

Extra oefeningen H 2

Gebruik de oplossingen van de extra oefeningen op H 1 als uitgangsbasis daar waar het aangegeven is.

1. Zie H1-oef 5 enkel voor de departementen 50,80 en 90.

```
SELECT department_name, manager_id  
FROM departments  
WHERE department_id in (50,80,90);
```

2. Zie H1-oef 5 enkel waar manager_id groter of gelijk is aan 200.

```
SELECT department_name, manager_id  
FROM departments  
WHERE manager_id >= '200';
```

3. Zie H1-oef 5 enkel voor die rijen waar manager_id onbekend is.

```
SELECT department_name, manager_id  
FROM departments  
WHERE manager_id is null;
```

4. Toon employeenummer, startdatum en einddatum van alle employees die in een eerste of nieuwe afdeling begonnen te werken op 1 januari 1999.

```
SELECT employee_id, start_date, end_date  
FROM job_history  
WHERE start_date = '01-JAN-99';
```

5. Idem vorige oefening maar waarbij de startdatum in 1998 valt en het departement 50 is.

```
SELECT employee_id, start_date, end_date  
FROM job_history  
WHERE start_date LIKE '%98' AND department_id=50;
```

6. Zie H1-oef 6 enkel voor die rijen waarvan het verschil in salaris groter of gelijk is aan 5000 maar kleiner of gelijk is aan 10000. Plaats in volgorde van dit verschil in salaris. Van klein naar groot.

```
SELECT job_id, max_salary - min_salary as "Verschil in salaris"  
FROM jobs  
WHERE (max_salary - min_salary) BETWEEN 5000 AND 10000  
ORDER BY 2;
```

7. Zie H1-oef 7 maar enkel voor die rijen waarin het job_id MAN bevat en voor alle rijen waarvan de job_title op de 2de plaats een a bevat.

```
SELECT job_id || ' is a ' || job_title as functie  
FROM jobs  
WHERE job_id LIKE '%MAN%' OR job_title LIKE '_a%';
```

8. Idem vorige opgave maar moet ook werken ongeacht er hoofd- of kleine letters gebruikt worden.

```
SELECT job_id || ' is a ' || job_title as functie  
FROM jobs  
WHERE UPPER(job_id) LIKE '%MAN%' OR UPPER(job_title) LIKE '_A%';
```

9. Toon de historiek gegevens van alle employees uit een op te geven department_id.

```
SELECT employee_id, job_id, department_id  
FROM job_history  
WHERE department_id = '&departementsnummer';
```

10. Toon naam en voornaam van alle medewerkers die meer dan 20 jaar geleden aangeworven zijn. Plaats de laatst aangeworvene bovenaan.

```
SELECT last_name, first_name, hire_date  
FROM employees  
WHERE hire_date < sysdate - 20*365,25  
ORDER BY 3 desc;
```

11. Toon alle medewerkers, behalve die van afdeling 80 en de mensen met job ST_CLERK van afdeling 50.

```
SELECT last_name, first_name, job_id, department_id
FROM employees
WHERE department_id != 80
AND not (department_id = 50 AND job_id = 'ST_CLERK');
```

12. Toon naam en voornaam (in alfabetische volgorde) van alle medewerkers waarvan de naam begint met een letter tussen C en J.

```
SELECT last_name, first_name, job_id, department_id
FROM employees
WHERE last_name BETWEEN 'C' AND 'J'
ORDER BY last_name, first_name;  of ORDER BY 1,2;
```

13. Geef de verschillende job_id's van de medewerkers.

```
SELECT distinct job_id
FROM employees;
```

14. Geef een lijst waarop de naam van de medewerkers staat, zijn huidig salaris en zijn toekomstig salaris (iedereen zou 10% opslag krijgen). Zorg voor verzorgde attributkoppen.

```
SELECT last_name, salary, salary*1.1 as "toekomstig salaris"
FROM employees;
```