistem Digital	A
Ahmad	Somad	Basis Data	B
Maulana	Cici	Landasan Pendidikan	A
Maulana	Caca	Algoritma dan Pemrograman I	A
Aksan	Cici	Landasan Pendidikan	A
Aksan	Caca	Algoritma dan Pemrograman I	B
Maulana	Cucu	Kalkulus	A
Maulana	Danang	Sistem Digital	A
Maulana	Somad	Basis Data	B
Iip	Cucu	Kalkulus	A
Iip	Danang	Sistem Digital	A
Iip	Somad	Basis Data	A

Left Join

Query ini akan menampilkan semua record dari table kiri walaupun dari table kanan ada yang tidak memiliki pasangan data di table kiri. Ilustrasi gambar sebagai berikut:



Sintaks:

```
SELECT nama_field FROM nama_table1 LEFT JOIN nama_table2 ON
nama_table1.nama_field=nama_table2.nama_field;
```

Contoh:

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,
nilai.kode_dosen, nilai.nilai FROM matkul LEFT JOIN nilai ON
matkul.kode_matkul=nilai.kode_matkul ORDER BY nilai.kode_nilai;
```

kode_nilai	kode_matkul	nama_matkul	nim	kode_dosen	nilai
NULL	M-008	Pengantar Teknologi Informasi	NULL	NULL	NULL
NULL	M-009	Konsep Teknologi	NULL	NULL	NULL
1	M-001	Kalkulus	1203141	NP-001	A
2	M-004	Sistem Digital	1203141	NP-004	A
3	M-005	Basis Data	1203141	NP-005	B
4	M-002	Landasan Pendidikan	1203142	NP-002	A
5	M-003	Algoritma dan Pemrograman I	1203142	NP-003	A
6	M-002	Landasan Pendidikan	1203143	NP-002	A
7	M-003	Algoritma dan Pemrograman I	1203143	NP-003	B
8	M-001	Kalkulus	1203144	NP-001	A
9	M-004	Sistem Digital	1203144	NP-004	A
10	M-005	Basis Data	1203144	NP-005	B
11	M-001	Kalkulus	1203145	NP-001	A
12	M-004	Sistem Digital	1203145	NP-004	A
13	M-005	Basis Data	1203145	NP-005	A

Apabila table kanan tidak memiliki irisan dengan table kiri, maka table kanan yang tidak memiliki irisan akan bernilai NULL dan semua data dari table kiri akan muncul:

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,
nilai.kode_dosen, nilai.nilai FROM matkul LEFT JOIN nilai ON
matkul.kode_matkul=nilai.kode_dosen ORDER BY nilai.kode_nilai;
```

kode_nilai	kode_matkul	nama_matkul	nim	kode_dosen	nilai
NULL	M-003	Algoritma dan Pemrograman I	NULL	NULL	NULL
NULL	M-008	Pengantar Teknologi Informasi	NULL	NULL	NULL
NULL	M-001	Kalkulus	NULL	NULL	NULL
NULL	M-004	Sistem Digital	NULL	NULL	NULL
NULL	M-009	Konsep Teknologi	NULL	NULL	NULL
NULL	M-002	Landasan Pendidikan	NULL	NULL	NULL
NULL	M-005	Basis Data	NULL	NULL	NULL

Selain itu fungsi dari left join juga bisa menampilkan data dimana dari table kanannya NULL, yang artinya tidak memiliki pasangan di table kiri. Begitupun sebaliknya. Jika ingin menampilkan data table kiri dengan syarat table kanan NULL, maka sintaksnya sebagai berikut:

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,
nilai.kode_dosen, nilai.nilai FROM matkul LEFT JOIN nilai ON
matkul.kode_matkul=nilai.kode_matkul WHERE nilai.kode_matkul IS NULL
ORDER BY nilai.kode_nilai;
```

kode_nilai	kode_matkul	nama_matkul	nim	kode_dosen	nilai
NULL	M-008	Pengantar Teknologi Informasi	NULL	NULL	NULL
NULL	M-009	Konsep Teknologi	NULL	NULL	NULL

Tapi...

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,
nilai.kode_dosen, nilai.nilai FROM matkul LEFT JOIN nilai ON
matkul.kode_matkul=nilai.kode_matkul WHERE matkul.kode_matkul IS NULL
ORDER BY nilai.kode_nilai;
```

Mengapa? -_-'

Empty set (0.00 sec)

Ada yang mau jawab?

Dari percobaan pertama kenapa menampilkan dari table nilai yang NULL? Karena yang disebelah kiri adalah table matkul maka otomatis dari table nilai yang tidak memiliki irisan akan ditampilkan dengan statements NULL. Dari percobaan 2, apakah ada kode_matkul dari table matkul yang NULL? Tidak ada kan? Maka empty set. Semoga mengerti!

(^_^)

/|\

|

/ \

Dan jika ingin menampilkan data table kiri dengan syarat table kanan NOT NULL, maka sintaksnya sebagai berikut:

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,
nilai.kode_dosen, nilai.nilai FROM matkul LEFT JOIN nilai ON
matkul.kode_matkul=nilai.kode_matkul WHERE nilai.kode_matkul IS NOT NULL
ORDER BY nilai.kode_nilai;
```

kode_nilai	kode_matkul	nama_matkul	nim	kode_dosen	nilai
1	M-001	Kalkulus	1203141	NP-001	A
2	M-004	Sistem Digital	1203141	NP-004	A
3	M-005	Basis Data	1203141	NP-005	B
4	M-002	Landasan Pendidikan	1203142	NP-002	A
5	M-003	Algoritma dan Pemrograman I	1203142	NP-003	A
6	M-002	Landasan Pendidikan	1203143	NP-002	A
7	M-003	Algoritma dan Pemrograman I	1203143	NP-003	B
8	M-001	Kalkulus	1203144	NP-001	A
9	M-004	Sistem Digital	1203144	NP-004	A
10	M-005	Basis Data	1203144	NP-005	B
11	M-001	Kalkulus	1203145	NP-001	A
12	M-004	Sistem Digital	1203145	NP-004	A
13	M-005	Basis Data	1203145	NP-005	A

Mengapa? Tapi ketika sintaksnya:

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,
nilai.kode_dosen, nilai.nilai FROM matkul LEFT JOIN nilai ON
matkul.kode_matkul=nilai.kode_matkul WHERE matkul.kode_matkul IS NOT NULL
ORDER BY nilai.kode_nilai;
```

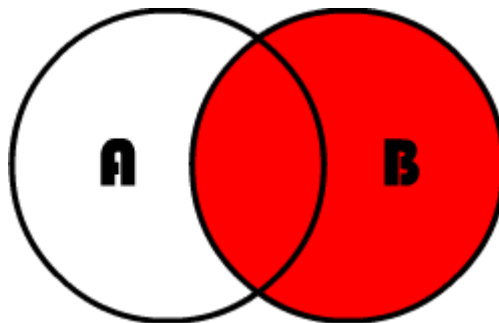
kode_nilai	kode_matkul	nama_matkul	nim	kode_dosen	nilai
NULL	M-008	Pengantar Teknologi Informasi	NULL	NULL	NULL
NULL	M-009	Konsep Teknologi	NULL	NULL	NULL
1	M-001	Kalkulus	1203141	NP-001	A
2	M-004	Sistem Digital	1203141	NP-004	A
3	M-005	Basis Data	1203141	NP-005	B
4	M-002	Landasan Pendidikan	1203142	NP-002	A
5	M-003	Algoritma dan Pemrograman I	1203142	NP-003	A
6	M-002	Landasan Pendidikan	1203143	NP-002	A
7	M-003	Algoritma dan Pemrograman I	1203143	NP-003	B
8	M-001	Kalkulus	1203144	NP-001	A
9	M-004	Sistem Digital	1203144	NP-004	A
10	M-005	Basis Data	1203144	NP-005	B
11	M-001	Kalkulus	1203145	NP-001	A
12	M-004	Sistem Digital	1203145	NP-004	A
13	M-005	Basis Data	1203145	NP-005	A

Tau kan alasannya kenapa? Sama aja dengan yang sebelumnya. ^_^

Dari sintaks-sintaks Left Join barusan, apa yang dapat disimpulkan dengan Inner Join? Silahkan analisis bareng-bareng dan jawab pertanyaannya setelah Right Join.

Right Join

Query ini merupakan kebalikan dari Left Join. Dalam Left Join table kanan yang tidak memiliki irisan dengan table kiri, akan bernilai NULL dengan semua data dari table kiri akan ditampilkan. Sedangkan dalam Right Join table kiri yang tidak memiliki irisan dengan table kanan, akan bernilai NULL dengan semua data dari table kanan akan ditampilkan. Ilustrasi gambar sebagai berikut:



Sintaks:

```
SELECT nama_field FROM nama_table1 RIGHT JOIN nama_table2 ON
nama_table1.nama_field=nama_table2.nama_field;
```

Contoh:

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,
nilai.kode_dosen, nilai.nilai FROM nilai RIGHT JOIN matkul ON
matkul.kode_matkul=nilai.kode_matkul ORDER BY nilai.kode_nilai;
```

kode_nilai	kode_matkul	nama_matkul	nim	kode_dosen	nilai
NULL	M-008	Pengantar Teknologi Informasi	NULL	NULL	NULL
NULL	M-009	Konsep Teknologi	NULL	NULL	NULL
1	M-001	Kalkulus	1203141	NP-001	A
2	M-004	Sistem Digital	1203141	NP-004	A
3	M-005	Basis Data	1203141	NP-005	B
4	M-002	Landasan Pendidikan	1203142	NP-002	A
5	M-003	Algoritma dan Pemrograman I	1203142	NP-003	A
6	M-002	Landasan Pendidikan	1203143	NP-002	A
7	M-003	Algoritma dan Pemrograman I	1203143	NP-003	B
8	M-001	Kalkulus	1203144	NP-001	A
9	M-004	Sistem Digital	1203144	NP-004	A
10	M-005	Basis Data	1203144	NP-005	B
11	M-001	Kalkulus	1203145	NP-001	A
12	M-004	Sistem Digital	1203145	NP-004	A
13	M-005	Basis Data	1203145	NP-005	A

Apabila table kiri tidak memiliki irisan dengan table kanan, maka table kiri yang tidak memiliki irisan akan bernilai NULL dan semua data dari table kanan akan muncul. Sintaksnya sebagai berikut:

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,
nilai.kode_dosen, nilai.nilai FROM nilai RIGHT JOIN matkul ON
matkul.kode_matkul=nilai.nim ORDER BY nilai.kode_nilai;
```

kode_nilai	kode_matkul	nama_matkul	nim	kode_dosen	nilai
NULL	M-003	Algoritma dan Pemrograman I	NULL	NULL	NULL
NULL	M-008	Pengantar Teknologi Informasi	NULL	NULL	NULL
NULL	M-001	Kalkulus	NULL	NULL	NULL
NULL	M-004	Sistem Digital	NULL	NULL	NULL
NULL	M-009	Konsep Teknologi	NULL	NULL	NULL
NULL	M-002	Landasan Pendidikan	NULL	NULL	NULL
NULL	M-005	Basis Data	NULL	NULL	NULL

Right Join juga memiliki fungsi yang sama dengan Left Join, dimana kamu bisa menampilkan data table kanan dengan syarat table kiri NULL. Sintaksnya sebagai berikut:

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,
nilai.kode_dosen, nilai.nilai FROM nilai RIGHT JOIN matkul ON
matkul.kode_matkul=nilai.kode_matkul WHERE nilai.kode_matkul IS NULL
ORDER BY nilai.kode_nilai;
```

kode_nilai	kode_matkul	nama_matkul	nim	kode_dosen	nilai
NULL	M-008	Pengantar Teknologi Informasi	NULL	NULL	NULL
NULL	M-009	Konsep Teknologi	NULL	NULL	NULL

Jika kamu ingin menampilkan data table kanan dengan syarat data table kiri NOT NULL. Sintaksnya sebagai berikut:

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,
nilai.kode_dosen, nilai.nilai FROM nilai RIGHT JOIN matkul ON
matkul.kode_matkul=nilai.kode_matkul WHERE nilai.kode_matkul IS NOT NULL
ORDER BY nilai.kode_nilai;
```

kode_nilai	kode_matkul	nama_matkul	nim	kode_dosen	nilai
1	M-001	Kalkulus	1203141	NP-001	A
2	M-004	Sistem Digital	1203141	NP-004	A
3	M-005	Basis Data	1203141	NP-005	B
4	M-002	Landasan Pendidikan	1203142	NP-002	A
5	M-003	Algoritma dan Pemrograman I	1203142	NP-003	A
6	M-002	Landasan Pendidikan	1203143	NP-002	A
7	M-003	Algoritma dan Pemrograman I	1203143	NP-003	B
8	M-001	Kalkulus	1203144	NP-001	A
9	M-004	Sistem Digital	1203144	NP-004	A
10	M-005	Basis Data	1203144	NP-005	B
11	M-001	Kalkulus	1203145	NP-001	A
12	M-004	Sistem Digital	1203145	NP-004	A
13	M-005	Basis Data	1203145	NP-005	A

Sekarang saya mau bertanya, kenapa sintaks:

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,  
nilai.kode_dosen, nilai.nilai FROM matkul RIGHT JOIN nilai ON  
matkul.kode_matkul=nilai.kode_matkul ORDER BY nilai.kode_nilai;
```

Sama dengan

```
SELECT nilai.kode_nilai, matkul.kode_matkul, matkul.nama_matkul, nilai.nim,  
nilai.kode_dosen, nilai.nilai FROM nilai RIGHT JOIN matkul ON  
matkul.kode_matkul=nilai.kode_matkul WHERE nilai.kode_matkul IS NOT NULL  
ORDER BY nilai.kode_nilai;
```

(^_^)

/|\

|

/ \

Ayo jelaskan kenapa sebelum muncul pertanyaan dibawah ini!

Nah, sekarang silahkan bagi teman-teman menjawab kesimpulan dari Inner Join, Left Join, dan Right Join sebelum dipaparkan! Kemudian ada kemiripan antara Inner Join, Left Join, dan Right Join, apa itu? Sehingga outputnya sama saja.

Kesimpulan : Inner Join hasilnya akan sama dengan Left Join yang table kanannya NOT NULL dan Right Join yang table kirinya NOT NULL. Mengapa demikian? Mari Berimajinasi! Left Join dengan table kanan yang NOT NULL berarti akan menampilkan yang hanya memiliki irisan. Benar? Dan Right Join dengan table kiri NOT NULL berarti akan menampilkan yang hanya memiliki irisan juga. Inner Join juga menampilkan yang hanya memiliki irisan diantara table. Dengan kata lain Left Join dengan table kanan NOT NULL dan Right Join dengan table kiri yang NOT NULL akan sama dengan Inner Join.

NB : Bila ingin menampilkan semua data yang ada di tabel sebelah kiri baik yang mempunyai pasangan ataupun yang tidak mempunyai pasangan dengan data pada tabel di sebelah kanan, maka gunakan left join. Sedangkan bila yang terjadi adalah kebalikannya, yaitu ingin menampilkan semua data yang ada di sebelah kanan baik yang mempunyai pasangan ataupun yang tidak mempunyai pasangan dengan data pada tabel di sebelah kiri, maka gunakan right join.

“Jika ilmu tidak bertambah, maka emosi yang bertambah. Jika iman tidak bertambah, maka nafsu yang bertambah”

Referensi

Modul #4 Praktikum Basis Data [Agregasi, Grouping, Join] 2013