

PENGAMBILAN DATA DARI MULTIPLE TABEL

Gde Brahupadhya Subiksa, S.Kom

TIPE JOIN

JOIN adalah Salah satu fitur SQL yang paling berguna untuk menggabungkan table dengan query – query yang mendapatkan kembali data. Join adalah salah satu dari beberapa operasi paling penting yang dapat anda lakukan dengan menggunakan statement SELECT.

CONTOH TABEL

EMPLOYEES			DEPARTMENTS		
EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID	DEPARTMENT_ID	DEPARTMENT_NAME	LOCATION_ID
180	King	90	10	Administration	1700
191	Hochstein	90	20	Marketing	1800
...			30	Shipping	1900
202	Fay	30	50	IT	1400
205	Higgins	110	60	Sales	2500
206	Chatz	110	80	Executive	1700
			110	Accounting	1700
			120	Contracting	1700

EMPLOYEE_ID	DEPARTMENT_ID	DEPARTMENT_NAME
200	10	Administration
201	20	Marketing
202	20	Marketing
...		
192	80	Executive
205	110	Accounting
206	110	Accounting

MENDAPATKAN DATA DARI TABEL

- Kadang-kadang Anda perlu untuk menggunakan data dari beberapa tabel. Contoh pada slide, suatu laporan menampilkan data dari dua tabel yang terpisah:
 - Nomor-nomor pegawai ada pada tabel EMPLOYEES.
 - Nomor-nomor departemen ada pada tabel EMPLOYEES dan tabel DEPARTEMENTS.
 - Nama-nama departemen ada pada tabel DEPARTEMENTS.
- Untuk menghasilkan laporan, Anda perlu untuk menghubungkan tabel EMPLOYEES dan tabel DEPARTEMENTS dan mengakses data dari kedua tabel tersebut.

TIPE-TIPE JOIN

- INNER JOIN
- CROSS JOIN
- OUTER JOIN
- STRAIGHT JOIN

INNER JOIN

- Tipe join ini akan mengambil semua row dari table asal dan table tujuan dengan kondisi nilai key yang terkait saja - jika ada, dan jika tidak maka row tersebut tidak akan muncul.
- Kalau tidak terdapat kondisi key terkait antar table, maka semua row dari kedua table dikombinasikan.

INNER JOIN

- SYNTAX :

table_reference [INNER] JOIN table_factor [join_condition]

```
SELECT * FROM ms_cabang INNER JOIN ms_kota ON
ms_cabang.kode_kota = ms_kota.kode_kota
```

Disini table sumber adalah ms_cabang (left) mencari referensi row lain dari table ms_kota (right) dengan kondisi nilai **kode_kota** diantara kedua table tersebut sama. Kondisi ini menggunakan keyword **ON**.

HASIL QUERY

```
1 SELECT * FROM ms_cabang
2 INNER JOIN ms_kota
3 ON ms_cabang.kode_kota = ms_kota.kode_kota
```

kode_cabang	nama_cabang	kode_kota	kode_kota	nama_kota	kode_propinsi
CABANG-001	FHI Mini Market - Lhokseumawe 01	KOTA-003	KOTA-003	Lhokseumawe	730
CABANG-002	FHI Mini Market - Bau-Bau 01	KOTA-083	KOTA-083	Bau-Bau	725
CABANG-003	FHI Mini Market - Bogor 01	KOTA-039	KOTA-039	Bogor	708
CABANG-004	FHI Mini Market - Medan 01	KOTA-007	KOTA-007	Medan	730
CABANG-005	FHI Mini Market - Bogor 02	KOTA-039	KOTA-039	Bogor	708
CABANG-006	FHI Mini Market - Sukabumi 01	KOTA-043	KOTA-043	Sukabumi	708
CABANG-007	FHI Mini Market - Tegal 01	KOTA-055	KOTA-055	Tegal	709
CABANG-008	FHI Mini Market - Jakarta Timur 01	KOTA-048	KOTA-048	Jakarta Timur	732
CABANG-009	FHI Mini Market - Panusunan 01	KOTA-062	KOTA-062	Panusunan	710
CABANG-010	FHI Mini Market - Langsa 01	KOTA-062	KOTA-062	Langsa	733
CABANG-011	FHI Mini Market - Tareknan 01	KOTA-078	KOTA-078	Tareknan	713
CABANG-012	FHI Mini Market - Tebing Tinggi 01	KOTA-012	KOTA-012	Tebing Tinggi	730
CABANG-013	FHI Mini Market - Bekasi 01	KOTA-038	KOTA-038	Bekasi	708
CABANG-014	FHI Mini Market - Batu 01	KOTA-056	KOTA-056	Batu	710
CABANG-015	FHI Mini Market - Banjarmasin 01	KOTA-079	KOTA-079	Banjarmasin	714

INNER JOIN

- untuk row pertama kita memiliki kode_kota dengan nilai "KOTA-003", ini akan dicari referensinya ke table ms_kota untuk nilai yang sama dan kita dapatkan row dengan nilai nama_kota "Lhokseumawe" adalah padanannya.
- untuk row kedua kita memiliki kode_kota dengan nilai "KOTA-083", ini akan dicari referensinya ke table ms_kota untuk nilai yang sama dan kita dapatkan row dengan nilai nama_kota "Bau-bau" adalah padanannya.
- demikian seterusnya.

CROSS JOIN

- FUNGSI DAN PENJELASANNYA SAMA DENGAN INNER JOIN
- SYNTAX :

```
SELECT ms_cabang.nama_cabang,
ms_kota.nama_kota,
ms_propinsi.nama_propinsi
FROM
ms_cabang
CROSS JOIN
ms_kota ON ms_cabang.kode_kota = ms_kota.kode_kota
CROSS JOIN
ms_propinsi ON ms_kota.kode_propinsi =
ms_propinsi.kode_propinsi
```

OUTER JOIN

- **OUTER JOIN** merupakan tipe join yang mencari referensi data dari suatu table sumber ke table lain dengan tidak menghilangkan data sumber apabila referensi tidak ditemukan.
- Untuk menggunakan tipe OUTER JOIN maka perlu memperhatikan beberapa hal berikut :
 - perlu dibedakan antara table sumber dan table referensi, ini ditentukan dengan cara menspesifikasikan kedudukan table sumber apakah di kiri (**LEFT**) atau di kanan (**RIGHT**).
 - jika tidak ada data dari table referensi yang cocok dengan kondisi join maka hanya data dari table sumber yang ditampilkan tetapi kolom-kolom table referensi akan berisi null.

Query

```

1 SELECT ms_cabang.*,
2       ms_kota.nama_kota
3 FROM ms_cabang
4 LEFT OUTER JOIN ms_kota
5     ON ms_cabang.kode_kota = ms_kota.kode_kota
6
7

```

1 Result

kode_cabang	nama_cabang	kode_kota	nama_kota
<input type="checkbox"/>	CABANG-001	PHI Mini Market - Lhokseumawe 01	KOTA-003 (NULL)
<input type="checkbox"/>	CABANG-002	PHI Mini Market - Bau-Bau 01	KOTA-083 Bau-Bau
<input type="checkbox"/>	CABANG-003	PHI Mini Market - Bogor 01	KOTA-039 Bogor
<input type="checkbox"/>	CABANG-004	PHI Mini Market - Medan 01	KOTA-007 Medan
<input type="checkbox"/>	CABANG-005	PHI Mini Market - Bogor 02	KOTA-039 Bogor
<input type="checkbox"/>	CABANG-006	PHI Mini Market - Sukabumi 01	KOTA-043 Sukabumi
<input type="checkbox"/>	CABANG-007	PHI Mini Market - Tegal 01	KOTA-055 Tegal
<input type="checkbox"/>	CABANG-008	PHI Mini Market - Jakarta Timur 01	KOTA-048 Jakarta Timur
<input type="checkbox"/>	CABANG-009	PHI Mini Market - Pasuruan 01	KOTA-062 Pasuruan
<input type="checkbox"/>	CABANG-010	PHI Mini Market - Langsa 01	KOTA-002 Langsa
<input type="checkbox"/>	CABANG-011	PHI Mini Market - Tarakan 01	KOTA-078 Tarakan

CARTESIAN PRODUCT

- Cartesian Product adalah hasil dari join yang tidak valid, sehingga menyebabkan ditampilkannya seluruh kombinasi data dari tabel-tabel yang di-join-kan. Seluruh baris dari tabel pertama di-join-kan dengan seluruh baris pada tabel kedua.
- Jika saat melakukan select data yang terlihat double atau ganda maka inilah disebut CARTESIAN PRODUCT

UNTUK LEBIH JELAS

<http://mysql.phi-integration.com/sql/tipe-tipe-join-pada-mysql-5>

<http://alvian-cs.blogspot.co.id/2014/11/penggunaan-inner-left-right-join-pada.html>

TERIMA KASIH