**IF2240 Basis Data**

**Semester II 2020/2021**

Praktikum I : Basic SQL

Dipersiapkan oleh:

Asisten Laboratorium Basis Data

Pelaksanaan:

Waktu Mulai: Jumat, 5 Februari 2021 – 16.00 WIB

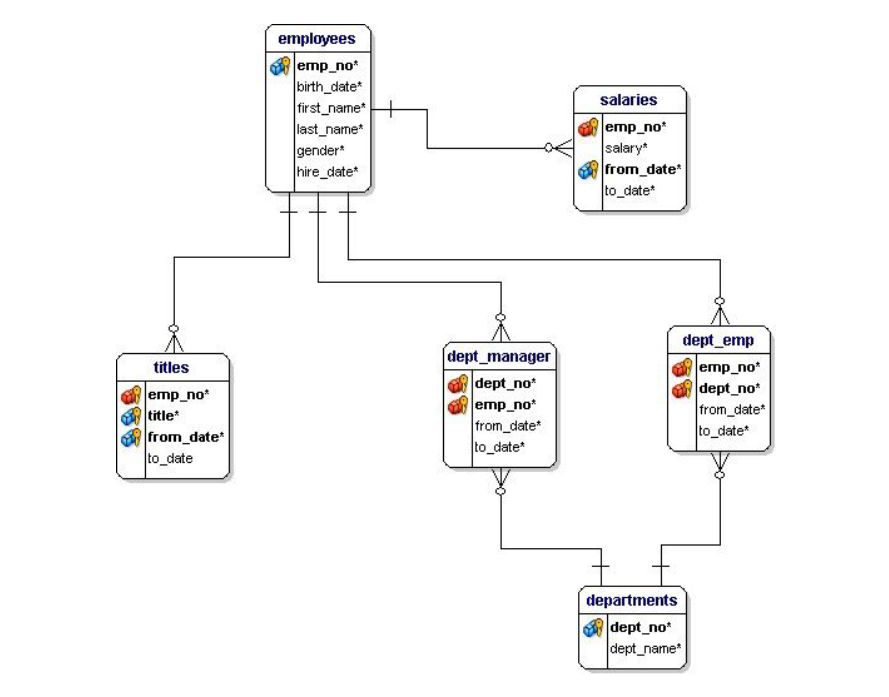
Waktu Selesai: Jumat, 5 Februari 2021 – 18.00 WIB

**I. Deskripsi Sistem**

Async Company melakukan pencatatan terhadap data pegawai-pegawai yang bekerja di perusahaan mereka. Setiap pegawai memiliki data pencatatan mengenai posisi jabatan berikut dengan durasi menjabatnya. Setiap jabatan memiliki gaji dengan nominal yang berbeda-beda, sehingga setiap adanya pergantian jabatan dari seorang pegawai, maka akan dilakukan pencatatan gaji baru. Setiap pegawai terdaftar pada setidaknya satu departemen dan dilakukan juga pencatatan mengenai perpindahan pegawai dari departemen satu ke departemen lainnya. Setiap departemennya, dapat dikepalai lebih dari satu manager.

**II. Skema Basis Data**

Berikut skema relasional basis data berdasarkan deskripsi sistem yang telah dijabarkan pada bagian sebelumnya.



Gambar 1 Skema Basis Data

|  |
| --- |
| employees = (emp\_no, birth\_date, first\_name, last\_name, gender, hire\_date)  salaries = (emp\_no, salary, from\_date, to\_date)  titles = (emp\_no, title, from\_date, to\_date)  dept\_manager = (dept\_no, emp\_no, from\_date, to\_date)  dept\_emp = (emp\_no, dept\_no, from\_date, to\_date)  departments = (dept\_no, dept\_name) |

**III. Permasalahan**

Buatlah basis data dengan nama “AsyncCompany”. Gunakan basis data tersebut, kemudian masukkan (*import*) script SQL Praktikum1.sql yang telah diberikan bersama deskripsi tugas ini.

**IV. Soal**

Lakukan hal-hal berikut.

1. Buatlah queryyang menampilkan nama lengkap (*full\_name*), tanggal lahir, jenis kelamin, dan nama departemen dari manajer perusahaan yang mulai bekerja pada tahun 1985!

**Query**

|  |
| --- |
| SELECT CONCAT(first\_name," ",last\_name) as full\_name, birth\_date, gender, dept\_name from employees NATURAL INNER JOIN dept\_manager NATURAL INNER JOIN departments WHERE Year(hire\_date)=1985; |

**Hasil Query (screenshot)**

|  |
| --- |
|  |

1. Buatlah query untuk menampilkan nama lengkap, title, dan tanggal bergabung pegawai yang bekerja di departemen Quality Management sejak bulan Juli 2002.

**Query**

|  |
| --- |
| WITH emp\_quality(emp\_no) as (SELECT emp\_no FROM dept\_emp NATURAL JOIN departments WHERE dept\_name ="Quality Management" UNION SELECT emp\_no FROM dept\_manager NATURAL JOIN departments WHERE dept\_name ="Quality Management" )  SELECT CONCAT(first\_name," ",last\_name) as full\_name, title, hire\_date FROM employees NATURAL JOIN titles WHERE  emp\_no IN (SELECT emp\_no FROM emp\_quality) AND hire\_date >= "2002-07-01"; |

**Hasil Query (screenshot)**

|  |
| --- |
|  |

1. Buatlah query untuk menampilkan 10 nama pegawai tertua dan tanggal lahirnya yang jenis kelaminnya tidak bernilai null.

**Query**

|  |
| --- |
| SELECT CONCAT(first\_name," ",last\_name) as full\_name, birth\_date FROM employees WHERE gender IS NOT NULL ORDER BY birth\_date asc LIMIT 10; |

**Hasil Query (screenshot)**

|  |
| --- |
|  |

1. Buatlah query yang menampilkan gender dan rata-rata gaji untuk setiap gender

**Query**

|  |
| --- |
| SELECT gender, avg(salary) FROM employees NATURAL JOIN salaries GROUP BY gender; |

**Hasil Query (screenshot)**

|  |
| --- |
|  |

1. Buatlah query untuk memperlihatkan rata-rata gaji saat ini dari pegawai di departemen d009 berdasarkan nama belakangnya! (Hint: gunakan CURDATE() untuk mendapatkan tanggal hari ini)

**Query**

|  |
| --- |
| WITH d009\_emp(emp\_no) as (SELECT emp\_no FROM dept\_emp WHERE dept\_no ="d009" UNION SELECT emp\_no FROM dept\_manager WHERE dept\_no ="d009" )  SELECT last\_name, avg(salary) FROM employees NATURAL JOIN salaries WHERE emp\_no IN (SELECT emp\_no FROM d009\_emp) and to\_date >= CURDATE() GROUP BY last\_name; |

**Hasil Query (screenshot)**

|  |
| --- |
|  |

1. Buatlah query yang menampilkan nama departemen dan jumlah pegawai pada masing-masing departemen yang mulai menjabat di tahun 2000.

**Query**

|  |
| --- |
| WITH emp\_dept(dept\_name, count) as(SELECT dept\_name , count(emp\_no) FROM employees NATURAL INNER JOIN dept\_emp NATURAL JOIN departments WHERE YEAR(hire\_date) >= 2000 GROUP BY dept\_name), emp\_mana(dept\_name, count) as (SELECT dept\_name , count(emp\_no) FROM employees NATURAL INNER JOIN dept\_manager NATURAL JOIN departments WHERE YEAR(hire\_date) >= 2000 GROUP BY dept\_name)  SELECT d.dept\_name , d.count + m.count as count\_pegawai FROM emp\_dept as d , emp\_mana as m WHERE d.dept\_name = m.dept\_name; |

**Hasil Query (screenshot)**

|  |
| --- |
| Niatnya mau nambahin count dari dua tabel yang berbeda tapi gatau caranya maaf kakkk |

1. Buatlah query untuk mencari departemen-departemen yang rata-rata gaji manajernya di atas 50.000. Tampilkan nama departemen dan rata-rata gaji manajer pada departemen tersebut!

**Query**

|  |
| --- |
| WITH salary2(emp\_no,salary) as (SELECT emp\_no, salary FROM salaries)  SELECT dept\_name,AVG(salary) from salary2 NATURAL INNER JOIN dept\_manager NATURAL JOIN departments GROUP BY dept\_name HAVING AVG(salary) > 50000; |

**Hasil Query (screenshot)**

|  |
| --- |
|  |

1. Buatlah sebuah query yang menampilkan 10 baris pertama dari nomor karyawan, nama pertama, tanggal ulang tahun, gaji terbaru, dan diurutkan menurun berdasarkan gaji saat ini, kemudian menaik berdasarkan tanggal lahir!

**Query**

|  |
| --- |
| WITH nowSalary(emp\_no,salary) as (SELECT emp\_no, salary FROM salaries WHERE to\_date >= CURDATE())  SELECT emp\_no, first\_name, birth\_date, salary from employees NATURAL INNER JOIN nowSalary ORDER BY salary DESC, birth\_date ASC LIMIT 10; |

**Hasil Query (screenshot)**

|  |
| --- |
|  |

1. Buatlah query untuk menampilkan nama belakang pegawai non manajer yang memiliki nama belakang yang sama dengan manajer yang memiliki gaji di atas 90.000. CATATAN: Gunakan *set operation.*

**Query**

|  |
| --- |
| WITH salary2(emp\_no,salary) as (SELECT emp\_no, salary FROM salaries), dept\_emp2(emp\_no,dept\_no) as (SELECT emp\_no,dept\_no FROM dept\_emp), dept\_manager2(emp\_no,dept\_no) as (SELECT emp\_no,dept\_no FROM dept\_manager), manager2(emp\_no) as (SELECT emp\_no from dept\_manager2 NATURAL INNER JOIN salary2 WHERE salary >90000)  SELECT last\_name FROM employees NATURAL INNER JOIN dept\_emp WHERE last\_name IN(SELECT last\_name FROM employees NATURAL INNER JOIN manager2); |

**Hasil Query (screenshot)**

|  |
| --- |
|  |

1. Buatlah query untuk menampilkan 10 baris pertama dari semua data karyawan yang saat ini memiliki posisi sebagai engineer dan memiliki gaji kurang dari 50000

**Query**

|  |
| --- |
| WITH salary2(emp\_no,salary) as (SELECT emp\_no, salary FROM salaries), titles2(emp\_no,title) as (SELECT emp\_no, title FROM titles)  SELECT DISTINCT emp\_no,first\_name,last\_name,birth\_date,hire\_date,title,salary FROM employees NATURAL JOIN titles2 NATURAL JOIN salary2 WHERE title ="Engineer" and salary <50000 LIMIT 10; |

**Hasil Query (screenshot)**

|  |
| --- |
|  |

1. Buatlah query untuk menampilkan semua employee dengan title manager yang membawahi departemen dengan nomor d009

**Query**

|  |
| --- |
| WITH d009\_emp(emp\_no) as (SELECT emp\_no FROM dept\_emp WHERE dept\_no ="d009" UNION SELECT emp\_no FROM dept\_manager WHERE dept\_no ="d009" )  SELECT emp\_no, CONCAT(first\_name," ",last\_name) as full\_name FROM employees NATURAL JOIN titles WHERE emp\_no in (SELECT emp\_no FROM d009\_emp) and title ="Manager"; |

**Hasil Query (screenshot)**

|  |
| --- |
|  |

1. Buatlah query untuk memasukkan data baru ke tabel departemen dengan nama departemen Engineer dan nomor departemen d010

**Query**

|  |
| --- |
| INSERT INTO departments(dept\_no,dept\_name) values ("d010","departemen Engineer"); |

**Hasil Query (screenshot)**

|  |
| --- |
|  |