Nama : Raden Sadiah Maharani

NIM : 2107126368

Basis Data Lanjut

Praktikum 8

1. Tuliskan query dengan perintah menggunakan penyataan INSERT eksplisit untuk menambah empat CD baru pada DJ on Demand ke tabel copy\_d\_cds. Setelah menyelesaikan entri, jalankan pernyataan SELECT \* untuk memverifikasi pekerjaan Anda.

→ CREATE TABLE copy\_d\_cds  
AS SELECT \* FROM d\_cds;

INSERT INTO copy\_d\_cds(cd\_number, title, producer, year) VALUES (97, 'Celebrate the Day', 'R & B Inc.', 2003);

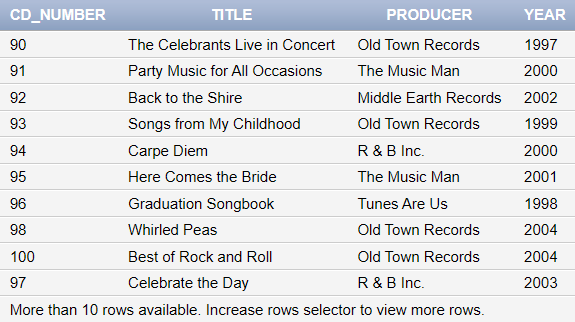
INSERT INTO copy\_d\_cds(cd\_number, title, producer, year) VALUES (98, 'Holiday Tunes for All Ages', 'Tunes are Us', 2004);

INSERT INTO copy\_d\_cds(cd\_number, title, producer, year) VALUES (99, 'Party Music', 'Old Town Records', 2004);

INSERT INTO copy\_d\_cds(cd\_number, title, producer, year) VALUES (100, 'Best of Rock and Roll', 'Old Town Records', 2004);

SELECT \* FROM copy\_d\_cds;

Result :



1. DJ on Demand memiliki dua acara baru yang akan datang. Satu acara adalah pesta sepak bola musim gugur dan acara lainnya adalah pesta bertema tahun enam puluhan. Klien DJ on Demand meminta lagu yang ditampilkan di tabel untuk acara mereka. Tambahkan lagu-lagu ini ke tabel copy\_d\_songs menggunakan pernyataan INSERT implisit.

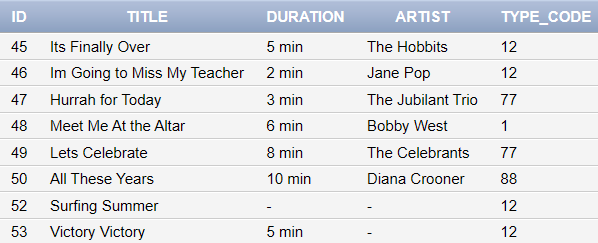
→ CREATE TABLE copy\_d\_songs AS   
SELECT \* FROM d\_songs;

INSERT INTO copy\_d\_songs VALUES(52,'Surfing Summer',NULL,NULL,12);

INSERT INTO copy\_d\_songs VALUES(53,'Victory Victory','5 min',NULL,12);

SELECT \* FROM copy\_d\_songs;

Result :



1. Buat tabel bernama rep\_email menggunakan pernyataan berikut:

first\_name VARCHAR2(10),

last\_name VARCHAR2(10),

email\_address VARCHAR2(10))

Isi tabel ini dengan menjalankan kueri pada tabel karyawan yang hanya menyertakan karyawan yang merupakan REP.

→ ALTER TABLE rep\_email DROP column id;

ALTER TABLE rep\_email ADD id NUMBER(6,0) CONSTRAINT rel\_id\_pk PRIMARY KEY;

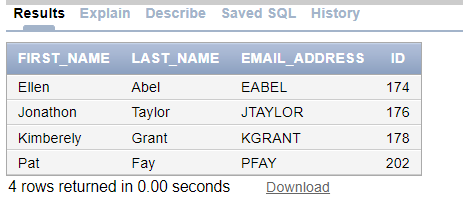
INSERT INTO rep\_email(id, first\_name, last\_name, email\_address) SELECT employee\_id, first\_name, last\_name, email

FROM employees

WHERE job\_id LIKE '%\\_REP' ESCAPE '\';

SELECT \* FROM rep\_email;

Result :



1. Buat perubahan harga pada tabel copy\_f\_food\_items. Dengan perubahan harga strawberry shake akan dinaikkan dari $3,59 menjadi $3,75, dan harga kentang goreng akan naik menjadi $1,20.

→ CREATE TABLE copy\_f\_food\_items AS ( SELECT \* FROM f\_food\_items);

UPDATE copy\_f\_food\_items SET price = 3.75 WHERE LOWER(description) = 'strawberry shake';

UPDATE copy\_f\_food\_items SET price = 1.20 WHERE LOWER(description) = 'fries';

SELECT \* FROM copy\_f\_food\_items;

Result :



1. Tambahkan pesanan yang ditampilkan ke tabel copy\_f\_orders Makanan Cepat Saji Global:



→ CREATE TABLE copy\_f\_orders

AS ( SELECT \* FROM f\_orders);

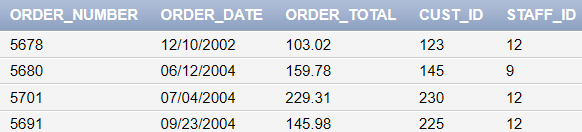
INSERT INTO copy\_f\_orders(order\_number,order\_date,order\_total,cust\_id,staff\_id) VALUES(5680,TO\_DATE('June 12, 2004','fmMonth dd, yyyy'),159.78,145,9);

INSERT INTO copy\_f\_orders(order\_number,order\_date,order\_total,cust\_id,staff\_id) VALUES(5691,TO\_DATE('09-23-2004','mm-dd-yyyy'),145.98,225,12);

INSERT INTO copy\_f\_orders(order\_number,order\_date,order\_total,cust\_id,staff\_id) VALUES(5701,TO\_DATE('July 4, 2004','fmMonth dd, yyyy'),229.31,230,12);

SELECT \* FROM copy\_f\_orders;

Result :



1. Sue Doe telah menjadi anggota staf Global Foods yang luar biasa dan telah mendapatkan kenaikan gaji. Dia sekarang akan dibayar sama dengan Bob Miller. Perbarui catatannya di copy\_f\_staffs.

→ CREATE TABLE copy\_f\_staffs AS ( SELECT \* FROM f\_staffs);

UPDATE copy\_f\_staffs SET salary = (SELECT salary FROM copy\_f\_staffs WHERE LOWER(first\_name || ' ' || last\_name) = 'bob miller')

WHERE LOWER(first\_name || ' ' || last\_name) = 'sue doe';

SELECT \* from copy\_f\_staffs;

Result :



1. Sekarang semua informasi tersedia untuk Kai Kim, perbarui catatan Global Fast Foods-nya untuk menyertakan yang berikut: Kai akan memiliki manajer yang sama dengan Sue Doe. Dia tidak memenuhi syarat untuk lembur. Biarkan nilai untuk pelatihan, anggaran manajer, dan target manajer sebagai null.

→ INSERT INTO

copy\_f\_staffs(id, first\_name, last\_name, birthdate, salary, overtime\_rate, training, staff\_type, manager\_id, manager\_budget, manager\_target)

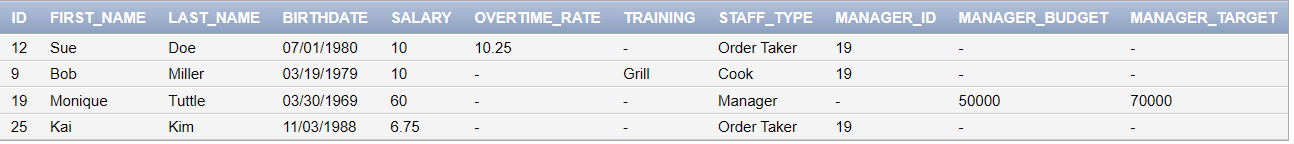
VALUES (25, 'Kai', 'Kim', TO\_DATE('03-Nov-1988','fmdd-Mon-yyyy'), 6.75, NULL, NULL, 'Order Taker', NULL, NULL, NULL);

UPDATE copy\_f\_staffs SET manager\_id = (SELECT manager\_id FROM copy\_f\_staffs WHERE LOWER(first\_name || ' ' || last\_name) = 'sue doe')

WHERE LOWER(first\_name || ' ' || last\_name) = 'kai kim';

SELECT \* FROM copy\_f\_staffs;

Result :



1. Buat salinan tabel karyawan dan beri nama lesson7\_emp;

Setelah tabel ini ada, tulis pernyataan hapus berkorelasi yang akan menghapus semua karyawan dari tabel lesson7\_employees yang juga ada di tabel job\_history.

→ CREATE TABLE lesson7\_emp

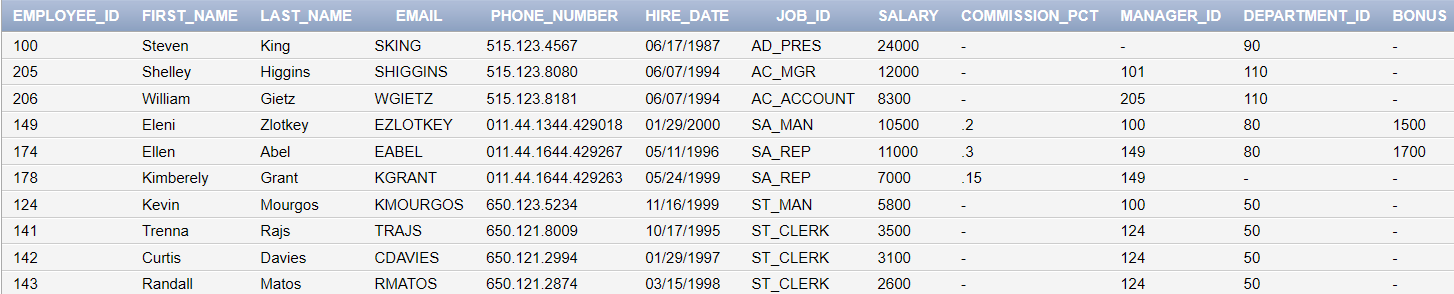
AS ( SELECT \* FROM employees);

DELETE FROM lesson7\_emp

WHERE employee\_id IN ( SELECT DISTINCT employee\_id FROM job\_history);

SELECT \* FROM lesson7\_emp;

Result :



1. Jalankan 3 pernyataan berikut untuk membuat 3 tabel baru untuk digunakan dalam pernyataan penyisipan Multi-tabel. Semua 3 tabel harus kosong pada saat pembuatan, karenanya kondisi WHERE 1=2 dalam klausa WHERE.

CREATE TABLE sal\_history (employee\_id, hire\_date, salary) AS SELECT employee\_id, hire\_date, salary

FROM employees

WHERE 1=2;

CREATE TABLE mgr\_history (employee\_id, manager\_id, salary) AS SELECT employee\_id, manager\_id, salary

FROM employees

WHERE 1=2;

CREATE TABLE special\_sal (employee\_id, salary)

AS SELECT employee\_id, salary

FROM employees WHERE 1=2;

Setelah tabel ada di akun Anda, tulis pernyataan penyisipan Multi-Tabel untuk memilih terlebih dahulu employee\_id, hire\_date, salary, dan manager\_id dari semua karyawan. Jika gaji lebih dari 20000 masukkan employee\_id dan salary ke dalam tabel special\_sal. Masukkan detail employee\_id, hire\_date, dan salary ke dalam tabel

sal\_history. Masukkan employee\_id, manager\_id, dan salary ke dalam tabel mgr\_history.

Anda harus mendapatkan pesan kembali yang mengatakan 39 baris telah dimasukkan. Verifikasi Anda mendapatkan pesan ini dan verifikasi Anda memiliki jumlah baris berikut di setiap tabel:

Sal\_history: 19 baris

Mgr\_history: 19 baris

Spesial\_sal: 1

→ CREATE TABLE sal\_history (employee\_id, hire\_date, salary) AS SELECT employee\_id, hire\_date, salary

FROM employees

WHERE 1=2;

CREATE TABLE mgr\_history (employee\_id, manager\_id, salary) AS SELECT employee\_id, manager\_id, salary

FROM employees

WHERE 1=2;

CREATE TABLE special\_sal (employee\_id, salary)

AS SELECT employee\_id, salary

FROM employees WHERE 1=2;

INSERT FIRST

WHEN salary > 20000 THEN

INTO special\_sal (employee\_id, salary) VALUES(employee\_id, salary)

WHEN salary <= 20000 THEN

INTO sal\_history (employee\_id, hire\_date, salary) VALUES(employee\_id, hire\_date, salary)

INTO mgr\_history (employee\_id, manager\_id, salary) VALUES(employee\_id, manager\_id, salary)

SELECT employee\_id, salary, hire\_date, manager\_id FROM employees;

SELECT COUNT(\*) FROM sal\_history;

SELECT COUNT(\*) FROM mgr\_history;

SELECT COUNT(\*) FROM special\_sal;

Result :

Sal\_history:



Mgr\_history:



Spesial\_sal:

