

TD 5 : SQL – Sélection des données : manipulation des chaînes de caractères et de dates

L'objectif de ce travail dirigé est de vous familiariser avec les fonctions de manipulation des chaînes de caractères et des dates.

Nous utilisons une table nommée employees, dont le MLD est défini comme suite :

- employees(id_emp, nom, prenom, date_embauche, salaire)

Écrire les requêtes SQL pour :

1. Affichez le nom et le prénom de chaque employé en une seule colonne appelée nom_complet.

```
SELECT CONCAT(nom, ' ', prenom) AS nom_complet  
FROM employees;
```

2. Affichez le nom et le prénom de chaque employé dans une seule colonne appelée "nom_complet", avec le nom en majuscules et le prénom en minuscules.

```
SELECT CONCAT(UPPER(nom), ' ', LOWER(prenom)) AS nom_complet  
FROM employees;
```

3. Affichez les noms des employés dont la longueur du nom dépasse 10 caractères.

```
SELECT nom  
FROM employees  
WHERE LENGTH(nom) > 10;
```

4. Affichez le salaire annuel de chaque employé.

```
SELECT id, salaire * 12 AS salaire_annuel  
FROM employees;
```

5. Affichez les noms et la date d'embauche des employés embauchés après le 1er janvier 2020.

```
SELECT nom, date_embauche  
FROM employees  
WHERE date_embauche > '2020-01-01';
```

6. Affichez la durée d'emploi de chaque employé en jours.

```
SELECT id_emp, DATEDIFF(NOW(), date_embauche) AS jours_emploi  
FROM employees;
```

⇒ DATEDIFF permet de calculer le nombre de jours entre deux dates. Le résultat correspond donc à une valeur positive ou négative.

⇒ Exemples :

```
DATEDIFF ("2022-01-27", "2022-01-17") => 10  
DATEDIFF ("2022-01-01", "2022-01-30") => -29
```

7. Affichez les employés embauchés ce mois-ci.

```
SELECT * FROM employees  
WHERE MONTH(date_embauche) = MONTH(NOW()) AND YEAR(date_embauche) =  
YEAR(NOW());
```

8. Affichez les employés embauchés il y a plus de 3 ans.

```
SELECT *  
FROM employees  
WHERE DATEDIFF(NOW(), date_embauche) > 1095;
```

-- 1095 jours correspondent à 3 ans : 365*3

9. Calculer l'ancienneté moyenne des employés en années.

```
SELECT AVG(DATEDIFF(NOW(), date_embauche) / 365) AS anciennete_moyenne  
FROM employees;
```

10. Affichez les employés embauchés entre 2019-01-01 et 2022-01-01.

```
SELECT *  
FROM employees  
WHERE date_embauche BETWEEN '2019-01-01' AND '2022-01-01';
```

11. Affichez les employés embauchés le mardi de cette semaine.

```
SELECT *  
FROM employees  
WHERE YEAR( date_embauche ) = YEAR( NOW( ) )  
AND WEEK( date_embauche ) = WEEK( NOW( ) )  
AND DAYOFWEEK( date_embauche ) =3;
```

Ou

```
SELECT *  
FROM employees  
WHERE YEAR(date_embauche) = YEAR(CURDATE())  
AND WEEK(date_embauche) = WEEK(CURDATE())  
AND DAYOFWEEK(date_embauche) = 3;
```

- CURDATE() : renvoie la date actuelle sans l'heure ou le temps.
- NOW() : renvoie la date et l'heure actuelles.

12. Affichez les années où le nombre d'employés embauchés en décembre dépasse 10 employés.

```
SELECT YEAR(date_embauche) AS annee_embauche, COUNT(*) AS nb_employees  
FROM employees  
WHERE MONTH(date_embauche) = 12  
GROUP BY YEAR(date_embauche)  
HAVING COUNT(*) > 10;
```

Pour plus d'informations sur les fonctions sur les dates :

<https://dev.mysql.com/doc/refman/8.0/en/date-and-time-functions.html>