

## Séquence 2 :

**Structuration des requêtes pour l'obtention des données utiles**

Définir les requêtes SQL qui permettent d'obtenir les informations spécifiées.

**Q1.1 - Obtention de la liste des opérateurs (login) à partir de la table Personnel :**

```
SELECT login
FROM personnel
WHERE qualite = 'OPERATEUR';
```

The screenshot shows a SQL query editor with the following text:

```
SELECT login
FROM personnel
WHERE qualite = 'OPERATEUR'
LIMIT 0 , 30
```

Below the query, there is a control panel with the following elements:

- A button labeled "Afficher :" followed by a text input field containing "30" and the text "ligne(s) à partir de la ligne n°" followed by a text input field containing "0".
- A dropdown menu labeled "Trier sur l'index:" with the value "Aucune" selected.
- A button labeled "+ Options" with a table icon.
- A table with two columns: "login" and an empty cell.
- Below the table, there are two rows of actions:
  - Row 1: ☐ Modifier, ☐ Éditer en place, ☐ Copier, ☐ Effacer, op1
  - Row 2: ☐ Modifier, ☐ Éditer en place, ☐ Copier, ☐ Effacer, op2

**Q1.2 - Information permettant de déterminer si un individu (login + password) est un opérateur :**

The screenshot shows a SQL query editor with the following text:

```
SELECT COUNT( * )
FROM personnel
WHERE login = @userLogin
AND PASSWORD = @userPassword
AND qualite = 'OPERATEUR'
LIMIT 0 , 30
```

Below the query, there is a control panel with the following elements:

- A button labeled "Afficher :" followed by a text input field containing "30" and the text "ligne(s) à partir de la ligne n°" followed by a text input field containing "0".
- A dropdown menu labeled "horizontal" with a value "horizontal" selected.
- A button labeled "+ Options" with a table icon.
- A table with two columns: "COUNT(\*)" and "0".
- Below the table, there is a row of actions:
  - Row 1: ☐ Modifier, ☐ Éditer en place, ☐ Copier, ☐ Effacer, 1

```
SET @userPassword = '123abc';
SET @userLogin = 'op1';
SELECT COUNT(*)
FROM personnel
WHERE login = @userLogin AND password = @userPassword AND qualite = 'OPERATEUR';
//(userPassword, userLogin à adapter. Si count = 1 alors y'a un utilisateur, sinon quelque chose est faux.)
```

**Q1.3 - Obtention de la liste des OF non traités :**

```
SELECT id_OrdreFabrication, reference
FROM ordrefabrication
WHERE etat = 'X';
```

```
SELECT id_OrdreFabrication, reference
FROM ordrefabrication
WHERE etat = 'X'
LIMIT 0, 30
```

Afficher :

30

ligne(s) à partir de la ligne n°

0

en mode

horizontal

Options



id\_OrdreFabrication

reference



Modifier



Éditer en place



Copier



Effacer

6

Ref\_OF\_3

**Q1.4 - A partir de la référence d'un OF obtenir la composition de la formule (masse des composants) correspondant à cet OF :**

```
SET @refof= 'REF_OF_1';
```

```
SELECT f.pvcBase, f.plastifiant, f.lubrifiant
FROM formule f
JOIN ordrefabrication of ON f.id_Formule = of.id_Formule
WHERE of.reference = @refof;
```

```
SELECT f.pvcBase, f.plastifiant, f.lubrifiant
FROM formule f
JOIN ordrefabrication of ON f.id_Formule = of.id_Formule
WHERE of.reference = @refof
LIMIT 0, 30
```

[ Modifier ] [ &lt;

Afficher :

30

ligne(s) à partir de la ligne n°

0

en mode

horizontal



et répéter les en-têtes à chaque groupe de

100

Options



pvcBase

plastifiant

lubrifiant



Modifier



Éditer en place



Copier



Effacer

80.00

10.00

10.00

**Q1.5 - Mettre à jour l'indicateur d'état d'un OF (par rapport à sa référence) de "en cours" à "terminé" :**

```
Set @refof= 'Ref_OF_2';
UPDATE ordrefabrication
SET etat = 'T'
WHERE reference = @refof;
```

✓ 1 ligne affectée. ( Traitement en 0.2146 sec )

**UPDATE** ordrefabrication **SET** etat = 'T' **WHERE** reference = 'Ref\_OF\_2'

Exécuter une ou des requêtes SQL sur la base beaujardt\_b: 

```
UPDATE ordrefabrication
SET etat = 'T'
WHERE reference = 'Ref_OF_2';
```

5	Ref_OF_2	2	3.00	T	2007-09-24 23:21:22	4
---	----------	---	------	---	---------------------	---

**Q1.6 - Mettre à jour la date, l'heure actuelle et l'id de l'opérateur (par rapport à son login) ayant réalisé la fabrication dans un OF (par rapport à sa référence) :**

```
SET @refof= 'Ref_OF_3';
SET @userLogin = 'op1';
UPDATE ordrefabrication
SET heurefin = NOW(), id_Personnel = (SELECT id_Personnel FROM personnel WHERE login =
@userLogin)
WHERE reference = @refof;
```

✓ 1 ligne affectée. ( Traitement en 0.1795 sec )

**UPDATE** ordrefabrication **SET** heurefin = NOW( ) ,  
id\_Personnel = ( **SELECT** id\_Personnel  
**FROM** personnel  
**WHERE** login = 'op1' )  
**WHERE** reference = 'Ref\_OF\_3'

6	Ref_OF_3	1	3.00	X	2023-09-22 17:02:38	1
---	----------	---	------	---	---------------------	---

SET @refof= 'Ref\_OF\_3';# MySQL a retourné un résultat vide (aucune ligne).

SET @userLogin = 'op1';# MySQL a retourné un résultat vide (aucune ligne).

```
UPDATE ordrefabrication
SET heurefin = NOW(), id_Personnel = (SELECT id_Personnel FROM personnel WHERE login = @userLogin)
WHERE reference = @refof;# 1 ligne affectée.
```

6	Ref_OF_3	1	3.00	X	2023-10-12 09:11:35	1
---	----------	---	------	---	---------------------	---

//la commande avec les variables fonctionne, mais cela ajoute directement des commentaires au lieu de mettre en couleur. Cela est sûrement normal.