

Interrogation : Requêtes SQL

CORRECTION

Instructions

Répondez aux questions suivantes en utilisant les requêtes SQL appropriées. Certaines questions demandent l'utilisation de jointures entre les tables.

Tables disponibles :

1. Table "Employes" :

id_employe (INT),
nom (VARCHAR),
prenom (VARCHAR),
salaire (DECIMAL),
id_departement (INT)

2. Table "Departements" :

id_departement (INT),
nom_departement (VARCHAR)

3. Table "Projets" :

id_projet (INT),
nom_projet (VARCHAR),
id_employe (INT) — L employe responsable du projet

Questions :

1. Sélectionnez tous les employés ayant un salaire supérieur à 4000.
2. Affichez les noms et prénoms des employés appartenant au département d'identifiant 3.
3. Affichez tous les employés qui n'ont pas d'employé responsable d'un projet (en d'autres termes, ceux qui ne sont pas responsables de projets).
4. Trouvez les employés dont le nom de famille est 'MARTIN', les afficher dans l'ordre alphabétique des prénoms.
5. Affichez le salaire moyen de tous les employés.
6. Récupérez les noms des employés et le nom de leur département.

7. Récupérez les employés responsables de projets ainsi que le nom du projet.

8. Affichez le nom des employés ainsi que le nombre de projets dont ils sont responsables.

9. Affichez les départements qui comptent au moins un employé ayant un salaire supérieur à 5000.

10. Récupérez la liste des départements et le nombre total d'employés dans chaque département.

Correction :

Requêtes sans jointure

1. Sélectionnez tous les employés ayant un salaire supérieur à 4000 :

```
SELECT nom, prenom
FROM Employes
WHERE salaire > 4000;
```

2. Affichez les noms et prénoms des employés appartenant au département d'identifiant 3 :

```
SELECT nom, prenom
FROM Employes
WHERE id_departement = 3;
```

3. Affichez tous les employés qui n'ont pas d'employé responsable d'un projet :

```
SELECT nom, prenom
FROM Employes
WHERE id_employe NOT IN (SELECT id_employe FROM Projets);
```

4. Trouvez les employés dont le nom commence par la lettre "M" :

```
SELECT nom, prenom
FROM Employes
WHERE nom='MARTIN'
ORDER BY prenom ASC;
```

5. Affichez le salaire moyen de tous les employés :

```
SELECT AVG(salaire) AS salaire_moyen
FROM Employes;
```

Requêtes avec jointure

6. Récupérez les noms des employés et le nom de leur département :

```
SELECT e.nom, e.prenom, d.nom_departement
FROM Employes e
JOIN Departements d ON e.id_departement = d.id_departement;
```

7. Récupérez les employés responsables de projets ainsi que le nom du projet :

```
SELECT e.nom, e.prenom, p.nom_projet
FROM Employes e
JOIN Projets p ON e.id_employe = p.id_employe;
```

8. Affichez le nom des employés ainsi que le nombre de projets dont ils sont responsables :

```
SELECT e.nom, e.prenom, COUNT(p.id_projet) AS nombre_projets
FROM Employes e JOIN Projets p ON e.id_employe = p.id_employe
GROUP BY e.nom, e.prenom;
```

9. Affichez les départements qui comptent au moins un employé ayant un salaire supérieur à 5000 :

```
SELECT DISTINCT d.nom_departement
FROM Departements d
JOIN Employes e ON d.id_departement = e.id_departement
WHERE e.salaire > 5000;
```

10. Récupérez la liste des départements et le nombre total d'employés dans chaque département :

```
SELECT d.nom_departement, COUNT(e.id_employe) AS nombre_employes
FROM Departements d JOIN Employes e ON d.id_departement = e.id_departement
GROUP BY d.nom_departement;
```