الفصل السابع

ترتيب السجلات وتجميعها

في هذا الفصل سوف نتعرف على خيارات جديدة في لغة الاستعلام **SQL Queries** *، للحصول علي ترتيب معين للسجلات أو لتجميعها في مجموعات أو استخراج المجموعات بناء علي شروط معينة .*
بالانتهاء من هذا الفصل ستكتسب المعارف وتتدرب على المهارات التى تجعلك قادرا على:

- *الحصول علي السجلات طبقا لشروط معينة* **Where**
- *ترتيب السجلات* **Order by**
- *تجميع السجلات* **Group by**
- *تجميع السجلات بشروط محددة* **Having**

لكي تعرف عزيزي القارئ الشكل العام لجملة **SELECT** والخيارات العديدة التي يمكنك استخدامها معها يمكنك الدخول إلى **SQL** وعن طريق **Help** سوف تحصل علي الشكل العام لاستخدام جملة **SELECT** وهو مشابه للشكل الآتي :

```
SELECT select_list
[ INTO new_table ]
FROM table_source
[ WHERE search_condition ]
[ GROUP BY group_by_expression ]
[ HAVING search_condition ]
[ ORDER BY order_expression [ ASC | DESC ] ]
```

في هذا الشكل كما في جميع صيغ لغات البرمجة الأخرى فإن الخيارات المحددة بالأقواس[ ] اختيارية بمعني يجوز إضافتها للجملة أو لا يجوز حسب حاجتك إليها . في هذا الفصل سنتعرف علي هذه الخيارات وكيفية استخدامها

استخدام الشرط Where

لعلك تذكر أننا استخدمنا **WHERE** في الفصل الرابع مع **SELECT** وطبعا من أكثر التعليمات استخداما هي **SELECT** وباستخدام **WHERE** معها فاننا نطلب معلومات أكثر تحديدا "مشروطة" من داخل قاعدة البيانات .

مثال :

```
SQL> SELECT * FROM PERSONAL;

FIRSTNAME  LASTNAM  SALARY
medo       jakoub   2500
hoba       yakoub   3200
nona       ahmed    4500
ali        hasan    4500
saad       saeed    5000
gamal      gameel   6000
6 ROWS SELECTED
```

في هذا المثال سوف نبحث عن بيانات الموظفين الذين تزيد مرتباتهم عن **4500** أو تقل عن **3200**

```
SQL> SELECT * FROM PERSONAL
WHERE SALARY > 4500
OR SALARY<3200;

FIRSTNAME  LASTNAME  SALARY
medo       jakoub   2500
saad       saeed    5000
gamal      gameel   6000
4 ROWS SELECTED
```

في المثال التالي سوف نبحث عن بيانات موظف اسمه الأول ali واسمه الأخير hasan

```
SQL> SELECT * FROM PERSONAL
WHERE FIRSTNAME = 'ali'
AND LASTNAME ='hasan';

 FIRSTNAME   LASTNAM   SALARY
 ali         hassan    4500
1 rows selected
```

ترتيب السجلات :

عندما تريد استخراج بيانات كبيرة فيجب أن تكون مرتبة بطريقة يسهل عليك قراءتها ، فإذا كانت بيانات موظفين قد تحتاج أن تكون مرتبة حسب الاسم ، أو حسب المرتب أو النوع أو غير ذلك من الأعمدة ، نستخدم لهذا الغرض الاختيار order by ، وباستخدام هذه الجملة تستطيع تقسيم البيانات الى فئات حسب السن مثلا وداخل الفئات تكون مرتبة حسب المرتب ، والترتيب يمكن أن يكون تصاعديا أو تنازليا .

مثال

```
Mysql> select * from personal
order by age;
```

fname	lastname	gender	age	salary	tel	empdate	gyear
ahmed	mohammad	m	27	3500	2717572	2001	1999
farida	samir	f	28	3300	6734521	2003	1997
israa	mohammad	f	29	4300	2334453	2000	1995
amir	fahmy	m	33	4520	1122334	2002	1985
kareem	ramadan	m	37	5600	8877665	1998	1975
badr	moon	m	43	5400	9876543	1999	1979
medo	Yakoub	m	55	9999	2901346	2001	1971

7 rows in set (0.12 sec)

انظر إلي بيانات حقل age تجدها مرتبة ترتيبا تصاعديا وهو الترتيب التلقائي الذي ترتب به السجلات إذا أردت ترتيب السجلات ترتيبا تنازليا أضف الاختيار DESC ومعناها تنازليا للجملة السابقة .

مثال على الفرز التنازلي descending ، لاحظ أن الفرز على خانة العمر age لهذا فان الفرز التنازلي يبدأ من الأكبر ثم الذي يليه .

```
Mysql> select * from personal
order by age desc;
```

fname	lastname	gender	age	salary	tel	empdate	gyear
medo	Yakoub	m	55	9999	2901346	2001	1971
badr	moon	m	43	5400	9876543	1999	1979
kareem	ramadan	m	37	5600	8877665	1998	1975
amir	fahmy	m	33	4520	1122334	2002	1985
israa	mohammad	f	29	4300	2334453	2000	1995
farida	samir	f	28	3300	6734521	2003	1997
ahmed	mohammad	m	27	3500	2717572	2001	1999

7 rows in set (0.12 sec)

إذا كنت تعرف ترتيب الحقل الذي تريد الترتيب او الفرز على أساسه مثلا الحقل الأول fname في المثال السابق والحقل الثاني lastname يمكنك الفرز باستخدام هذا الترتيب ، نقول مثلا Order by 2 معناها الفرز باستخدام حقل lastname بينما Order by 1 معناها الفرز على أساس حقل fname وهكذا ، مثلا

```
Mysql>select * from personal
order by 2;
```

fname	Lastname	gender	age	salary	tel	empdate	gyear
amir	fahmy	m	33	4520	1122334	2002	1985
ahmed	Mohammad	m	27	3500	2717572	2001	1999

israa	Mohammad	f	29	4300	2334453	2000	1995
badr	moon	m	43	5400	9876543	1999	1979
kareem	ramadan	m	37	5600	8877665	1998	1975
farida	samir	f	28	3300	6734521	2003	1997
medo	Yakoub	m	55	9999	2901346	2001	1971

هنا تم الفرز على أساس الحقل الثاني وهو lastname

تجميع السجلات داخل مجموعات

تعلمنا في الفصل السابق كيف نقوم بعملية الجمع sum لحقل معين ، تعال نتذكر :

Mysql> select * from sectors;

fname	Lastname	sector	salary
amir	fahmy	Training	1500
ahmed	Mohammad	Training	2500
israa	Mohammad	Finance	1200
badr	moon	Admin	1000
kareem	ramadan	Finance	1300
farida	samir	Admin	2000
medo	yakoub	Admin	3300
rania	mokhtar	Training	1500
hayam	helmy	Training	6000
rasha	ali	Finance	5000
hajar	hamed	Admin	2400
hazem	hassan	Finance	4200
khaled	sobhy	Admin	2100
heba	yakoub	Admin	6800

14 rows in set (0.13 sec)

للحصول على مجموع المرتبات استخدم الدالة sum كما تعلمنا في الفصل السابق

```
Mysql> select sum(salary) from sectors;
```

Sum(salary)
40800

ولكننا نريد الحصول على مجموع المرتبات salary لكل قطاع sector على حدة في هذه الحالة يجب استخدام الاختيار Group by ومعناها التجميع لكل مجموعة . انظر المثال التالي :

```
Mysql> select sector, sum(salary)
       From sectors
Group by sector;
```

Sector	sum(salary)
Admin	17600
Finance	11700
Training	11500

ولمعرفة عدد الموظفين في كل قطاع بالإضافة إلي إجمالي مرتبات كل قطاع انظر المثال التالي

```
Mysql> select sector, sum(salary), count(sector)
       From sectors
       Group by sector;
```

sector	sum(salary)	count(sector)
Admin	17600	6
Finance	11700	4
Training	11500	4

تجميع السجلات بشروط محددة :

بإضافة الاختيار Having إلي الاختيار Group by الذي شرحناه في البند السابق يمكن إظهار السجلات التي ينطبق عليها شرط معين من مجموعة السجلات الناتجة من الأمر

SELECT وهي وظيفة مشابهه جدا لوظيفة **WHERE** التي تقوم بتحديد مجموعة من السجلات طبقا لشرط معين من الجدول كله أو من مجموعة الجداول التي تتعامل معها. إذن الفرق الوحيد أن **HAVING** تبحث عن السجلات التي توافق الشرط داخل السجلات الناتجة من جملة **SELECT** أما **WHERE** فتبحث عن السجلات التي توافق الشرط في الجدول كله.

بالرجوع إلي الجدول السابق إذا أردنا تجميع المرتبات لكل قطاع من القطاعات وعدد موظفي كل قطاع ولكن بشرط أن يكون متوسط مرتبات كل قطاع أكبر من **2800** يجب أن نستخدم أمر **SELECT** علي النحو التالي :

```
SELECT  SECTOR , SUM ( SALARY ), COUNT (SECTOR )
FROM SECTORS GROUP BY SECTOR
HAVING AVG (SALARY)>2800;
```

استخدام أكثر من اختيار في جملة Select

يمكن استخدام أكثر ن اختيار في جملة **select** وذلك بهدف التحكم في إظهار السجلات التي ينطبق عليها شروط معينة من مجموعة السجلات الناتجة عن استخدام جملة **select**. سنتعرف هنا على بعض الأمثلة التي تشمل عدة وظائف في نص واحد **combined clauses in one query** ونأمل أن تتابع هذه الأمثلة وتستخلص كيفية عمل عدة اختيارات في جملة واحدة .

المثال التالي يعرض بيانات حقل القطاع **sector** وحقل المرتب **salary** من جدول **emp** بشرط أن يكون اسم القطاع **Admin** أو يبدأ بحرفي **fi** ولا يهم باقي الاسم مع ترتيب السجلات الناتجة تصاعديا طبقا لبيانات حقل **sector** .

```
mysql> select sector, salary from emp
where sector = 'admin'
```

```
or sector like 'fi%'
order by sector;
```

Sector	salary
Admin	1000
Admin	3300
Admin	2400
Admin	2100
Admin	2000
Admin	6800
Finance	1300
Finance	1200
Finance	4200
Finance	5000

generated 12/30/2004 7:12:03 AM by MySQL-Front 1.22

في هذا المثال استخدمنا كل من **where, or, like, orderby**

نريد عرض بيانات القطاعات التي تزيد فيها إجمالي المرتبات عن ١٥٠٠٠

```
mysql> select sector, sum(salary) total
from emp
group by(sector)
having sum(salary) > 15000 ;
```

sector	total
Admin	17600