

Exercice 1 – Vérification de la qualité des emails

Objectif : Identifier les emails manquants ou invalides.

-- Trouver les emails manquants ou invalides

```
SELECT ID_Client, Nom, Prenom, Email  
FROM Clients  
WHERE Email IS NULL OR Email NOT LIKE '%_@___.__%';
```

Questions :

1. Combien d'emails sont invalides ou manquants ?
 2. Quels clients sont concernés ?
-

Exercice 2 – Vérification des dates

Objectif : Déetecter les dates incohérentes (dans le futur).

-- Clients avec date d'inscription dans le futur

```
SELECT ID_Client, Nom, Prenom, Date_Inscription  
FROM Clients  
WHERE Date_Inscription > CURRENT_DATE;
```

Questions :

1. Quels clients ont des dates incohérentes ?
 2. Quelle correction proposeriez-vous pour ces données ?
-

Exercice 3 – Détection des doublons

Objectif : Repérer les doublons sur le couple Nom + Prenom.

-- Clients en doublon

```
SELECT Nom, Prenom, COUNT(*) AS Nb_Records
FROM Clients
GROUP BY Nom, Prenom
HAVING COUNT(*) > 1;
```

Questions :

1. Quels sont les doublons détectés ?
 2. Proposez une requête pour lister tous les enregistrements exacts des doublons.
-

Exercice 4 – Production d'un rapport de conformité

Objectif : Compter les clients totaux, les doublons et les emails invalides.

```
SELECT
    (SELECT COUNT(*) FROM Clients) AS Total_Clients,
    (SELECT COUNT(*)
        SELECT Nom, Prenom
        FROM Clients
        GROUP BY Nom, Prenom
        HAVING COUNT(*) > 1
    ) AS Doublons) AS Nb_Doublons,
    (SELECT COUNT(*) FROM Clients
        WHERE Email IS NULL OR Email NOT LIKE '%_@__%.__%') AS
    Emails_Invalides;
```

Questions :

1. Quels indicateurs de qualité peut-on en tirer ?
 2. Comment ces indicateurs pourraient-ils guider un plan de nettoyage des données ?
-

Exercice 5 – Suivi des modifications

Objectif : Créer une table `Clients_Historique` pour tracer les modifications.

-- Table historique

```
CREATE TABLE Clients_Historique (
    ID_Historique INT PRIMARY KEY,
    ID_Client INT,
    Email VARCHAR(100),
    Statut VARCHAR(20),
    Date_Modification DATETIME
);
```

-- Exemple d'insertion

```
INSERT INTO Clients_Historique (ID_Historique, ID_Client, Email, Statut,
Date_Modification)
VALUES (1, 2, 'claire.martin@example', 'Actif', '2025-12-01 10:00:00');
```

Requête d'application :

-- Historique d'un client

```
SELECT *
FROM Clients_Historique
WHERE ID_Client = 2
ORDER BY Date_Modification DESC;
```

Questions :

1. Pourquoi est-il utile de garder un historique des modifications ?
2. Comment peut-on l'utiliser pour détecter des erreurs ou fraudes dans les données ?