

REVIEW

SINGLE AND GROUP FUNCTION



	A Nama Lengkap	B Telpon
1	WinstonTaylor	650.507.9876
2	WilliamSmith	011.44.1343.629268
3	WilliamGietz	515.123.8181
4	VanceJones	650.501.4876
5	ValliPataballa	Tidak ada
6	TrennaRajs	650.121.8009
7	TimothyGates	650.505.3876
8	TaylerFox	011.44.1343.729268
9	TJOlson	650.124.8234
10	SusanMavris	515.123.7777
11	SunditaKumar	011.44.1343.329268
12	SundarAnde	011.44.1346.629268
13	StevenMarkle	650.124.1434
14	StevenKing	Tidak ada

1. Kolom apa saja yang digunakan?
2. Fungsi apa saja yang digunakan?
3. Apakah ada metode *sorting*? Jika ada, *sorting* berdasarkan kolom apa dan apa metode *sorting* yang digunakan?
4. Apakah ada penggunaan Alias? Jika ada, cara apa yang digunakan untuk membuat Alias?

Nama Lengkap	SALARY	TINGKAT
DONALD OCONNELL	2600	Kelas 1
DOUGLAS GRANT	2600	Kelas 1
JENNIFER WHALEN	4400	Kelas 1
MICHAEL HARTSTEIN	13000	Kelas 2
PAT FAY	6000	Kelas 2
SUSAN MAVRIS	6500	Kelas 2
HERMANN BAER	10000	Kelas 2
SHELLEY HIGGINS	12000	Kelas 2
WILLIAM GIETZ	8300	Kelas 2
STEVEN KING	24000	Kelas 3
NEENA KOCHHAR	17000	Kelas 3

1. Kolom apa saja yang digunakan?
2. Fungsi apa saja yang digunakan?
3. Apakah ada penggunaan Alias? Jika ada, cara apa yang digunakan untuk membuat Alias?

NAMA LENGKAP	PHONE_NUMBER	Kategori Job
Donald OConnell	650.507.9833	Senior
Douglas Grant	650.507.9844	Senior
Jennifer Whalen	515.123.4444	Junior
Michael Hartstein	515.123.5555	Baru
Pat Fay	603.123.6666	Baru
Susan Mavris	515.123.7777	Baru
Hermann Baer	XXX	Baru
Shelley Higgins	515.123.8080	Junior
William Gietz	515.123.8181	Junior
Steven King	XXX	Junior
Neena Kochhar	515.123.4568	Junior
Lex De Haan	515.123.4569	Junior

Tampilkan nama lengkap karyawan beserta nomor telpon dan kategori job-nya.

Ketentuan kategori job adalah :

Jika job diawali AC dan AD, kategorinya "Junior"

Jika job diawali FI dan IT, kategorinya "Madya"

Jika job diawali SA dan SH, kategorinya "Senior"

Jika tidak termasuk kelompok diatas, kategorinya "Baru"

Menampilkan gaji tertinggi, gaji terendah, rata-rata gaji, dan total seluruh gaji karyawan.

Gaji Tertinggi	Gaji Terendah	Rata2 Gaji	Total Gaji
24000	2100	6461.68224299065	691400

JOB_ID	TERTINGGI
SA_MAN	14000
SA_REP	11500
ST_MAN	8200
SH_CLERK	4200
ST_CLERK	3600

Menampilkan gaji tertinggi untuk masing-masing job untuk job yang diawali huruf "S". Urutkan berdasarkan gaji tertinggi yang paling tinggi!

JOB_ID	Jumlah Karyawan
SA_REP	30
SH_CLERK	20
ST_CLERK	20

Menampilkan jumlah karyawan per kode job karyawan dan tampilkan yang jumlah karyawannya lebih dari 15.

JOIN TABLE





Menampilkan Data Dari Beberapa Tabel

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Mendapatkan Data dari Beberapa Tabel

EMPLOYEES

EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID
100	King	90
101	Kochhar	90
...		
202	Fay	20
205	Higgins	110
206	Gietz	110

DEPARTMENTS

DEPARTMENT_ID	DEPARTMENT_NAME	LOCATION_ID
10	Administration	1700
20	Marketing	1800
50	Shipping	1500
60	IT	1400
80	Sales	2500
90	Executive	1700
110	Accounting	1700
190	Contracting	1700

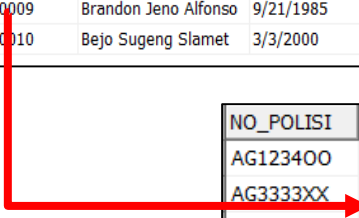


EMPLOYEE_ID	DEPARTMENT_ID	DEPARTMENT_NAME
200	10	Administration
201	20	Marketing
202	20	Marketing
...		
102	90	Executive
205	110	Accounting
206	110	Accounting

ORACLE

Kegunaan Pernyataan SELECT

ID_DRIVER	NAMA_DRIVER	LAHIR_DRIVER	ALMT_DRIVER	HP_DRIVER	JKEL_DRIVER	KTP_DRIVER	NO_SIM	FOTO_DRIVER	TGL_DAFTAR
D0001	Surya Adi Sastra	7/23/1998	Jl. Mawar 12 Semarang	0812229182	L	355220000903	778776000903	D:/Driver/Foto/D0001.jpg	1/10/2017
D0002	Joko Lelono	4/1/2001	Jl. Kedung Baruk 121 Surabaya	0833433232	L	355220000934	778776000666	D:/Driver/Foto/D0002.jpg	7/20/2018
D0003	Reno Alfian	8/19/2003	Gubeng Kertajaya IV/23 Surabaya	0323110331	L	355220000935	778776000777	D:/Driver/Foto/D0003.jpg	3/3/2019
D0004	Moreno Syah	1/5/1989	Bratang Gede III - 12 Surabaya	0833433232	L	355220000936	778776000454	D:/Driver/Foto/D0004.jpg	1/10/2017
D0005	Candra Budiono	1/15/2001	Jl. Ahmad Yani II / 234 Sidoarjo	1343756133	L	355220000937	778776000345	D:/Driver/Foto/D0005.jpg	7/20/2018
D0006	Sapta Argananto	3/3/1987	Kertajaya Utara VI / 23 Surabaya	1854079034	L	355220000938	778776000233	D:/Driver/Foto/D0006.jpg	8/15/2017
D0007	Jeni Anindita	9/10/2000	Jl. Merpati V - 87 Semarang	2364401935	P	355220000939	778776000567	D:/Driver/Foto/D0007.jpg	7/20/2018
D0008	Renso Malako	1/11/2001	Jl. Kenjeran Selatan 34 Surabaya	1420242460	L	355220000945	778776000765	D:/Driver/Foto/D0008.jpg	1/10/2017
D0009	Brandon Jenso Alfonso	9/21/1985	Gubeng Kertajaya Tengah V/55 Surabaya	0909919559	L	355220004544	778776000899	D:/Driver/Foto/D0009.jpg	9/17/2018
D0010	Bejo Sugeng Slamet	3/3/2000	Kedung Asem III - 34 Surabaya	0399596658	L	355220045555	778776000123	D:/Driver/Foto/D0010.jpg	7/20/2018

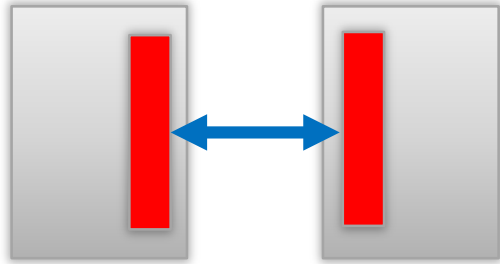


NO_POLISI	ID_DRIVER	MEREK	TIPE	CC	WARNA
AG123400	D0001	Honda	Beat	125	Hitam
AG3333XX	D0002	Honda	Supra	125	Biru
AE4545LL	D0003	Yamaha	Mio	125	Merah
L4564XX	D0004	Yamaha	Mio	125	Hitam

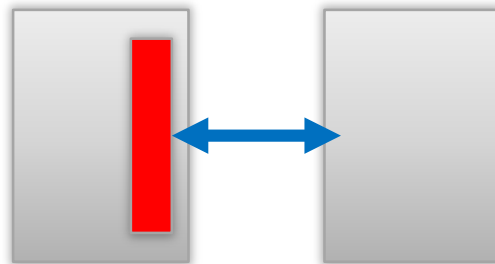
Menampilkan data DRIVER dan KENDARAANNYA

ID_DRIVER	NAMA_DRIVER	JKEL_DRIVER	NO_SIM	TGL_DAFTAR	NO_POLISI	MEREK	TIPE	CC	WARNA
D0001	Surya Adi Sastra	L	778776000903	1/10/2017	AG123400	Honda	Beat	125	Hitam
D0002	Joko Lelono	L	778776000666	7/20/2018	AG3333XX	Honda	Supra	125	Biru
D0003	Reno Alfian	L	778776000777	3/3/2019	AE4545LL	Yamaha	Mio	125	Merah
D0004	Moreno Syah	L	778776000454	1/10/2017	L4564XX	Yamaha	Mio	125	Hitam

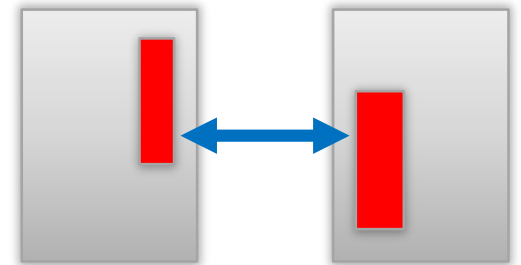
Type-Type Join



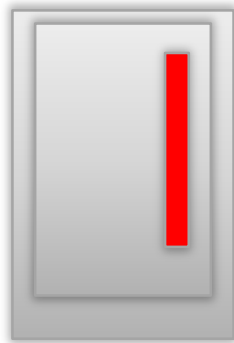
Equijoin



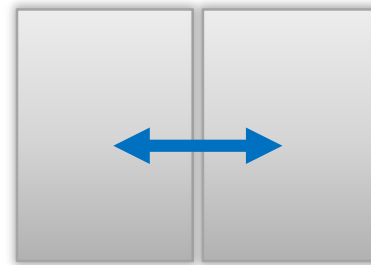
Non-Equijoin



Outer Join



Self Join



Cartesian Product

Menggabungkan Tabel-tabel Menggunakan Sintak SQL:1999

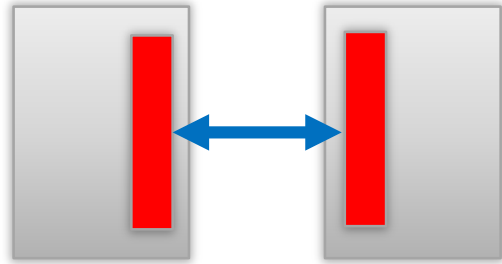
Gunakan suatu *join* untuk meng-query data dari beberapa tabel :

```
SELECT  table1.column, table2.column
FROM    table1
[NATURAL JOIN table2] |
[JOIN table2 USING (column_name)] |
[JOIN table2
  ON (table1.column_name = table2.column_name)] |
[LEFT|RIGHT|FULL OUTER JOIN table2
  ON (table1.column_name = table2.column_name)] |
[CROSS JOIN table2];
```

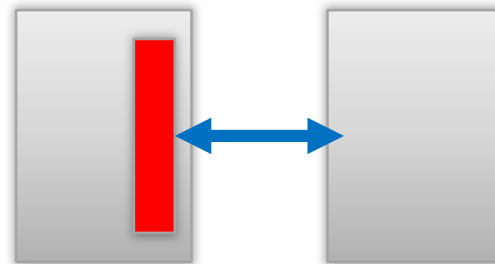
ORACLE

Copyright © 2004, Oracle. All rights reserved.

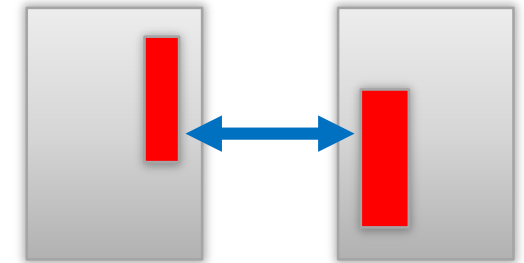
Type-Type Join



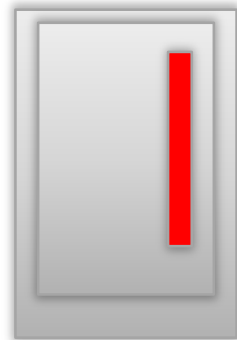
Equijoin
NATURAL JOIN
JOIN...USING...
JOIN...ON....



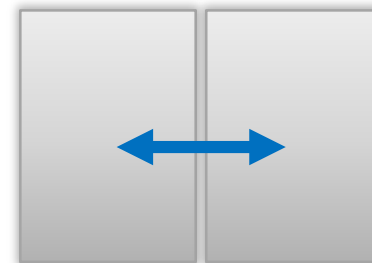
Non-Equijoin
JOIN...ON....



Outer Join
LEFT OUTER JOIN...ON....
RIGHT OUTER JOIN...ON...
FULL OUTER JOIN...ON...



Self Join
JOIN...ON....



Cartesian Product
....CROSS JOIN....

Mendapatkan *Record-Record* dengan *Natural Joins*

```
SELECT department_id, department_name,  
       location_id, city  
FROM   departments  
NATURAL JOIN locations ;
```

DEPARTMENT_ID	DEPARTMENT_NAME	LOCATION_ID	CITY
60	IT	1400	Southlake
50	Shipping	1500	South San Francisco
10	Administration	1700	Seattle
90	Executive	1700	Seattle
110	Accounting	1700	Seattle
190	Contracting	1700	Seattle
20	Marketing	1800	Toronto
80	Sales	2500	Oxford

8 rows selected.

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Mendapatkan *Record-Record* dengan Klausa USING

```
SELECT employees.employee_id, employees.last_name,  
       departments.location_id, department_id  
FROM   employees JOIN departments  
USING (department_id);
```

EMPLOYEE_ID	LAST_NAME	LOCATION_ID	DEPARTMENT_ID
200	Whalen	1700	10
201	Hartstein	1800	20
202	Fay	1800	20
134	Mourgos	1500	50
141	Rajs	1500	50
142	Davies	1500	50
144	Vargas	1500	50
143	Matos	1500	50

19 rows selected.

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Mengambil *Record-Record* dengan Klausa ON

```
SELECT e.employee_id, e.last_name, e.department_id,  
       d.department_id, d.location_id  
FROM   employees e JOIN departments d  
ON     (e.department_id = d.department_id);
```

EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID	DEPARTMENT_ID	LOCATION_ID
200	Whalen	10	10	1700
201	Hartstein	20	20	1800
202	Fay	20	20	1800
124	Mourgos	50	50	1500
141	Rajs	50	50	1500
142	Davies	50	50	1500
143	Mates	50	50	1500

19 rows selected.

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Self-Joins Menggunakan Klausa ON

```
SELECT e.last_name emp, m.last_name mgr
FROM   employees e JOIN employees m
ON     (e.manager_id = m.employee_id);
```

EMP	MGR
Hartstein	King
Zlotkey	King
Mourges	King
De Haan	King
Kochhar	King

19 rows selected.

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Menerapkan Kondisi-Kondisi Tambahan untuk Suatu *Join*

```
SELECT e.employee_id, e.last_name, e.department_id,  
       d.department_id, d.location_id  
FROM   employees e JOIN departments d  
ON     (e.department_id = d.department_id)  
AND   e.manager_id = 149 ;
```

EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID	DEPARTMENT_ID	LOCATION_ID
174	Abel	80	80	2500
176	Taylor	80	80	2500

Membuat *Three-Way Joins* dengan Klausa ON

```
SELECT employee_id, city, department_name
FROM   employees e
JOIN   departments d
ON     d.department_id = e.department_id
JOIN   locations l
ON     d.location_id = l.location_id;
```

EMPLOYEE_ID	CITY	DEPARTMENT_NAME
103	Southlake	IT
104	Southlake	IT
107	Southlake	IT
124	South San Francisco	Shipping
141	South San Francisco	Shipping
142	South San Francisco	Shipping
143	South San Francisco	Shipping
144	South San Francisco	Shipping

19 rows selected.

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Mendapatkan *Record-Record* dengan *Non-Equi*joins

```
SELECT e.last_name, e.salary, j.grade_level
FROM   employees e JOIN job_grades j
ON     e.salary
      BETWEEN j.lowest_sal AND j.highest_sal;
```

LAST_NAME	SALARY	GRA
Matos	2600	A
Vargas	2600	A
Lorentz	4200	B
Mourgos	6800	B
Rajs	3600	B
Dowies	3100	B
Whalen	4400	B
Hunold	9000	C
Ernat	6000	C

20 rows selected.

ORACLE

Outer Joins

DEPARTMENTS

DEPARTMENT_NAME	DEPARTMENT_ID
Administration	10
Marketing	20
Shipping	50
IT	60
Sales	60
Executive	90
Accounting	110
Contracting	190

8 rows selected.

EMPLOYEES

DEPARTMENT_ID	LAST_NAME
90	King
90	Kochhar
90	De Haan
60	Hunold
60	Ernst
60	Lorentz
60	Mourgos
60	Rajs
60	Davies
60	Matos
60	Verges
60	Zlotkey

20 rows selected.

Tidak ada pegawai di departemen 190.

ORACLE

Copyright © 2004, Oracle. All rights reserved.

LEFT OUTER JOIN

```
SELECT e.last_name, e.department_id, d.department_name
FROM   employees e LEFT OUTER JOIN departments d
ON     (e.department_id = d.department_id) ;
```

LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
Whalen	10	Administration
Fay	20	Marketing
Hartstein	20	Marketing
...		
De Haan	90	Executive
Kochhar	90	Executive
King	90	Executive
Gietz	110	Accounting
Higgins	110	Accounting
Grant		

20 rows selected.

ORACLE

Copyright © 2004, Oracle. All rights reserved.

RIGHT OUTER JOIN

```
SELECT e.last_name, e.department_id, d.department_name
FROM   employees e RIGHT OUTER JOIN departments d
ON     (e.department_id = d.department_id) ;
```

LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
Whalen	10	Administration
Fay	20	Marketing
Hartstein	20	Marketing
Davies	50	Shipping

Kochhar	90	Executive
Gietz	110	Accounting
Higgins	110	Accounting
	190	Contracting

20 rows selected.

ORACLE

Copyright © 2004, Oracle. All rights reserved.

FULL OUTER JOIN

```
SELECT e.last_name, d.department_id, d.department_name
FROM   employees e FULL OUTER JOIN departments d
ON     (e.department_id = d.department_id) ;
```

LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
Whalen	10	Administration
Fay	20	Marketing
Hartstein	20	Marketing
...		
King	90	Executive
Gietz	110	Accounting
Higgins	110	Accounting
Grant		
	190	Contracting

21 rows selected.

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Cartesian Products

- Suatu ***Cartesian product*** terbentuk ketika :
 - Suatu kondisi *join* dihilangkan
 - Suatu kondisi *join* tidak tepat
 - Seluruh baris dalam tabel pertama digabungkan ke seluruh baris dalam tabel kedua
- Untuk menghindari suatu ***Cartesian product***, selalu sertakan sebuah kondisi *join* yang tepat.

Membangkitkan Suatu *Cartesian Product*

EMPLOYEES (20 rows)

EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID
100	King	90
101	Kochhar	90
...		
202	Fay	20
205	Higgins	110
206	Kozl	110

20 rows selected.

DEPARTMENTS (8 rows)

DEPARTMENT_ID	DEPARTMENT_NAME	LOCATION_ID
10	Administration	1700
20	Marketing	1800
50	Shipping	1500
60	IT	1400
80	Sales	2500
90	Executive	1700
110	Accounting	1700
190	Contracting	1700

8 rows selected.

Cartesian product:
 $20 \times 8 = 160$ baris

EMPLOYEE_ID	DEPARTMENT_ID	LOCATION_ID
100	90	1700
101	90	1700
102	90	1700
103	60	1700
104	60	1700
107	60	1700
...		

160 rows selected.

ORACLE

Membuat *Cross Join*

- Klausa `CROSS JOIN` menghasilkan *cross-product* dari dua tabel.
- `CROSS JOIN` disebut juga *Cartesian product* antara dua tabel.

```
SELECT last_name, department_name  
FROM employees  
CROSS JOIN departments ;
```

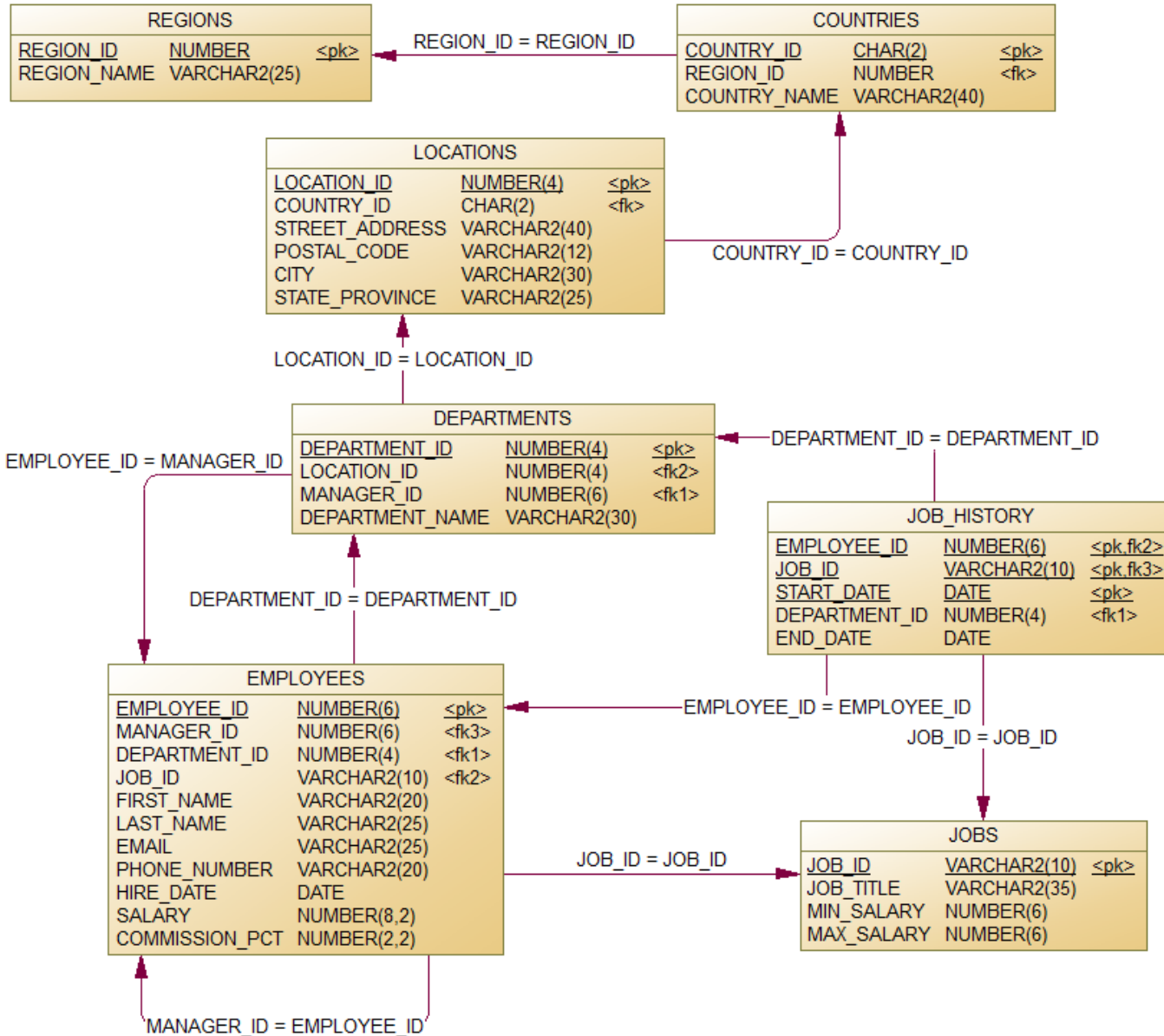
LAST_NAME	DEPARTMENT_NAME
King	Administration
Kochhar	Administration
De Haan	Administration
Hunold	Administration

160 rows selected.

ORACLE

Copyright © 2004, Oracle. All rights reserved.

SKEMA HR



Praktekkan Klausa JOIN dari skema Akademik!

1. Tampilkan data nama mahasiswa beserta nama prodi nya dengan detil yang ditampilkan adalah NIM, nama lengkap mahasiswa, kota tinggal, jenis kelamin, kode prodi, nama prodi, dan masa studinya. Urutkan berdasarkan NIM!
2. Tampilkan data dosen beserta jadwal mengajarnya dengan detil data adalah nama dan NID dosen, kode MK, hari, jam mulai dan jam selesai mengajar. Tampilkan hanya jadwal hari Senin dan Rabu saja!
3. Tampilkan data mahasiswa beserta hasil studinya dengan detil data : NIM, nama lengkap mahasiswa, kode prodi, jenis kelamin, kode MK, nama MK, SKS, semester, nilai tugas, UTS, dan nilai UAS nya. Tampilkan hanya untuk mahasiswa prodi SI Desain Komunikasi dan Visual dan SI Teknologi Informasi saja. Urutkan berdasarkan kode prodi dan NIM nya (A-Z)!
4. **Buat tangkapan layar** untuk hasil klausa JOIN diatas. Hasil kumpulan tangkapan layar diunggah melalui Brilian dengan nama file : **Demo5_Q1_NimPanjang>Nama.PDF atau Demo5_P1_NimPanjang>Nama.PDF paling lambat hari Sabtu, 19 Juni 2021 pukul 23:59 WIB.**

Tetap
Semangat