Nama : Afina Putri Dayanti

NIM : 825200049

Jurusan : Sistem Informasi

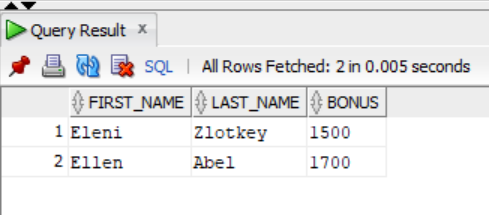
Mata Kuliah : UAS Database Systems (Praktek)

1. Kerjakan soal-soal kasus berikut ini dengan seksama dan sertakan hasil outputnya ?
2. Tuliskan SQL statement yang menampilkan nama depan, nama belakang, dan bonus karyawan yang memiliki bonus di antara 1500 sampai dengan 2000 (menggunakan BETWEEN)

SELECT first\_name, last\_name, bonus

FROM employees

WHERE bonus between 1500 and 2000;

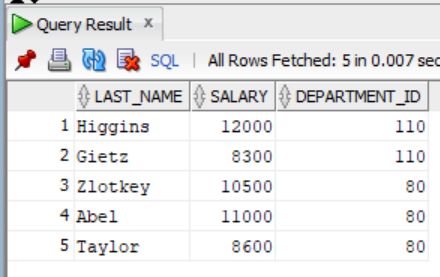
****

1. Tuliskan SQL statement yang menampilkan last\_name, salary, department\_id untuk karyawan yang berada di departemen 70, 80 dan 110 (menggunakan IN)

SELECT last\_name, salary, department\_id

FROM employees

WHERE department\_id IN (70,80,110);

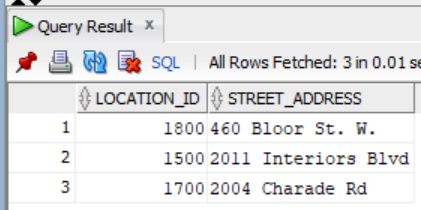


1. Tampilkan seluruh location ID dan street address dari tabel locations, kecuali yang memiliki location ID 2500 dan 1400.

SELECT location\_id, street\_address

FROM locations

WHERE location\_id NOT IN (2500, 1400);

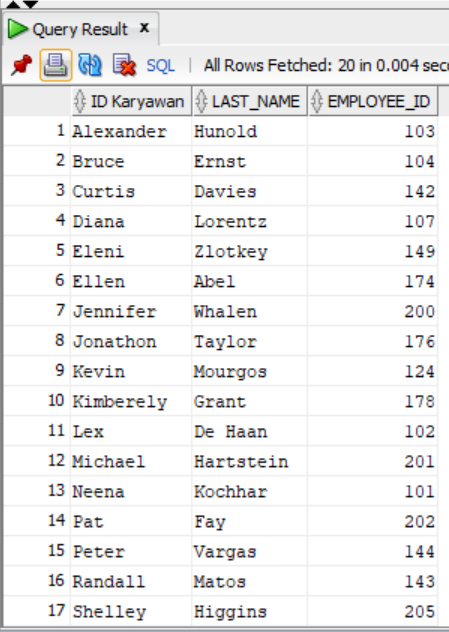


1. Tuliskan SQL Statement yang menampilkan employee\_id, first\_name dan last\_name karyawan. Ganti nama kolom dengan alias yaitu "ID Karyawan". Dengan menggunakan ORDER BY dan column alias, urutkan ID Karyawan dari yang terkecil sampai terbesar.

SELECT first\_name as "ID Karyawan", last\_name, employee\_id

FROM employees

ORDER BY first\_name ASC;

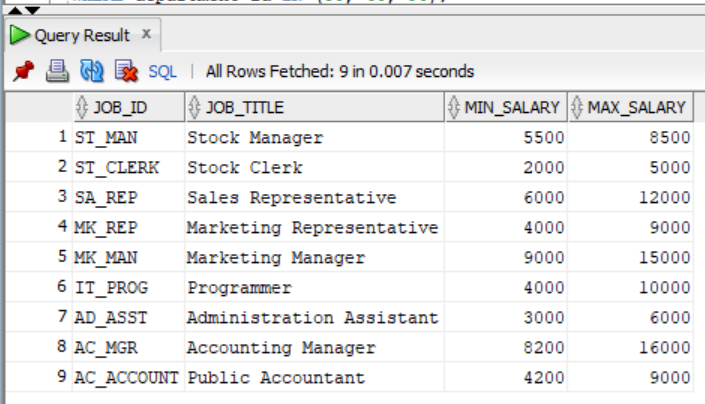


1. Tuliskan SQL Statement yang menampilkan seluruh kolom dalam tabel jobs,yang memiliki minimum salary kurang dari 10000, lalu urutkan job\_id dari Z-A

SELECT \* FROM jobs

WHERE min\_salary < 10000

ORDER BY job\_id DESC;

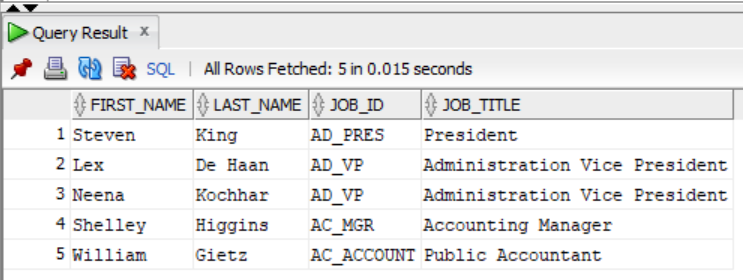


1. Jelaskan makna perintah2 query berikut ini dan bagaimana hasil outputnya:
2. SELECT first\_name, last\_name, job\_id, job\_title

FROM employees NATURAL JOIN jobs

WHERE department\_id > 80;

* Menampilkan first\_name, last\_name, job\_id, dan job\_title yang memiliki department\_id > 80 dari table employees yang di NATURAL JOIN dengan table jobs

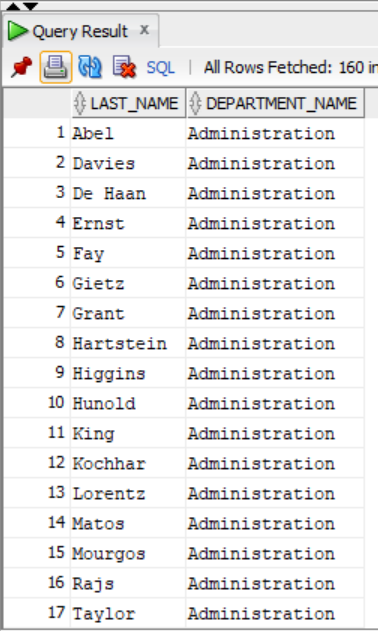


1. SELECT last\_name, department\_name

FROM employees

CROSS JOIN departments;

* Menampilkan last\_name dan department\_name dari table employees yang di CROSS JOIN dengan table departments

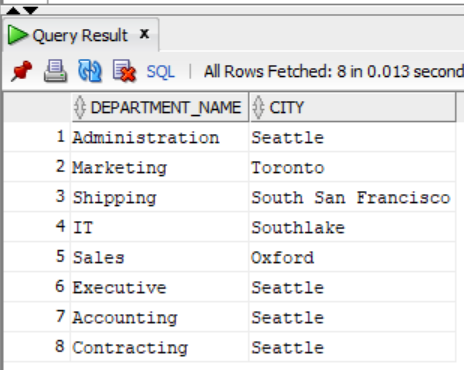
  dst

1. SELECT department\_name, city

FROM departments

NATURAL JOIN locations;

* Menampilkan department\_name dan city dari table departments yang di NATURAL JOIN dengan table locations



1. SELECT first\_name, last\_name, department\_id, department\_name

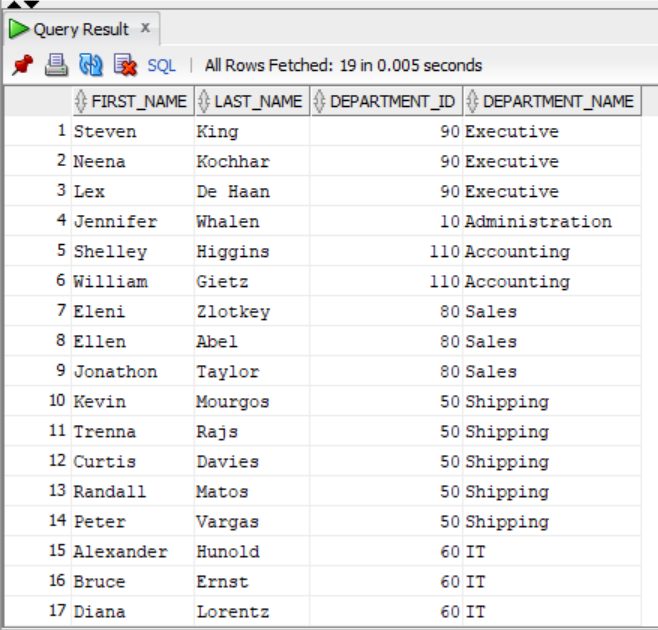
FROM employees

JOIN departments

USING (department\_id);

* Menampilkan first\_name, last\_name, department\_id, department\_name dari table employees yang di

JOIN dengan table departments menggunakan department\_id sebagai foreign key

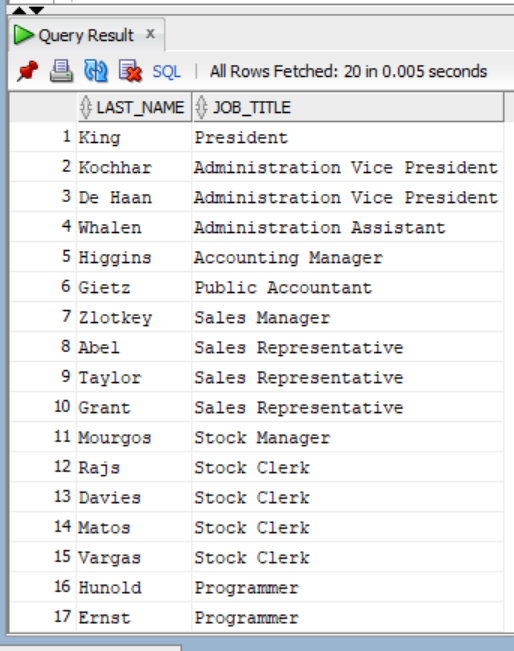


1. SELECT last\_name, job\_title

FROM employees e

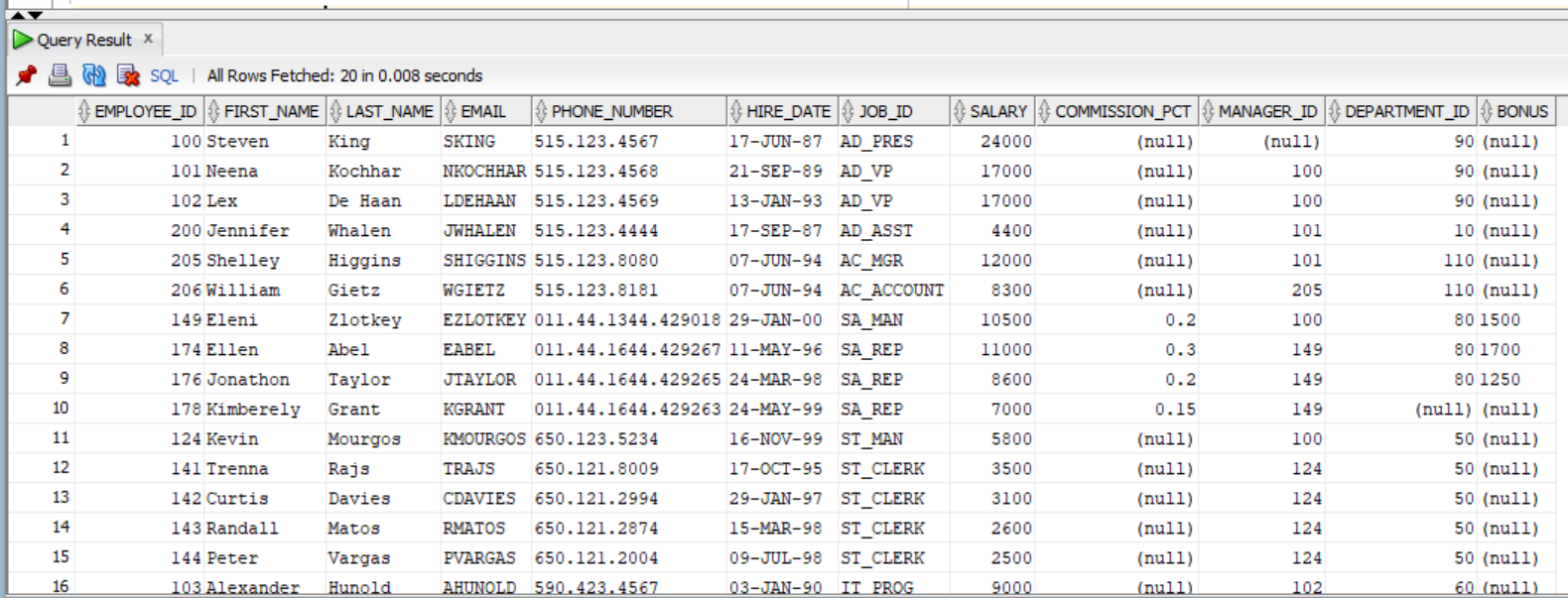
JOIN jobs j ON (e.job\_id = j.job\_id);

* Menampilkan last\_name dan job\_title dari table employees yang diinisialisasikan e, lalu di JOIN dengan table jobs yang diinisialisasikan j dan job\_id pada table employees dan jobs sebagai foreign key



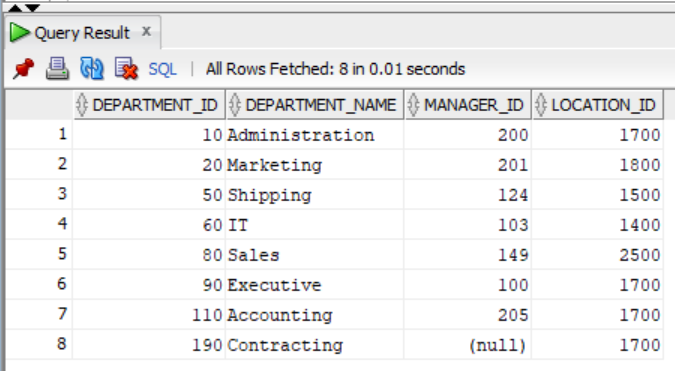
1. SELECT \* FROM employees;

* Menampilkan seluruh column pada table employees



SELECT \* FROM departments;

* Menampilkan seluruh column dari table departments



SELECT last\_name, department\_id, department\_name

FROM employees

NATURAL JOIN departments

WHERE department\_id IN (50, 60, 90);

* Menampilkan last\_name, department\_id, dan department\_name dari table employees yang di

NATURAL JOIN dengan table departments dimana department\_id yang diminta adalah 50, 60, dan 90

