التطبيقات الرياضية

المحاضرة 12

السنة الرابعة — د. محمد فراس الحلبي — الفصل الثاني

PLUS
LIBRARY
Plus Library
برامكة - حرم كلية العلوم
Mobil: 0944879460 / Phone: 011-2151436

معاملات المقارنة الأخرى :

تتضمن SQL أربعة معاملات أخرى بالإضافة إلى المعاملات التي تم الإشارة إليها في الفقرة السابقة، التي يمكن استخدامها مع جملة where كما هو مبين في الجدول الآتي:

Operator المعامل	Meaning المعنى
Between .... And	بين قيمتين (مجال)
In (List)	تتيح لك اختيار أي قيمة من list
Like { %, _ }	يشابه أو يماثل
Is Null	فارغاً.

المعامل Between :

يمكننا معامل Between من إجراء عمليات البحث داخل نطاقه بين من لقيم، وذلك باستخدامه مع جملة where .

مثال:

إذا أردنا استعادة بيانات جميع الموظفين الذين يتقاضون راتباً مابين 15000 و 50000 فيمكننا القيام بذلك باستخدام العبارة الآتية:

```
SElect First_Name, last_Name, Job_id, Salary from Employees
where Salary Between 15000 And 50000 ;
```


(Mob; 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم – دمشق حرم كلية العلوم Plus

وتكون نتيجة الاستعلام السابق هي الآتية:

FIRST_NAME	LAST_NAME	JOB_ID	SALARY
Steven	King	AD_PRES	24000
Neena	Kochhar	AD_VP	17000
Lex	De Haan	AD_VP	17000

في هذا المثال تم استعمال معامل Between مع جملة Where لإعادة السجلات التي تقع ضمن نطاق معين من القيم.

وكان يمكن استعمال معاملات المقارنة أصغر أكبر على الشكل الآتي:

```
Select First_name,Last_name,Job_id,Salary from EMPLOYEES
where Salary > 15000 And Salary < 50 000;
```

وسنحصل على النتيجة نفسها.

المعامل In :

يمكننا معامل In من القيام بعمليات البحث عن سجل داخل إحدى القوائم، حيث يمكنك استرجاع السجلات التي قيم عمود فيها (محدد في جملة where) تساوي إحدى القيم المحددة ضمن مجموعة.

مثال:

لإظهار جميع السجلات الخاصة بالموظفين ذوي الأرقام 100 أو 110 أو 120 نقوم بكتابة العبارة الآتية:

```
Select * from EMPLOYEES where EMPLOYEE_ID In(100,110,120);
```


(Mob; 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم ـ دمشق حرم كلية العلوم Plus

Select
Fr'av
ول

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	MANAGER_ID	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	DEPARTMENT_ID
100	Steven	King	-	SKING	515.123.4567	06/12/2003	AD_PRES	4000	-	90
110	John	Chen	108	JCHEN	515.124.4269	09/28/2005	FI_ACCOUNT	8200	-	100
120	Matthew	Weiss	100	MWEISS	650.123.1234	07/18/2004	ST_MAN	8000	-	50

في هذا المثال تم استرجاع السجلات التي فيها الحقل Employee_Id يساوي إما 100 أو 110 أو 120، فبدل أن نكتب شرط المقارنة In مرتبطة بالعامل or، حيث كنا الحصول على النتيجة نفسها في المثال السابق باستخدام العامل or على الشكل الآتي:

```
Select * from EMPLOYEES where EMPLOYEE_ID=100 or
EMPLOYEE_ID=110 or EMPLOYEE_ID=120;
```

مثال آخر:

```
Select * from EMPLOYEES where first_name in ('Lex','David','Nancy','John')
```

في هذا المثال سيتم إستعادة جميع السجلات الخاصة بالموظفين الموجودين في القائمة التي تلي معامل in، حيث تم استعادة جميع السجلات حيث يكون First_Name يساوي 'David' أو 'Nancy' أو 'John' أو 'Lex'.

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	MANAGER_ID	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	DEPARTEMENT_ID
102	Lex	De Haan	100	LDEHAAN	515.123.4569	01/13/2001	AD_VP	17000	-	90
105	David	Austin	103	DAUSTIN	590.423.4569	06/25/2005	IT_PROG	4600	-	60
108	Nancy	Greenberg	101	NGREENBERG	515.124.4569	08/17/2002	FI_MGR	12008	-	100
110	John	Chen	108	JCHEN	515.124.4269	09/28/2005	FI_ACCOUNT	8200	-	100

كما أنه عدة اعتبارات لابد من مراعاتها عند استخدام معامل In مع عبارة Where:

(Mob; 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم ـ دمشق حرم كلية العلوم Plus

أ- وضع قيم المجموعة بين قوسين ( ) .

ب- فصل قيم المجموعة عن بعضها بفاصلة , .

ج- يمكن وضع القيم بأي ترتيب .

د- عندما تكون قيم المجموعة حرفية أو تاريخ ، لابد من وضعها بين علامتي تنصيص أحاديات ' ' .

ه- يمكن استخدام Not In لعكس الاختيار كما في الصيغة التالية:

```
Select * from EMPLOYEES where First_name not In('Lex',
'David','Nancy','John') ;
```

في هذه العبارة تم استعادة جميع السجلات من جدول EMPLOYEES عندما يكون First_Name لا يساوي 'David' أو 'Nancy' أو 'Lex' أو 'John' .

* و يمكننا الحصول على النتيجة نفسها إذا استخدمنا العامل And على النحو الآتي:

```
Select * from EMPLOYEES where First_name != 'Lex' And
First_name != 'David' And First_name != 'Nancy And
First_name != 'John' ;
```

إذاً يمكن الاستعاضة عن In بالعامل or

ويمكن الاستعاضة عن not in بالعامل And

العامل Like :

يمكننا استخدام العامل Like مع عبارة where لاستعادة واسترجاع السجلات التي قيم عمود فيها تشابه سلسلة أحرف نبحث عنها. ليس هذا فحسب، بل

(Mob; 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم – دمشق حرم كلية العلوم Plus

مثلاً: لاسترجاع جميع الأسماء التي تبدأ بالحرف (A) يتم كتابة العبارة الآتية

```
SELECT First_name from EMPLOYEES where First_name like 'A%'
```

* % يعني أنه توجد مجموعة أحرف بغض النظر عن عددها وعن حالتها سواء أكبيرة أم صغيرة وذلك مهما كانت هذه الأحرف.

أي أن: % اختصار لتتابع أي سلسلة من أحرف غير معينة مؤلفة من أي قيمة وبأي طول.

* تستخدم like مع الأعمدة التي نوع بياناتها حرفي لذلك نضع ما بعدها ضمن علامتي تنصيص أحاديات.

أما نتيجة عبارة الاستعلام السابقة هي:

FIRST_NAME
Amit
Alexis
Anthony
Adam
Alexander
Alyssa
Alexander
Allan
Alana

مثال آخر: للبحث عن الأسماء التي تشتمل على أحرف ma نكتب العبارة الآتية

Select* from EMPLOYEES where First_name like '%ma%'

* هنا أثناء وضعنا رمز % قبل ma وبعدها فإن هذا يعني أن نظهر جميع الأسماء التي فيها ma سواء أكان ma في أولها أو وسطها أو آخرها.

أي النتائج التي تظهر ستكون على أربعة أنواع:

1- إما أن لا يوجد أي حرف قبل ma، بل تأتي ma مباشرة ثم بعدها أحرف

2- أو أن يوجد حروف قبل ma ولكن لا يوجد حروف بعدها.

3- أو أن لا توجد حروف لا قبل ma ولا بعدها فتكون الاسم هو ma فقط

4- أو أن توجد حروف قبل ma وحروف قبلها وبأعداد متساوية أو غير متساوية.

مثال آخر: لاسترجاع بيانات الموظفين الذين تم تعيينهم خلال عام 2005 نكتب العبارة:

SELECT* from EMPLOYEES where HIRE_DATE like '%2005';

مثال آخر: لاسترجاع جميع الأسماء التي يكون الحرف (a) فيها يأخذ الترتيب الثاني يتم كتابة العبارة الآتية:

Select First_name from EMPLOYEES where First_name Like '_a%';

* الرمز _ اختصار لأي حرف وحيد.

أما نتيجة الاستعلام السابق فهي الآتية:

FIRST_NAME
David
Sarah
David

(Mob; 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم - دمشق حرم كلية العلوم Plus

Laura
Harrison
Nanette
Karen
Daniel
Pat
Tayler
Nancy
Danielle
Vance
Payam
Janette

إذا أردنا أن يكون
الحرف a هو الحرف الثالث
فنضع عبارة الاستعلام ذاكر
السابقة ولكن نضع بعد like
الآتي:
'__a%' like ---
وهكذا...

الحاصل IsNull :

قبل استخدام معامل IsNull مع where لابدّ من الإجابة على ماهي
القيمة Null ؟
تستخدم القيمة Null للإشارة إلى قيمة لا يمكن التعرف على بياناتها، حيث لا تعني
القيمة Null صفر أو مساحة Blank Spaces .

مثال: لاسترجاع جميع بيانات الموظفين الذين لا يتقاضون عمولة Commission
يتم كتابة الكود الآتي:

```
Select * from EMPLOYEES where COMMISSION_PCT is null;
```

إن نتيجة كتابة الكود السابق هو الجدول ذاته الذي كان مكتوباً في المحاضرة

(Mob; 0944-879460 / Pho 011-2151430) خدمات كلية العلوم - دمشق حرم كلية العلوم Plus

الحادية عشرة في العينة السابقة، فلاحظ أنّ كل الموظفين المذكورين في الجدول المذكور لا يتقاضون عوائد

ملاحظة هامة جداً:

يجب الانتباه إلى أن وضع العبارة الآتية

Select * from EMPLOYEES where COMMISSION_PCT= null

خاطئة.

لأنّ null ليست قيمة حسابية فلا يمكن استخدامها مع عمليات المقارنة

لاسترجاع جميع بيانات الموظفين الذين يتقاضون عوائد Commission

يتم كتابة الكود الآتي:

SELECT * from EMPLOYEES where Commission is not null

استخدام المعاملات المنطقية:

بالإضافة إلى المعاملات التي تمّ الإشارة إليها سابقاً، تتضمن SQL ثلاث معاملات منطقية يمكن استخدامها مع جملة where كما هو مبين في الجدول الآتي:

Operator العامل	Meaning المعنى
And	و
Or	أو
Not	النفي

تنبيه: إنّ المعاملات المنطقية و معاملات المقارنة الأخرى هي كلمات محجوزة

(Mob; 0944-879460 / Pho: 011-2151436) Plus خدمات كلية العلوم ـ دمشق حرم كلية العلوم

المعامل and :

يستخدم المعامل and مع where لاستعادة السجلات التي تحقق كافة الشروط المحددة في جملة where ، حيث يعني and أنه لاستعادة السجلات لابد من تحقق الشرط كافة.

مثال: لاستعراض جميع بيانات الموظفين بالقسم IT_Prog الذين يتقاضون راتباً أكبر من 6000 نكتب العبارة الآتية:

```
Select * from EMPLOYEES where Salary>6000 and
                              JOB_ID='IT_PROG';
```

* وضعنا IT_PROG ضمن علامتي اقتباس لأنها أحاديات ولم نضع 6000 ضمن علامتي اقتباس لأن IT_PROG نص أما 6000 رقم.

مثال آخر: لاستعراض جميع بيانات الموظفين بالقسم It_Prog الذين يتقاضون راتباً أقل من 6000 نكتب العبارة الآتية:

```
Select * from EMPLOYEES where Salary<6000 and JOB_ID='IT_PROG';
```

المعامل or :

نستخدم معامل or مع where لاستعادة السجلات التي تحقق شرطاً واحداً من عدة شروط، حيث يعني المعامل or أنه استعادة البيانات تم وفقاً

Mob: 0944-879460 / Pho: 011-2151430) Plus خدمات كلية العلوم - دمشق حرم كلية العلوم

تحقق أي شرط من الشروط.

مثال: لاسترجاع بيانات الموظفين الذين يتقاضون راتباً أكبر من 6000 ليرة أو يعملون بالوظيفة IT_PROG نكتب العبارة الآتية:

```
Select * from EMPLOYEES where Salary >6000
or JOB_ID='IT_PROG';
```

إن نتيجة العبارة السابقة هي ذات النتيجة التي ظهرت في الصفحة السابقة من المحاضرة العاشرة.

ملاحظة: لقد أنهينا عرض بيانات الموظفين الذين يتقاضون راتباً أكبر من 6000 أو يعملون في القسم IT_PROG.

العامل not:

كما رأينا سابقاً، فإن استخدام العامل not يمكننا من استثناء سجلات معينة من اختيارنا، حيث يمكننا العامل not من استرجاع السجلات المعاكسة لاختيارنا.

مثال: لاختيار كافة السجلات من جدول Employees باستثناء التي تكون الـ Commission_PCT لها قيمته Null نكتب العبارة الآتية:

```
SELECT * from EMPLOYEES where COMMISSION_PCT is not null;
```


(Mob: 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم – دمشق حرم كلية العلوم Plus

مثال آخر:

لاسترجاع بيانات الموظفين باستثناء الذين وظيفتهم هي:
Ad_Pres أو Fi_Account أو Pu_Man

نكتب العبارة الآتية:

```
SELECT * From employess
WHere JOB_ID NOT IN ('AD-PRES', 'FI_ACCOUNT',
                     'PU_MAN');
```

* كما وجدنا سابقاً يمكن استخدام معامل Not مع المعاملات الأخرى مثل (Between, like, Null)

مثاليك التالية الآتية:

- Select * from employees
  where salary not between 3000 and 4000

يستعرض جميع حقول جدول الموظفين ولكن لن تعرض السجلات التي يكون راتب موظفيها بين القيمتين 3000 و 4000.

- Select * from EMPLOYEES
  where first_name not like '%t%';

يستعرض جميع حقول جدول الموظفين ولكن لن تعرض السجلات التي يكون الاسم الأول للموظف يحوي على الحرف t.

- Select * fromEMPLOYEES where JOB_ID NOT In ('analyst', 'accountant'

(Mob; 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم - دمشق حرم كلية العلوم Plus

بعرض جميع حقول جدول الموظفين ولكنه لن يعرض سجلات الموظفين الذين مسمى الوظيفة هو analyst أو accountant.

```
Select * from EMPLOYEES where COMMISSION_PCT is not
    null
```

يسترجع جميع حقول جدول EMPLOYEES بشرط أن تكون السجلات التي سيتم عرضها قيمة عمود COMMISSION_PCT عنها ليس فارغاً.

قواعد أسبقية العمليات:

الجدول التالي يوضح قواعد أسبقية العمليات:

Order Evaluated الترتيب	Operator العامل
1	جميع معاملات المقارنة
2	Not
3	and
4	or

وكما تعلمنا سابقاً يمكننا تغيير تسلسل أسبقية العمليات باستخدام الأقواس مثل تأخذ الأقواس ( ) ، الأسبقية في التعابير الحسابية.

مثال:

```
Select FIRST_NAME,JOB_ID,SALARY from EMPLOYEES
where    JOB_ID = 'IT_PROG'
 or      JOB_ID = 'FI_ACCOUNT'
and      SALARY> 3000 ;
```


(Mob; 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم ـ دمشق حرم كلية العلوم Plus

إن الأولوية لعملية المقارنة التي بعد where مباشرة
هي = أولا
ثم يأتي بعدها العامل المنطقي and
ثم العامل المنطقي or

وعليه فإن العبارة السابقة تعني أنه سيتم عرض
الأعمدة first_name و job_id و salary من جدول EMPLOYEES
أما الصفوف التي ستظهر هي التي تحقق أن:
JOB_ID = 'IT_PROG' و SALARY > 3000
أو تحقق أن JOB_ID = 'FI_ACCOUNT'.

وعليه فإن نتيجة العبارة السابقة هي:

FIRST_NAME	JOB_ID	SALARY
Alexander	IT_PROG	9000
Bruce	IT_PROG	6000
David	IT_PROG	4800
Valli	IT_PROG	4800
Diana	IT_PROG	4200
Daniel	FI_ACCOUNT	9000
John	FI_ACCOUNT	8200
Ismael	FI_ACCOUNT	7700
Jose Manuel	FI_ACCOUNT	7800
Luis	FI_ACCOUNT	6900

ملاحظة: نلاحظ أنه من الذين رمز الوظيفي هو IT_PROG لم يظهر إلا
الذين رواتبهم فوق الـ 3000 وذلك بسبب الشرط الذي وضعناه
أما بالنسبة للذين رمز الوظيفي هو FI_ACCOUNT فلم يظهر موظفين رواتبهم
أقل من 3000 لأنهم غير موجودين، وإلا فإنه لو كان يوجد موظف رمزه الوظيفي هو

(Mob; 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم – دمشق حرم كلية العلوم Plus

FI_ACCOUNT و راتبه أقل من 3000 فإنه سوف يظهر في النتيجة وذلك بسبب الشرط الذي كتبناه في عبارة الاستعلام.

أما إذا أردنا عكس هذا الطرف أن كان موجوداً وظيفياً تغيير ترتيب أولوية العمليات وذلك باستخدام الأ

أي علينا أن نكتب العبارة الآتية:

```
Select FIRST_NAME, JOB_ID, SALARY from EMPLOYEES
where ( JOB_ID = 'IT_PROG' or JOB_ID = 'FI_ACCOUNT')
and salary > 3000;
```

استخدام عبارة Distinct لمنع تكرار السجلات:

من أجل منع تكرار قيم الحقول نستخدم عبارة Distinct مباشرة بعد كلمة select.

مثال: إذا أردنا أرقام الأقسام التي يعمل بها الموظفين فيمكننا عمل ذلك بطريقتين:

الطريقة الأولى هي:

```
select deptno from emp;
```

لنفرض أننا نعلم أن أرقام الأقسام هي 10 و 20 و 30.

فإذا كان لدينا 5000 موظف يعملون في القسم 10

و 50000 موظف يعملون في القسم 20

و 100000 موظف يعملون في القسم 30

فإن نتيجة الاستعلام السابق في هذه الحالة هي 155000 سجل

لأن 5000+50000+100000=155000 و أغلبهم "أغلب السجلات" هي بلا فائدة

(Mob; 0944-879460 / Pho: 011-2151436) Plus خدمات كلية العلوم – دمشق حرم كلية العلوم

لأننا فقط نريد معرفة أرقام الأقسام التي يعمل بها الموظفون.

أي أنّ تكرار القسم 10 ، 5000 مرة

وتكرار القسم 20 ، 50 000 مرة

وتكرار القسم 30 ، 100 000 مرة

لن يفيدنا بشيء.

لذلك فإنه من أجل منع تكرار قيم الحقل deptno فإننا سنستخدم:

الطريقة الثانية: وهي: Select Distinct deptno from emp;

إن نتيجة الاستعلام السابق هي

DEPTNO
10
20
30

ونستطيع أن نلاحظ عدم تكرر قيم الحقل deptno.

ترتيب النتائج باستخدام Order by:

يمكن ترتيب السجلات باستخدام OrderBy وفقاً لقيم العمود الذي نحدده ترتيباً تصاعدياً أو تنازلياً.

ويمكن الترتيب بناءً على قيم عمود أو عدة أعمدة.

وعند كتابة جملة Order By يجب اتباع الآتي:

١- تتبع Order By مباشرةً، في حال عدم وجود from{Table Name} الجملة where{condition(s)}

Plus خدمات كلية العلوم ـ دمشق حرم كلية العلوم (Mob; 0944-879460 / Pho: 011-2151436)

- تتبع عملة Order By عملة where{condition} مباشرة.
- يوجد لدينا خياران عند استخدام Order By :
  - Order By{Column Name} Asc : يسمح هذا الخيار بالترتيب التصاعدي
  - Order By{Column Name} Desc : يسمح هذا الخيار بالترتيب التنازلي

- يمكن الترتيب على أساس عدة أعمدة ويجب الفصل بين أسماء الأعمدة بفاصلة (,)
- يوجد لدينا طريقتان لكتابة عملة Order By وهما :
  - عن طريق اسم العمود.
  - عن طريق كتابة رقم العمود المنطقي (موقعة في عملة Select).

الترتيب التصاعدي باستخدام Order By :

مثال :

لاستعادة بيانات جميع الموظفين مرتبة على أساس تاريخ التعيين نكتب العبارة التالية :

```
Select * from EMPLOYEES order by HIRE_DATE asc;
```

وتكون النتيجة هي التالية :

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	MANAGER_ID	E_MAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	DEPARTMENT_ID
102	Lex	De Haan	100	LDEHAAN	515.123.4569	01/13/2001	AD_VP	17000	-	90
205	William	Gietz	205	WGIETZ	515.123.8181	06/07/2002	AC_ACCOUNT	8300	-	110
204	Hermann	Baer	101	HBAER	515.123.8888	06/07/2002	PR_REP	10000	-	70
203	Susan	Mavris	101	SMAVRIS	515.123.7777	06/07/2002	HR_REP	6500	-	40
205	Shelley	Higgins	101	SHIGGINS	515.123.8080	06/07/2002	AC_MGR	12008	-	110
109	Daniel	Faviet	108	DFAVIET	515.124.9469	08/08/2002	FI_ACCOUNT	9000	-	100
108	Nancy	Greenberg	101	NGREENBE	515.124.4569	08/08/2002	FI_MGR	12008	-	100
114	Den	Raphaely	100	DRAPHEAL	515.127.4561	12/07/2002	PU_MAN	11000	-	30
127	Payam	Kaufling	100	PKAUFLIN	515.123.3234	05/01/2003	ST_MAN	7900	-	50
115	Alexander	Khoo	114	AKHOO	515.127.4562	05/18/2003	PU_CLERK	3100	-	30
100	Steven	King	-	SKING	515.123.4567	06/17/2003	AD_PRES	24000	-	90
137	Renske	Ladwig	123	RLADWIG	650.121.1234	07/14/2003	ST_CLERK	3600	-	50
200	Jennifer	Whalen	101	JWHALEN	515.123.4444	09/17/2003	AD_ASST	4400	-	10
141	Trenna	Rajs	124	TRAJS	650.121.8009	10/17/2003	ST_CLERK	3500	-	50
184	Nandita	Sarchand	121	NSARCHAN	650.509.1876	01/27/2004	ST_CLERK	4200	-	50

(Mob; 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم – دمشق حرم كلية العلوم Plus

ونلاحظ أن الترتيب التصاعدي للتاريخ هو من الأقدم للأحدث.

مثال آخر:

Select FIRST_NAME, EMAIL, HIRE_DATE from EMPLOYEES order by HIRE_DATE asc

إن كتابة العبارة السابقة، أو كتابة العبارة:

Select FIRST_NAME, EMAIL, HIRE_DATE from EMPLOYEES order by HIRE_DATE

أو كتابة العبارة:

Select FIRST_NAME, EMAIL, HIRE_DATE from EMPLOYEES order by 3 asc;

أو كتابة العبارة:

select FIRST_NAME, EMAIL, HIRE_DATE from EMPLOYEES order by 3;

ستظهر النتيجة نفسها.

أي هي إظهار الحقول FIRST_NAME و EMAIL و HIRE_DATE أما ترتيب السجلات فسوف يتم حسب HIRE_DATE وذلك تصاعدياً

شرح:

1- في العبارة الأولى مفهوم لماذا النتيجة هي النتيجة السابقة.

2- أما العبارة الثانية فنلاحظ أنها تختلف عن العبارة الأولى بأننا فيها لم نصرح عن نوع الترتيب الذي نريد، إن كنا نريد تصاعدياً Asc أم تنازلياً Desc. وعلما أننا لم نصرح عن نوع الترتيب فإن الترتيب الافتراضي الذي يتم اعتباره هو الترتيب التصاعدي.

3- إن العبارة الثالثة تختلف عن العبارة الأولى بوجود اسم الجدول الذي نريد الترتيب بالنسبة له، ولكن يوجد عوضاً عنه الرقم 3.

والحقيقة: بما أن الجدول HIRE_DATE الذي نريد الترتيب بالنسبة له

(Mob; 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم – دمشق حرم كلية العلوم Plus

ه- الثالث في عبارة Select إذاً عكسنا أن نضع عن رغم الصباح عوضاً عن ذكر اسمه.

ع- إن العبارة الرابعة تختلف عن الثالثة بعد ذكر الترتيب الذي نريد أن يتم أصباط (إن كان تصاعدياً أم تنازلياً) وبالتالي، عما أنه الترتيب لم نذكر إذاً الترتيب الذي سيتم أخذه هو الترتيب التصاعدي.

مثال آخر:

```
Select FIRST_NAME, DEPARTEMENT_ID
       SALARY, HIRE_DATE  from EMPLOYEES
order by SALARY, HIRE_DATE ASC;
```

في هذا المثال سيتم عرض الحقول DEPARTEMENTID، FIRST_NAME، HIRE_DATE، SALARY من جدول EMPLOYEES مرتبة بالترتيب التصاعدي حسب قيم الحقلين HIRE_DATE، SALARY

الترتيب التنازلي باستخدام order by:

مثال: لاستعادة بيانات جميع الموظفين مرتبة تنازلياً على أساس تاريخ التعيين، نكتب العبارة الآتية:

```
Select * from EMPLOYEES order by HIRE_DATE desc;
```

ملاحظات:

أ- الترتيب الافتراضي في عملة orderby هو الترتيب التصاعدي عالم نضع عكس ذلك بالكلمة Desc.

(Mob; 0944-879460 / Pho; 011-2151436)

- يتم وضع القيم Null متأخراً عند الترتيب التصاعدي، والعكس
  يتم وضع القيم Null أولاً عند الترتيب التنازلي.
- يمكنك إجراء الترتيب - باستخدام order by - على أساس قيم أكثر من عمود، كما في المثال الآتي:

```
Select FIRST_NAME, EMAIL from EMPLOYEES
order by HIRE_DATE Desc;
```

وكما في المثال الآتي أيضاً:

```
Select FIRST_NAME, SALARY from EMPLOYEES
order by HIRE_DATE, SALARY Desc;
```

- يمكننا إجراء الترتيب باستخدام order by على أساس قيم أكثر من عمود

الترتيب باستخدام A column Alias

- يمكننا إجراء الترتيب في جملة order by باستخدام A column Alias كما في المثال الآتي:

```
Select FIRST_NAME, salary*12 as total
from EMPLOYEES
order by total Desc;
```

سيتم عرض الحقل FIRST_NAME و "salary*12 الذي سميناه بـ "total من الجدول EMPLOYEES مرتبة تنازلياً حسب قيم الحقل total "الذي يحوي قيم salary*12"

(Mob; 0944-879460 / Pho: 011-2151436) خدمات كلية العلوم – دمشق حرم كلية العلوم Plus

أعد استعلام سيكون على الشكل:

FIRST_NAME	TOTAL
Steven	288000
Neena	204000
Lex	204000
John	168000
Karen	162000
Michael	156000
Nancy	144096
Shelley	144096
Alberto	144000
Lisa	138000
Den	132000
Gerald	132000
Ellen	132000
Elena	126000
Clara	126000

النتيجة كالتالي: