

مبادئ قواعد المعطيات

المحاضرة السابعة

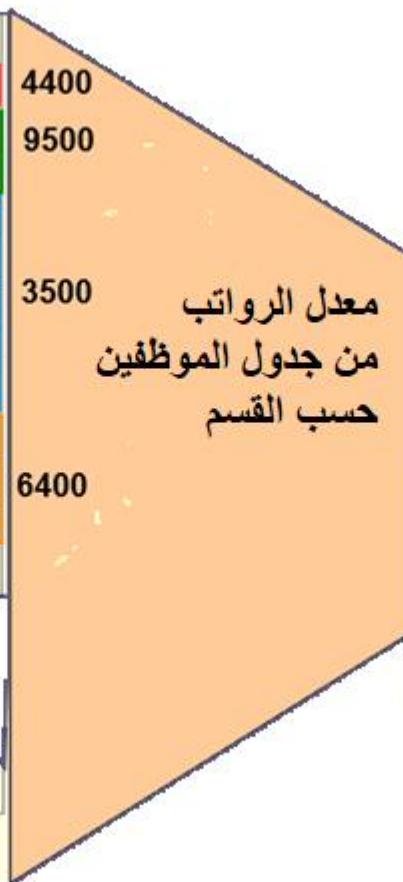
د. سمر الحلبي

قواعد المعطيات关系
لغة الاستفسارات البنوية SQL
(Structured Query Language)
الجزء الثالث

إنشاء مجموعات من المعطيات

جدول الموظفين

DEPARTMENT_ID	SALARY
10	4400
20	13000
20	6000
50	5800
50	3500
50	3100
50	2500
50	2600
60	9000
60	6000
60	4200
80	10500
110	8300
200	7000



DEPARTMENT_ID	Avg(Salary)
10	4400
20	9500
50	3500
60	6400
80	10500

8 rows selected.

20 rows selected.

إنشاء مجموعات من المعطيات

- لإنشاء مجموعات من المعطيات نستخدم عبارة `select` `group by` ضمن عبارة `select`.
- تقوم `group by` بتجمیع المعطیات في الجدول حسب عمود معین أو أكثر (أي تقسم الأسطر في الجدول إلى مجموعات أصغر) وسيتم التجمیع باستخدام احد توابع التجمیع.
- تصبح شکل عبارة `select` بالشكل التالي:

```
SELECT      column, group_function(column)
FROM        table
[WHERE      condition]
[GROUP BY   group_by_expression]
[ORDER BY   column];
```

استخدام group by

- مثال:

الاستفسار عن أرقام الأقسام مع معدل رواتب كل قسم.

```
SELECT department_id, AVG(salary)
FROM employees
GROUP BY department_id;
```

DEPARTMENT_ID	AVG(SALARY)
10	4400
20	9500
50	3500
60	6400
80	10033.3333
90	19333.3333
110	10150
	7000

8 rows selected.

استخدام group by

ملاحظات هامة :

١. كل الأعمدة في قائمة أعمدة select التي ليست من توابع التجميع يجب أن تكون في قسم group by

```
SELECT department_id, AVG(salary)  
FROM employees  
GROUP BY department_id;
```

٢. ليس بالضرورة الأعمدة في القسم select تكون من قائمة أعمدة group by

```
SELECT AVG(salary)  
FROM employees  
GROUP BY department_id;
```

استخدام group by على أكثر من عمود

مثال: الاستفسار عن مجموع رواتب كل قسم من الأقسام وذلك حسب رقم العمل.

```
SELECT      department_id dept_id, job_id, SUM(salary)
FROM        employees
GROUP BY    department_id, job_id;
```

DEPT_ID	JOB_ID	SUM(SALARY)
10	AD_ASST	4400
20	MK_MAN	13000
20	MK_REP	6000
50	ST_CLERK	11700
50	ST_MAN	5800
60	IT_PROG	19200
80	SA_MAN	10500
80	SA_REP	19600
90	AD_PRES	24000
90	AD_VP	24000



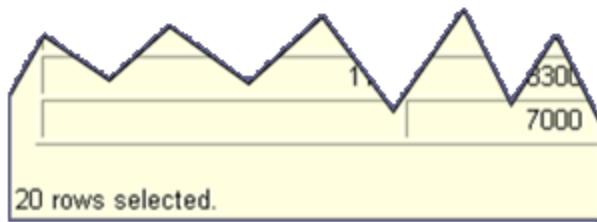
استبعاد بعض المجموعات

جدول الموظفين

DEPARTMENT_ID	SALARY
10	4400
20	13000
20	6000
50	5800
50	3500
50	3100
50	2500
50	2600
60	9000
60	6000
60	4200
80	10500
80	8600

الراتب الأعلى كل قسم
من الأقسام
شرط أن يكون أكبر من
10000

DEPARTMENT_ID	MAX(SALARY)
20	13000
80	11000
90	24000
110	12000



استبعاد بعض المجموعات

- لاستبعاد بعض المجموعات من المعطيات نستخدم القسم **having** لتحديد شرط الاستبعاد وذلك ضمن عبارة **select**.
- تصبح شكل عبارة **select** بالشكل التالي:

```
SELECT      column, group_function  
FROM        table  
[WHERE      condition]  
[GROUP BY   group_by_expression]  
[HAVING    group_condition]  
[ORDER BY   column];
```

استخدام having

- مثال:

الاستفسار عن الراتب الأعلى لكل قسم من الأقسام بشرط أن يكون أكبر من 10000

```
SELECT      department_id, MAX(salary)
FROM        employees
GROUP BY    department_id
HAVING      MAX(salary)>10000;
```

DEPARTMENT_ID	MAX(SALARY)
20	13000
80	11000
90	24000
110	12000

استخدام having

- مثال:

الاستفسار عن مجموع الرواتب حسب العمل بشرط أن يكون أكبر من

١٣٠٠٠

```
SELECT      job_id, SUM(salary) PAYROLL
FROM        employees
WHERE       job_id NOT LIKE '%REP%'
GROUP BY    job_id
HAVING      SUM(salary) > 13000
ORDER BY    SUM(salary);
```

JOB_ID	PAYROLL
IT_PROG	19200
AD_PRES	24000
AD_VP	34000

الاستفسارات الخاطئة التي تستخدم عبارة Group By

```
SELECT department_id, COUNT(last_name)  
FROM employees;
```

Column missing in the GROUP BY clause

```
SELECT department_id, COUNT(last_name)  
      *  
ERROR at line 1:  
ORA-00937: not a single-group group function
```

```
SELECT department_id, AVG(salary)  
FROM employees  
WHERE AVG(salary) > 8000  
GROUP BY department_id;
```

**Cannot use the WHERE clause
to restrict groups**

```
WHERE AVG(salary) > 8000  
      *  
ERROR at line 3:  
ORA-00934: group function is not allowed here
```

5

الاستفسارات الجزئية

استخدام الاستفسارات الجزئية لحل المسائل

Who has a salary greater than Abel's?

Main Query:



Which employees have salaries greater than Abel's salary?



Subquery:



What is Abel's salary?

قواعد كتابة الاستفسارات الجزئية

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

```

- الاستفسار الجزئي (الاستعلام الداخلي (inner query)) ينفذ لمرة واحدة قبل تنفيذ الاستفسار الرئيسي (outer query).
- تستخدم نتيجة الاستفسار الجزئي في الاستفسار الرئيسي.

استخدام الاستفسار الجزئي

• مثال:

الاستفسار عن الاسم الأخير للموظفين ذو الرواتب الأعلى من راتب

الموظف Abel

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

LAST_NAME

King
Kochhar
De Haan
Hartstein
Higgins

إرشادات استخدام الاستفسار الجزئي

- ضع الاستفسار الجزئي ضمن قوسين.
- ضع الاستفسار الجزئي في الطرف الأيمن من عملية المقارنة.
- عبارة `order by` غير ضرورية في الاستفسار الجزئي.

أنواع الاستفسارات الجزئية

- الاستفسارات الجزئية التي تعيد سطراً واحداً.



- الاستفسارات الجزئية التي تعيد أكثر من سطر.



الاستفسارات الجزئية التي تعيد سطراً واحداً

- تعيد سطراً واحداً فقط.
- تستخدم مع هذا النوع من الاستفسارات الجزئية معاملات المقارنة:

Operator	Meaning
=	Equal to
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
<>	Not equal to

تنفيذ الاستفسارات الجزئية التي تعيد سطراً واحداً

- مثال:

الاستفسار عن الاسم الأخير ورقم العمل للموظفين الذين رقم عملهم يساوي رقم عمل الموظف ذو الرقم ١٤١ وراتبه أكبر من راتب الموظف ذو الرقم ١٤٣

```
SELECT last_name, job_id, salary
FROM   employees
WHERE  job_id = ST_CLERK
       (SELECT job_id
        FROM   employees
        WHERE  employee_id = 141)
AND    salary > 2600
       (SELECT salary
        FROM   employees
        WHERE  employee_id = 143);
```

LAST_NAME	JOB_ID	SALARY
Rajs	ST_CLERK	3500
Davies	ST_CLERK	3100

استخدام توابع التجميع في الاستفسارات الجزئية

- مثال:

الاستفسار عن الاسم الأخير ورقم العمل و الراتب للموظفين الذين راتبهم يساوي
الراتب الأدنى لرواتب الموظفين

```
SELECT last_name, job_id, salary
FROM employees
WHERE salary = 2500
      (SELECT MIN(salary)
       FROM employees);
```

LAST_NAME	JOB_ID	SALARY
Vargas	ST_CLERK	2500

استخدام عبارة Having مع الاستفسارات الجزئية

- مثال:

الاستفسار عن رقم القسم والراتب الأدنى لكل قسم شرط أن يكون الراتب الأدنى للقسم أكبر من الراتب الأدنى للقسم ذو الرقم ٥٠

```
SELECT      department_id, MIN(salary)
FROM        employees
GROUP BY    department_id
HAVING      MIN(salary) > 2500
              ↙
              (SELECT MIN(salary)
               FROM employees
               WHERE department_id = 50);
```

الاستفسارات الجزئية التي تعيد أكثر من سطر

- تعيد أكثر من سطر.
- تستخدم مع هذا النوع من الاستفسارات الجزئية عواملات المقارنة التالية:

Operator	Meaning
IN	Equal to any member in the list
ANY	Compare value to each value returned by the subquery
ALL	Compare value to every value returned by the subquery

استخدام عبارة any مع الاستفسارات الجزئية التي تعيد أكثر من سطر

- مثال:

الاستفسار عن أرقام الموظفين و أسمائهم و عملهم مع رواتبهم الذين رواتبهم أصغر من راتب أي موظف في IT_PROG

```
SELECT employee_id, last_name, job_id, salary
FROM   employees      9000, 6000, 4200
WHERE  salary < ANY
       (SELECT salary
        FROM   employees
        WHERE  job_id = 'IT_PROG')
AND    job_id <> 'IT_PROG';
```

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



استخدام عبارة all مع الاستفسارات الجزئية التي تعيد أكثر من سطر

• مثال:

الاستفسار عن أرقام الموظفين و أسمائهم و عملهم مع رواتبهم الذين رواتبهم أصغر من راتب جميع موظفي IT_PROG

```
SELECT employee_id, last_name, job_id, salary
FROM   employees
WHERE  salary < ALL
          ← 9000, 6000, 4200
          ↓
          (SELECT salary
           FROM   employees
           WHERE  job_id = 'IT_PROG')
AND    job_id <> 'IT_PROG';
```

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

بعض الأمثلة الخاطئة

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

Single-row operator with multiple-row subquery

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

```
SELECT last_name, job_id
FROM employees
WHERE job_id =
```

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

no rows selected

Subquery returns no values