

6

Sous-interrogations

Objectifs

A la fin de ce chapitre, vous pourrez :

- **décrire les types de problème que les sous-interrogations permettent de résoudre**
- **définir des sous-interrogations**
- **énumérer les types de sous-interrogation**
- **écrire des interrogations monolignes et multilignes**

Résoudre un problème à l'aide d'une sous-interrogation

Qui touche un salaire supérieur à celui d'Abel ?

Interrogation principale :



Quels employés touchent un salaire supérieur à celui d'Abel ?

Sous-interrogation :



Quel est le salaire d'Abel ?



Syntaxe des sous-interrogations

```
SELECT  select_list  
FROM    table  
WHERE   expr operator
```

```
(SELECT      select_list  
FROM         table);
```

- La sous-interrogation (interrogation interne) s'exécute une fois avant l'interrogation principale.
- L'interrogation principale (interrogation externe) utilise le résultat de la sous-interrogation.

Utiliser une sous-interrogation

```
SELECT last_name  
FROM   employees 11000  
WHERE  salary >  
      (SELECT salary  
       FROM   employees  
       WHERE  last_name = 'Abel');
```

LAST_NAME
King
Kochhar
De Haan
Hartstein
Higgins

Règles d'utilisation des sous-interrogations

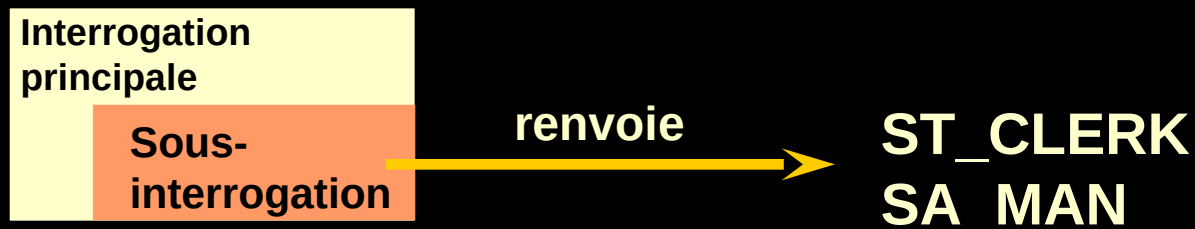
- Placez les sous-interrogations entre parenthèses.
- Placez les sous-interrogations dans la partie droite de la condition de comparaison.
- La clause ORDER BY de la sous-interrogation n'est requise que si vous effectuez une analyse de type n-premiers.
- Utilisez des opérateurs monolignes dans les sous-interrogations monolignes et des opérateurs multilignes dans les sous-interrogations multilignes.

Types de sous-interrogation

- **Sous-interrogation monoligne**



- **Sous-interrogation multiligne**

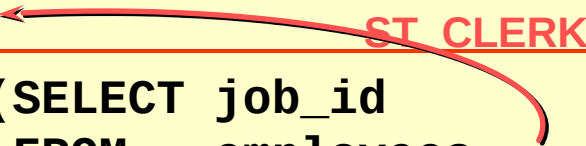
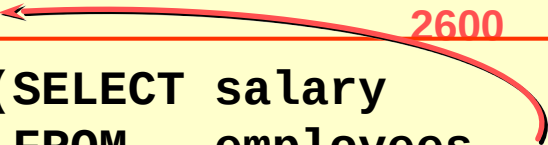


Sous-interrogations monolignes

- Renvoient une seule ligne
- Utilisent des opérateurs de comparaison monolignes

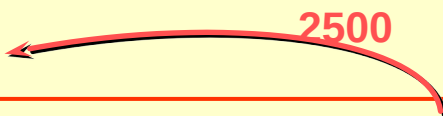
Opérateur	Signification
=	Egal à
>	Supérieur à
>=	Supérieur ou égal à
<	Inférieur à
<=	Inférieur ou égal à
<>	Différent de

Exécuter des sous-interrogations monolignes

```
SELECT last_name, job_id, salary
FROM employees
WHERE job_id =  (SELECT job_id
FROM employees
WHERE employee_id = 141)
AND salary >  (SELECT salary
FROM employees
WHERE employee_id = 143);
```

LAST_NAME	JOB_ID	SALARY
Rajs	ST_CLERK	3500
Davies	ST_CLERK	3100

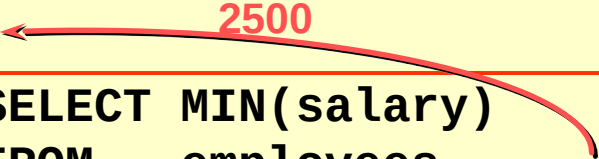
Utiliser des fonctions de groupe dans une sous-interrogation

```
SELECT last_name, job_id, salary
FROM   employees
WHERE  salary = 
           (SELECT MIN(salary)
            FROM   employees);
```

LAST_NAME	JOB_ID	SALARY
Vargas	ST_CLERK	2500

Clause HAVING et sous-interrogations

- Le serveur Oracle exécute d'abord les sous-interrogations.
- Le serveur Oracle renvoie les résultats dans la clause HAVING de l'interrogation principale.

```
SELECT    department_id, MIN(salary)
FROM      employees
GROUP BY  department_id
HAVING    MIN(salary) > 
            (SELECT MIN(salary)
             FROM      employees
             WHERE     department_id = 50);
```

Erreurs dans les sous-interrogations

```
SELECT employee_id, last_name
FROM   employees
WHERE  salary =
      (SELECT  MIN(salary)
       FROM    employees
       GROUP BY department id);
```

```
ERROR at line 4:
ORA-01427: single-row subquery returns more than
one row
```

Opérateur monoligne dans une sous-interrogation multiligne

Problèmes liés aux sous-interrogations

```
SELECT last_name, job_id
FROM   employees
WHERE  job_id =
      (SELECT job_id
       FROM   employees
       WHERE  last_name = 'Haas');
```

no rows selected

La sous-interrogation ne renvoie aucune valeur

Sous-interrogations multilignes

- **Renvoient plusieurs lignes**
- **Utilisent des opérateurs de comparaison multilignes**

Opérateur	Signification
IN	Egal à n'importe quel membre de la liste
ANY	Compare la valeur à chaque valeur renvoyée par la sous-interrogation
ALL	Compare la valeur à toutes les valeurs renvoyées par la sous-interrogation

Utiliser l'opérateur ANY dans les sous-interrogations multilignes

```
SELECT employee_id, last_name, job_id, salary
FROM employees
WHERE salary < ANY (SELECT salary
                     FROM employees
                     WHERE job_id = 'IT_PROG')
AND job_id <> 'IT_PROG';
```

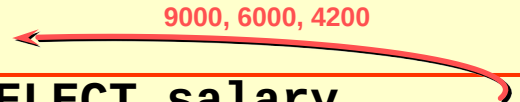
9000, 6000, 4200

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
124	Mourgos	ST_MAN	5800
141	Rajs	ST_CLERK	3500
142	Davies	ST_CLERK	3100
143	Matos	ST_CLERK	2600
144	Vargas	ST_CLERK	2500

10 rows selected.

Utiliser l'opérateur ALL dans les sous-interrogations multilignes

```
SELECT employee_id, last_name, job_id, salary
FROM   employees
WHERE  salary < ALL
      (SELECT salary
       FROM   employees
       WHERE  job_id = 'IT_PROG')
AND    job_id <> 'IT_PROG';
```



9000, 6000, 4200

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
141	Rajs	ST_CLERK	3500
142	Davies	ST_CLERK	3100
143	Matos	ST_CLERK	2600
144	Vargas	ST_CLERK	2500

Valeurs NULL dans une sous-interrogation

```
SELECT emp.last_name  
FROM   employees emp  
WHERE  emp.employee_id NOT IN  
                                (SELECT mgr.manager_id  
                                FROM   employees mgr);
```

no rows selected

Synthèse

Ce chapitre vous a permis d'apprendre à :

- déterminer quand une sous-interrogation peut aider à résoudre un problème
- écrire des sous-interrogations lorsqu'une interrogation est basée sur des valeurs inconnues

```
SELECT  select_list
FROM    table
WHERE   expr operator
        (SELECT select_list
         FROM    table);
```

Présentation de l'exercice 6

Dans cet exercice, vous allez :

- **créer des sous-interrogations pour interroger des valeurs basées sur un critère inconnu,**
- **utiliser des sous-interrogations pour déterminer quelles valeurs existent dans un ensemble de données et pas dans un autre.**