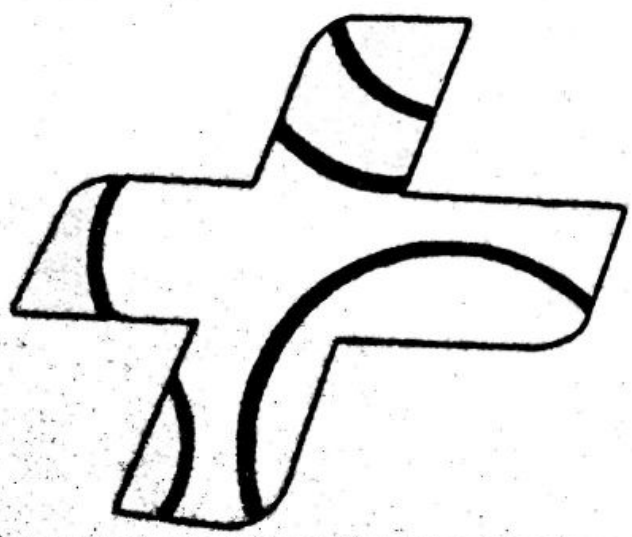
الرياضيات (التطبيقية)

التطبيقات الرياضية

المحاضرة 13

السنة الرابعة — د. محمد فراس الحلبي — الفصل الثاني



PLUS LIBRARY

Plus Library

برامكة - حرم كلية العلوم

Mobil: 0944879460 / Phone: 011-2151436

القيمة Null وتأثيراتها:

تستخدم القيمة Null للإشارة إلى عمود لا يحتوي على بيانات، بمعنى وجود قيم فارغة داخل الحقول.

ومن أشهر عيوب القيمة Null صعوبة إجراء أي عمليات حسابية على سجلات تحتوي بعضها على قيم فارغة.

استخدام القيمة Null:

أ- تستخدم القيمة Null للإشارة إلى عمود لا يحتوي على بيانات، ولا تعني القيمة Null الصفر، أو المسافة الفارغة Blank Space

ب- عند استخدام القيمة Null في التعابير الحسابية يكون الناتج Null.

مثلاً: إذا كتبنا العبارة الآتية:

```
Select FIRST_NAME, SALARY, COMMISSION_PCT
From EMPLOYEES;
```

ستظهر النتيجة الآتية و التي تدل على أن العمود COMMISSION_PCT يضم قيماً من النوع Null.

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

FIRST_NAME	SALARY	COMMISSION_PCT
steven	24000	-
Neena	17000	-
lex	17000	-
Alexander	9000	-
Bruce	6000	-
David	4800	-
Valli	4800	-
Diana	4200	-
Nancy	12008	-
Daniel	9000	-
John	8200	-
Ismael	7700	-
Jose Manuel	7800	-
Luis	6900	-
Den	11000	-

TUS

مثال آخر: إذا كتبنا العبارة الآتية:

```
select FIRST_NAME,SALARY, COMMISSION_PCT ,
       SALARY+ COMMISSION_PCT
from EMPLOYEES
where FIRST_NAME = 'Lex' ;
```

فإن النتيجة ستكون هي:

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

FIRST_NAME SALARY COMMISSION_PCT SALARY+COMMISSION_PCT

Lex 17000 - -

في هذا المثال تم تناول العمليات الحسابية على القيم Null حيث إننا أخذنا الـ Commission_pct الخاص بالموظف Lex وهي القيم Null ،

وكما تعلمنا فإن ناتج أي عملية حسابية طرفاها القيمة Null يكون Null أما إذا كان لدينا موظف ما يأخذ عمولة commission أي قيمة الحقل Commission_Pct عنده ليست null فإن قيمة الحقل SALARY+COMMISSION_PCT عند هذا الموظف لن تكون null .

العلاقات

الربط بين جدولين أو أكثر

رأينا كيفية استرجاع البيانات من جدول واحد، ولكننا كثيراً ما نحتاج إلى استرجاع البيانات من جدولين أو أكثر في استعلام واحد Query .

إن إجراء مثل هذا يتطلب منا إنشاء علاقة بين جدولين أو أكثر من جدول.

إن القوة الحقيقية لقاعدة البيانات العلائقية تكمن في قدرتها على استرجاع سجلات من جداول مختلفة في استعلام واحد توجد لدينا العديد من الأسباب التي تدعونا إلى إنشاء

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

علاقة بين الجداول ، منها :

١- دمج الأعمدة من جدولين أو أكثر.

٢- اختيار أعمدة موجودة في جدول واحد بناءً على شرط يطبق على عمود آخر (self join) (الربط الذاتي).

استرجاع بيانات من جدولين :

عند استرجاع بيانات من جدولين، هناك عدة اعتبارات يجب مراعاتها :

١- وضع أسماء الأعمدة في جملة select ، على أن يتم الفصل بينها بفاصلة.

٢- وضع أسماء الجداول في جملة From ، على أن يتم الفصل بينها بفاصلة.

٣- إضافة شرط الربط في جملة where.

٤- الصيغة العامة لاسترجاع بيانات من جدولين هي :

```
Select    table1.column , table2.column
From      table1, table2
where     table1.column1 = table2.column2
```

مثال :

```
select    Emp.Empno , Emp.Ename, Dept.Dname
from      Emp , Dept
where     Emp.Deptno = Dept.Deptno ;
```


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

في هذا المثال تمّ استرجاع بيانات من جدولي Dept, Emp
حيث إن الأعمدة التي سيتم عرضها هي:

* Empno من جدول Emp
* Ename من جدول Emp
* Dname من جدول Dept

وعرفنا ذلك لأنها موجودة في جملة select.

أما الجداول مصدر البيانات فهي Emp و Dept وهي موجودة في جملة From.

وشرط الربط موجود في جملة where.

ونلاحظ كيف تمّت كتابة اسم الجدول قبل اسم العمود مع الفصل بينهما بنقطة وذلك في كل من جملة select وجملة where.

وفي الحقيقة إن فعل هذا الأمر غير ضروري إلا في حال كان العمود (الوارد في جملة select أو الوارد في جملة where) يشترك بالاسم مع عمود في جدول آخر.

أي أنّ كتابة العبارة الآتية

```
select Empno, Ename, Dname
from Emp, Dept
where Emp.Deptno = Dept.Deptno ;
```

صحيحة وذلك على فرض أنّ الأعمدة Empno و Ename ليست موجودة في الجدول Dept

وعلى فرض أيضاً أن العمود Dname غير موجود في الجدول Emp

أما إذا كان العمود Empno موجوداً في الجدول Dept بالإضافة إلى وجوده في الجدول Emp فستتبع العبارة السابقة خاطئة، ولتصحيحها

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

لا بدّ من وضع اسم الجدول المراد (Dept أم Emp) ثم نقطة ثم اسم العمود (Empno) * وكذلك الأمر بالنسبة لباقي الأعمدة.

أما كتابة العبارة الآتية:

```
Select  Emp.Empno, Emp.Ename, Dept.Dname
From    Emp, Dept
Where   Deptno = Deptno ;
```

في حالة:

فنلاحظ أن أعمدة الربط لها الاسم Deptno نفسه مما يعني أن العمود Deptno موجود في كلا الجدولين Emp و Dept.

وبالتالي فإن هذا سيسبب مشاكل مع نظام إدارة قواعد البيانات DBMS حيث لا يعرف لأي جدول ينتمي العمود Deptno الموجود قبل إشارة المساواة ولا يعرف لأي جدول ينتمي العمود Deptno الموجود بعد إشارة المساواة.

أما إذا كان العمود Deptno موجود في الجدول Emp بالاسم Dnom مثلاً "أي غير الاسم" فيمكننا عندها وضع اسم الجدول أي يمكننا عمل الآتي:

... where Dnom = Deptno ;

استخدام أكثر من شرط في جملة where:

عند استرجاع بيانات من أكثر من جدول، يمكننا إضافة أكثر من شرط في جملة الشرط بعد شرط الربط

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

مثلاً: لدينا جدول بيانات يحتوي على الموظفين معرف، اسم، الوظيفة
، رقم القسم، موقع القسم
لابد من إضافة شرط في عبارة where كما يأتي:

```
Select Empno, Ename, Emp.Job, Emp.Deptno, Loc
From Emp, Dept
where Emp.Deptno=Dept.Deptno And Job='Analyst';
```

- إن ما تم استرجاعه من بيانات موجود في جدولين Emp و Dept
وعرفنا ذلك من خلال كلمة From.
- أما الأعمدة التي تم استرجاعها:
Loc و Deptno و Job و Ename و Empno
* العمودان Empno و Ename تم استرجاعها من جدول Emp وضوحاً
ولم يصرح عن ذلك ضمن عبارة SQL، لأن العمودان لا يوجدان
إلا في جدول Emp.
* العمودان Job و Deptno تم استرجاعها من جدول Emp
وقد تم التعبير عن ذلك صراحة ضمن عبارة SQL من خلال كتابة اسم الجدول
Emp ثم نقطة ثم اسم العمود، وذلك في كل منهما.
* العمود Loc تم استرجاعه من جدول Dept وضوحاً.
ولم يصرح عن ذلك ضمن عبارة SQL، لأن هذا العمود غير موجود إلا
في جدول Dept.
- أما شرطي الربط فهما:
أ- أن يكون رقم القسم الذي يعمل به الموظف موافقاً لأحد أرقام الأقسام
الموجودة في جدول الأقسام وقد تم التعبير عن ذلك بكتابة

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

Emp.Deptno = Dept.Deptno في جملة where .

رقم القسم في جدول Emp رقم القسم في جدول Dept

ـ إن الشرط لربط الجداول هو أن يكون عمود الوصل موجود في الجدول وقد تم التعبير عن ذلك بكتابة:

Job='Analyst' في جملة where .

ونلاحظ أنه تم الربط بين شرطي الربط بـ And .

لم نضع قبل اسم الحقل Job اسم الجدول الموجود به، وذلك لأننا نعلم مسبقاً أن الحقل Job لا يوجد إلا في الجدول Emp

استخدام Table Alias في جملة where :

كما سبق وتعلمنا أنه يمكن استخدام اسم اعتباري للعمود (الحقل)، فيمكننا أيضاً وضع اسم اعتباري للجدول.

مثال: إذا أردنا كتابة عبارة SQL السابقة، باستخدام الأسماء الاعتبارية للجداول Table Aliases، سوف تصبح العبارة بالشكل:

```
Select E.Empno, E.Ename, E.Deptno, D.Deptno
      ,DoLoc
From Emp E , Dept D
where E.Deptno = D.Deptno ;
```


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

ملاحظة: الاسم الاعتباري للجدول تمّ وضعه في عبارة from وذلك بعد اسم الجدول الأصلي وقد فُصِلَ بين الاسم الاعتباري للجدول وبين الاسم الأصلي للجدول بفراغ.

من أجل تسمية جدول ما باسم اعتباري: لا يصح أن نضع as بين اسم الجدول الاعتباري واسم الجدول الأصلي. بل عن مجبرين على أن نفصل بينهما بفراغ فقط.

هناك عدة اعتبارات يجب مراعاتها عند استخدام الأسماء الاعتبارية:

1. الاسم الاعتباري للجدول لا يزيد عن 30 حرفاً.
2. لا بدّ أن يستخدم الاسم الاعتباري أولاً في جملة where.
3. يفضّل أن يكون الاسم الاعتباري معبراً.

استرجاع بيانات من عدة جداول:

في بعض الأحيان نريد استرجاع بيانات من أكثر من جدول.

على سبيل المثال:

نريد استرجاع بيانات: اسم الزبون، رقم الطلبية، الأصناف، إجمالي الأصناف، إجمالي الطلبيات للعميل (للزبون) Tarek.

إن هذه البيانات يمكن استرجاعها من ثلاثة جداول هي:

Customer, Ord, Item

(زبون (عميل)) (طلبية) (صنف (مادة))

نذهب بكتابة العبارة الآتية:

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

```
Select C.Name, O.ordid, I.Item, I.Itemtot,
       O.Total
From   Customer C, Ord O, Item I
where  C.Custid = O.Custid
   and O.Ordid = I.Ordid
   and C.Name = 'Tarek';
```

لقد قمنا في عبارة SQL السابقة باسترجاع البيانات التالية:

- الحقل Name من الجدول Customer والذي يعني به اسم الزبون.
- الحقل ordid من الجدول Ord والذي يعني به رقم الطلبية.
- الحقل Item من الجدول Item والذي يعني به الصنف (المادة).
- الحقل Itemtot من الجدول Item والذي يعني به الإجمالي للأصناف.
- الحقل Total من الجدول Ord والذي يعني به الإجمالي للطلبيات.

أما في جملة From فقد حددنا من أسماء الجداول التي نريد استعادة بيانات الحقول منها و وضعنا لكل منها اسم اعتباري وذلك من أجل تسهيل الكتابة لاحقاً.

وفي جملة where قمنا بأمرين أساسيين وهما:

- ربط الجداول الثلاث مع بعضها وذلك من خلال عملية التقاطع المرجعي:

يجب أن يكون المفتاح الأساسي Custid في جدول Customer مساوياً للمفتاح الغريب Custid في جدول Ord.

ويجب أن يكون المفتاح الأساسي في جدول Ord والذي هو ordid مساوياً للمفتاح الغريب في جدول Item.

أي أننا ربطنا الجدول Customer مع الجدول Ord ثم ربطنا جدول Ord مع الجدول Item.

- أن يكون العميل الذي نسترجع بياناته هو العميل Tarek (هكذا نريد).

ولذلك وضعنا C.Name = 'Tarek' (وضعناها بين علامتي اقتباس لأنها char).

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

أنواع الروابط :

كما رأينا في بعض الأحيان نرغب بعرض بيانات من أكثر من جدول، من أجل ذلك، يجب علينا إجراء عملية ربط بين جدولين.
سنقوم بدراسة أنواع مختلفة من الربط في الفقرات الآتية :

١- الرابطة المتكافئة : Equijoin
لابدّ أن تحقق كل سجل في الجدول الأب يقابله سجل أو سجلات في الجدول الابن المرتبط به.

٢- الرابطة غير المتكافئة : Non equijoin
لاحتمال الجداول أعمدة وقيم مشتركة.

٣- الرابطة الخارجية : Outer join
تحتمل الجداول المرتبطة أعمدة مشتركة، ولكن لسجلات المرتبطة يمكن أن تضم قيماً مشتركة ويمكن ألا تضم ذلك.
بمعنى أنّ الجدول الرئيس الأب يضم سجلاً أو سجلات لا يقابلها سجل أو سجلات في الجدول الابن المرتبط به.

٤- الرابطة الذاتية : Self join
يمكننا عمل رابطة ذاتية عند ربط الجدول مع نفسه.

٥- الرابطة الديكارتية : Cartesian join
للحصول على الرابطة الديكارتية نقوم بحذف شرط الربط بين الجدولين في جملة where، عندها سيقوم نظام إدارة قواعد البيانات DBMS بربط كل سجل من الجدول الأول مع سجل من الجدول الثاني.

الرابطة المتكافئة :

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

مثلاً: إذا أردنا تحديد اسم الإدارة (القسم) لكل موظف فاعلينا سوى المقارنة بين قيم deptno في كل من الجدولين emp و dept وفق الصيغة الآتية:

```
Select emp.empno, emp.ename, emp.deptno,
       dept.deptno, dept.loc
From emp, dept
where emp.deptno = dept.deptno ;
```

وهنا يكون الربط كما فعلنا، لكون كل سجل من جدول الموظفين emp يرتبط بسجل واحد أو أكثر من جدول الأقسام dept.

مع الملاحظة أنّ:

1- في هذا المثال بفرض أنّ جدول Dept لا يوجد به إدارة ليس لها عاملاً في جدول Emp فني هذه الحالة فإن الاستعلام السابق لن يقوم باستعراض البيانات لكافة الأقسام.

2- بما أنّ عمود deptno له الاسم نفسه في كلا الجدولين، فلابدّ من وضع اسم الجدول قبل اسم العمود مع الوصل بينهما بنقطة.

3- يطلق على العلاقة Equijoin أيضاً Simple Join (الرابطة البسيطة) أو Inner Join (الرابطة الداخلية).

الرابطة غير المتكافئة:

نادراً ما نحتاج إلى ربط سجلات من جدولين لا يمتلكان أعمدة ولا قيماً مشتركة. إنّ مثل تلك الرابطة تسمى رابطة غير متكافئة.

مثلاً: لنفرض أن لدينا جدولاً للموظفين فيه أرقامهم، أسماؤهم ورواتبهم.

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

وجدولة آخر بين رتبة الراتب حيث كل رتبة فيها أقل حد للراتب وأكبر حد للراتب.

Emp

EMPNO	ENAME	SAL
1	tarek	40000
2	ayman	28500
3	said	24500
4	ali	29980
5	ahmed	13500
6	helal	17000
7	ebrahim	16000

Salgrade

GRADE	LOSAL	HISAL
1	8000	13000
2	13010	14000
3	14010	25000
4	25000	99990

نريد أردنا أن نعلم رواتب الموظفين ودرجات رواتبهم ، نكتب العبارة الآتية

```
Select  Emp.Ename, Emp.Sal, Salgrade.Grade
From    Emp, Salgrade
where   Emp.Sal Between Salgrade.Losal and Salgrade.HIsal;
```

في الحقيقة كتبنا في جملة where العبارة السابقة لأنه:

الربط بين الجدولين السابقين Emp و Salgrade لا يمكن أن يتم من خلال مفتاح أساسي و مفتاح غريب (لأنه لا يوجد عمودين لهما الاسم نفسه أحدهما في جدول والآخر في جدول مختلف)

وأيضاً لا يمكن أن يتم من خلال أن تكون قيمة حقل ما في الجدول Emp معتمدة على قيمة حقل آخر في الجدول Salgrade (فمثلاً لتحديد رتبة الراتب لا يكفي أن ننظر إلى أقل راتب في تلك الرتبة أو إلى أكبر راتب في تلك الرتبة بل لابد من النظر إلى قيمة الحقلين معاً)

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

فمثلاً هل يكون راتب الموظف 1 من المرتبة 1 يجب أن يحقق رابطة أنه محصور بين 8000 و 13000.

وهل يكون راتب الموظف 1 من المرتبة 2 يجب أن يكون راتبه يحقق أنه محصور بين 13010 و 14000 . وهكذا ...

وبالتالي بأنه هل يكون راتب موظف ما من مرتبة ما فيجب أن يكون راتبه محققاً لكونه محصوراً بين أقل راتب في هذه المرتبة وأكبر راتب في هذه المرتبة.

أي أنه يجب أن يحقق الشرط

Emp.Sal Between Salgrade.Losal and Salgrad.Hisal

وهذا ما كان مكافئ لعبارة SQL .

الرابطة الخارجية :

تمتلك الجداول المرتبطة أعمدة مشتركة يوافق السجلات في الجداول المرتبطة يمكن أن تضم قيماً مشتركة ويمكن الانضم.

مثلاً : إذا احتوى جدول Departements على إدارة (قسم) لا يوجد عاملين معنى ذلك أنّ تلك الإدارة لن يتم اختيارها في استعلام ربط بين الجدولين.

ففي الشكل التوضيحي نجد أنه لا يوجد موظفين في الإدارة Control And Credit .

Departement_ID	Departement_NAME
10	Adminstration
20	Marketing
30	Purchasing
40	Human Resources
50	Shipping

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

60	IT
70	Public Relations
80	Sales
90	Executive
100	Finance
110	Accounting
120	Treasury
130	Corporate Tax
140	Control and credit
150	Shareholder Services
160	Benefits
170	Manufacturing
180	Constructing
190	construction
200	operations

FIRST_NAME	DEPARTMENT_ID
Steven	90
Neena	90
Lex	90
Alexander	60
Bruce	60
David	60
Valli	60
Diana	60
Nancy	100
Daniel	100
John	100
Ismael	100
Jose Manuel	100
Luis	100
Den	30

جاء أحدنا بكتابة الصيغة الآتية :

```
Select E.First_Name, D.Departement_id, D.Departement_Name
From   Employees E , Departements D
where  E.Departement_id = D.departement_id ;
```

فسنجد أنّ هناك سجلات لا تحقق شرط الربط :

E.Departement_id = D.Departement_id

وبالتالي لن يتم استرجاعها في نتيجة الاستعلام.

وبما أنه لا يوجد موظفين بالإدارة (بالقسم) Control and Credit

وبالتالي فإن القسم Control and Credit لن يظهر في نتيجة الاستعلام

وذلك طبقاً لـ Equijoin .

لكي نعالج ذلك، لابدّ من إنشاء رابطة خارجية وذلك بوضع إشارة الجمع (+) محاطة بقوسين وذلك بعد اسم حدد شرط الربط where للجدول الذي لا يضم سجلات تقابلها سجل أو سجلات في الجدول المرتبط.

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

أي أنه في المثال السابق حتى نظهر اسم الـ Control و Control و
نجمع الاستعلام السابق، علينا من إعطاء جدولنا فارسة مستعار
لإشارة الجمع (+) مثلاً نجمع بعد اسم كل جدول ثم نربط في كل منهما
وذلك الجدول الذي لا يضم سجلات مقابلة سجل أو سجلات في الجدول
المرتبط وهو في **مثالنا** جدول Employees (Employeed.no) الذي لا يضم موظفين
تقابلهم إدارة (يعملون في إدارة Control and Credit).

الرابطة الذاتية:

يمكننا عمل الرابطة الذاتية عند ربط جدول ما مع نفسه (نوع العلاقة تعاودية)
عندما نريد ربط جدول مع نفسه، نكتب اسم الجدول نفسه عدة مرات في الجملة
where وبالتالي يتوجب علينا استخدام أسماء اعتبارية.

مثال: إن علاقة الإدارة بين الموظف والمدير هي علاقة تعاودية لأن المدير هو
موظف بحد ذاته
وبالتالي فإن الجدول Employees يحتوي على عمود (Employee_id) الذي يحدد
رقم الموظف والذي يعتبر المفتاح الرئيسي في جدول Employees
ويحتوي أيضاً على عمود Manager_id الذي يحدد رقم المدير.
لكن كل مدير هو موظف وليس كل موظف هو مدير.
لذلك توجد علاقة self join بين عمود Manager_id
و عمود Employee_id.
لذلك نكتب العبارة الآتية:

```
Select Worker.First_Name, Manager.First_Name
From Employees Worker, Employees Manager
where Worker.Manager_id = Manager.Employee_id
```


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

في جملة From عملنا Table Alias (اسم اعتباري للجدول)
EMPLOYEES مثل:

فأوّل اسم اعتباري للجدول EMPLOYEES هو Worker ('عامل')
وثاني اسم اعتباري للجدول EMPLOYEES هو Manager ('مدير')
وقد استخدمنا الاسمين الاعتباريين في عبارة SQL لنفسرا وذلك ممكن

في عبارة select طلبنا اظهار أسماء العاملين وأسماء مدرائهم.
وفي عبارة where كتبنا شرط ربط الجدولين Manager و Worker
والذي هو أنه حتى يكون مدير ما مديراً لموظف معين فيجب أن يتحقق أنّ:
رقم مدير ذلك الموظف هو ذاته الرمز الوظيفي للمدير.

في الحقيقة إننا نربط جدول واحد مع نفسه فجدول Worker وجدول Manager
هما اسمان اعتباريان لجدول واحد فقط وهو EMPLOYEES

فعندما نكتب Worker.manager_id كأننا نكتب employees.manager_id
وعندما نكتب Manager.employee_id كأننا نكتب employees.employee_id وهكذا..
أي أنّ العبارة السابقة (عبارة SQL السابقة) وكأننا نكتب:

```
Select E.First_Name, E.First_Name
from   EMPLOYEES  E
where  E.MANAGER_id = E.employee_id ;
```

ولكن العبارة السابقة خاطئة لأننا في شرط where ربطنا الجدول مع
نفسه مباشرة، من دون أسماء استعارية (اعتبارية)
وبالتالي فلذلك قلنا في البداية أنه عندما نريد ربط جدول مع نفسه يوجب علينا
استخدام أسماء اعتبارية

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

الرابطة الديكارتية:

للحصول على الرابطة الديكارتية، نقوم بحذف شرط الربط بين الجدولين في جملة where، عندها سيقوم SQL بربط كل سجل في الجدول الأول مع كل سجل في الجدول الثاني.

فعلى سبيل المثال، إذا كان جدول Departements يضم 10 سجلات، وجدول EMPLOYEES يحوي 30 سجلاً، فإذا تم حذف شرط الربط من جملة where، ففي نتيجة الاستعلام سيكون لدينا (30*10=300) سجل.

استخدام الدوال: Functions

مثلاً SQL دوالّ Functions فعالة وقوية عند استخدام عبارة select. تنقسم هذه الدوال إلى نوعين:

- الدوال التجميعية لأكثر من صف (Multiple Row Functions): تقوم بالتعامل مع أكثر من صف لإعطاء قيمة واحدة فقط.
- دوال الصف الواحد (Single Row Functions): تقوم بالتعامل مع بيانات الصف الواحد لإعطاء قيمة وحيدة.

استخدام الدوال لأكثر من سطر:

تزودنا SQL بعدد دوال تجميع Group Functions يمكن أن تنفذ على مجموعة من السجلات، وهي كما في الجدول الآتي:

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

الاستخدام	الدالة
حساب عدد السجلات في جدول ما	Count
تحديد القيمة العظمى لعمود ما	Max
تحديد القيمة الدنيا لعمود ما	Min
حساب المتوسط الحسابي لعمود ما	Avg
حساب الانحراف المعياري لعمود ما	Stddev
حساب المجموع الكلي لعمود ما	Sum
حساب التغاير لعمود ما	Variance

تقوم الدوال التجميعية Group Functions بمعالجة قيم العمود المختار من الجدول (أي تتعامل مع أكثر من سطر لذلك تسمى Multiple Row Functions) وتقوم النتيجة في شكل قيمة وحيدة تخص العمود المختار.

يجب علينا اتباع القواعد الآتية عند استخدام Group Functions :

١- وضع اسم العمود بين قوسين ( ) بعد الدالة مباشرة.

Select Max (Salary) From EMPLOYEES;

اسم العمود هو SALARY ونريد أكبر قيمة فيه

٢- يمكن لجميع الأعمدة في عملة select أن تكون كلها دوال تجميعية.

Select Max (SALARY), Min(salary), Sum(salary);

٣- استخدام دالة تجميعية أخرى (أي تركيب الدوال التجميعية) غير مسموح به في SQL.

Select Max(Avg(SALARY)) From Employees; هذه العبارة خاطئة.

٤- الدوال التجميعية تجاهل القيم الفارغة Null values.

٥- يمكن استخدام الدوال التجميعية ضمن التعابير الحسابية.

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

استخدام Count :

تستخدم الدالة Count لحساب عدد السجلات التي تضم قيماً فارغة (Not null) في العمود المحدد في المجموعة .

فمثلاً لحساب عدد السجلات في جدول EMPLOYEES الذي يملك قيماً غير فارغة في العمود First_Name نكتب العبارة الآتية:

```
Select count(First_Name) From employees;
```

ملاحظات :
- يتم وضع العمود المراد حساب عدد سجلاته بين قوسين بعد الدالة Count .
- لا يتم عد السجلات التي تحوي قيماً فارغة Null .

استخدام Count(*) :

عندما نستبدل اسم العمود بالنجمة (*) تقوم الدالة Count بحساب عدد السجلات بما فيها السجلات التي تحتوي على قيم فارغة .

مثلاً: لعد سجلات جدول EMPLOYEES فإننا نكتب العبارة الآتية:

```
Select count(*)
from EMPLOYEES ;
```

مثال آخر: لحساب عدد السجلات الناتجة من الجداء الديكارتي لجدولي Departements , EMPLOYEES نكتب العبارة الآتية:

```
Select count(*)
from EMPLOYEES,DEPARTEMENTS;
```


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

مثال آخر: لحساب عدد السجلات للموظفين الذين يعملون في القسم رقم 10 نكتب:

```
Select Count(*)
From EMPLOYEES
where Departement_id=10;
```

استخدام Distinct مع Count:

يمكننا استخدام كلمة Distinct المجموعة مع Count حيث نقوم باستبعاد السجلات المكررة.

على سبيل المثال: لحساب عدد الموظفين مختلفي الأسماء من جدول Employees نكتب العبارة الآتية:

```
Select Count(Distinct(First_Name)) From employees;
```

إن استخدام Distinct مع Count لن يأخذ القيم الفارغة (أي إذا كان لدينا 2 موظفين ليس لهم اسم أول فسيوفر count بعدهم لأننا أخذنا Distinct لهذا السبب)

استخدام الدالة Max:

تستخدم الدالة Max لحساب أكبر قيمة في مجموعة ما، حيث يمكن استخدام الدالة Max مع القيم: • العددية • أو النصية • أو التاريخ والوقت.

أي أنه يمكن استخدام الدالة Max مع أي نوع من البيانات.

- عند استخدام Max مع البيانات الحرفية فإنها تعيد أعلى قيمة في جدول Ascii (أي أن B سيكون أكبر من A و C أكبر من B وهكذا)
- وأيضاً سيكون B أكبر من b و C أكبر من c وهكذا).
- عند استخدام Max مع الأعمدة العددية فإنها تعيد أعلى قيمة عددية.
- عند استخدام Max مع التاريخ والوقت تعيد الأحدث في العمود.
- تجعل الدالة Max القيمة Null.

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

مثلاً: من أجل معرفة: أعلى راتب للموظفين ومعرفة أحدث وقت تم تعيين فيه موظف ما، ومعرفة الموظف ذو الاسم الأعرفي حسب Acsii

نكتب العبارة الآتية:

```
Select Max(Salary), Max(Hire_Date), Max(First_Name)
From EMPLOYEES;
```

استخدام الدالة Min:

تعيد الدالة Min أصغر قيمة لاتساوي Null.

كما أنها تستخدم مع أي نوع من البيانات.

مثلاً: لتحديد أقل راتب في القسم 10 نكتب العبارة:

```
Select Min(SALARY)
from EMPLOYEES
where Departement_id=10;
```

استخدام الدالة Avg:

تستخدم الدالة Avg لحساب المتوسط الحسابي لعمود ما، حيث تعيد Avg القيمة الوسطية لجميع القيم التي لاتساوي Null في العمود.

هناك عدة اعتبارات يجب مراعاتها عند استخدام Avg:

- يستخدم Avg مع القيم العددية فقط.
- تجاهل Avg القيمة Null عند حسابه للقيمة الوسطى.
- تجاهل Avg القيم المتكررة في العمود عند حسابه للقيمة الوسطى وذلك إذا وضعنا كتابة Distinct قبل اسم العمود.

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

مثال (١): يمكننا حساب المتوسط الحسابي لعمود Salary في جدول EMPLOYEES وذلك بكتابة العبارة الآتية:

```
Select Avg(SALARY) from EMPLOYEES;
```

مثال (٢): يمكننا حساب أعلى قيمة وأدنى قيمة والمتوسط الحسابي لعمود SALARY بكتابة العبارة الآتية:

```
Select Max(SALARY), Min(SALARY), AVG(SALARY)
From EMPLOYEES;
```

مثال (٣): يمكننا حساب المتوسط الحسابي لعمود COMMISSION_PCT في جدول employees وذلك بكتابة العبارة الآتية:

```
Select Avg(commission_pct)
from EMPLOYEES;
```

نلاحظ في هذا المثال أنّ Avg تتجاهل القيم Null.

إذا كانت كل قيم العمود Commission_Pct هي null فسوف يكون ناتج Avg هو null.

استخدام الدالة Sum:

تستخدم الدالة Sum لحساب المجموع الكلي لقيم عمود ما، هناك عدة اعتبارات يجب مراعاتها عند استخدام Sum:

- تستخدم Sum مع القيم العددية فقط.
- تهمل Sum القيمة Null عند حساب المجموع الكلي.
- تهمل Sum القيم المتكررة في العمود عند حساب المجموع الكلي وذلك إذا مررنا كتابة Distinct قبل اسم العمود.

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

مثال: حساب المجموع الكلي لعمود SALARY في جدول EMPLOYEES

نكتب الجملة التالية:

```
Select Sum(SALARY) From EMPLOYEES;
```

استخدام الدالة NVL مع الدوال التجميعية Group Functions

تستخدم الدالة NVL مع الأعمدة التي تحتوي القيم الفارغة Null Value حيث تقوم معالجة القيم الفارغة باستبدالها بالقيم الرقمية (أو أي قيمة أخرى) وبالتالي تمكن الدالة NVL من استخدام الدوال التجميعية مع الأعمدة التي تضم قيما فارغة، لكي تشمل جميع السجلات

مثلا: إذا كان لدينا عمود ما X وأردنا أن نستبدل كل قيمة Null في هذا العمود بالقيمة صفر فإننا نكتب: NVL(X, 0).

استخدام الدوال لسطر واحد:

تعالج Single Row Functions قيم العمود المختار من الجدول، وتقدم النتيجة على شكل قيمة وحيدة لكل سجل

في حين تستخدم Group Functions في معالجة قيم العمود المختار من الجدول وتقدم النتيجة على شكل قيمة وحيدة تخص العمود المختار.

أنواع الدوال لسطر واحد:

١- نصية ٢- رقمية ٣- تاريخ ٤- تحويلية ٥- عامة

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

الدوال المخصصة Character functions :

تملك SQL عدداً من الدوال التي تتعامل مع أنماط البيانات المخصصة (char, Var char)

وسنعرض لأهم هذه الدوال المخصصة مع ملاحظة طريقة الاستخدام.

الدالة المخصصة	
Initcap	• تغيير حالة الحرف الأول من كل سلسلة مخصصة إلى حرف كبير • تحويل حالة الأحرف التالية للحرف الأول إلى أحرف صغيرة
Length	• إعادة عدد الأحرف المكونة لسلسلة مخصصة
Lower	• تحويل كل حرف ضمن سلسلة مخصصة إلى أحرف صغيرة
Upper	• تحويل كل حرف ضمن سلسلة مخصصة إلى أحرف كبيرة
Concat	• دمج سلسلة مخصصة بسلسلة مخصصة أخرى
Replace	• استبدال سلسلة مخصصة بسلسلة مخصصة أخرى
Soundex	• إيجاد الكلمات المتشابهة من حيث اللفظ الصوتي
Substr	• نسخ الجزء المحدد من الكلمة استناداً من موقع محدد
Instr	• إعادة قيمة عددية تمثل موضع محرف أخر ضمن سلسلة مخصصة

سنقوم باستعراض أمثلة عن الدوال المخصصة كما هو موضح في الجدول الآتي:

Function الدالة	Result النتيجة
Lower ('Jicc Center')	jicc center
Upper ('Jicc Center')	JICC CENTER
Initcap ('jicc center')	Jicc Center
Concat ('Jicc'),('Center')	JiccCenter
Substr ('String',1,3)	Str (نسخنا ابتداءً من الحرف الأول إلى الثالث)
Length ('String')	6
Instr ('String','T')	2 (موضع أول ظهور لـ T في String)

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

مثال (1): إذا أردنا معرفة اسم كل موظف و دمج اسمه الأول مع رقم الوظيفة و عدد أحرف اسمه الأول و موضع حرف A من اسمه الأول.

نكتب العبارة الآتية:

```
Select First_Name, Concat(First_Name,JOB_ID),
       Length(First_Name), InStr(First_Name,'A')
from EMPLOYEES;
```

مثال (2): إذا أردنا معرفة راتب وعمل الموظف الذي اسمه Jicc Center فنكتب العبارة الآتية:

```
Select SALARY, JOB_id
from EMPLOYEES
where First_Name = 'Jicc Center';
```

ولكن إذا كان الاسم مكتوبا في قاعدة البيانات بالشكل (1) JICC CENTER

```
jicc center
```

أو بالشكل (2):

```
JiCc CeNteR
```

أو

فلن تظهر أي نتيجة، وبالتالي إذا أردنا أن تظهر النتيجة مهما كانت الطريقة التي مكتوب بها الاسم في قاعدة البيانات فعلينا استخدام الدوال الداخلية فنكتب مثلاً:

```
Select SALARY, JOB_id
from EMPLOYEES
where Upper(First_Name)='JICC CENTER';
```

عندئذ سنضمن ظهور نتيجة صحيحة مهما كانت حالة أحرف الاسم في قاعدة البيانات.

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

ويمكننا أيضاً أن نكتب:

```
Select SALARY, JOB_id
From EMPLOYEES
where Lower(First_Name)='jicc center';
```

أو أن نكتب:

```
Select SALARY, JOB_id
From EMPLOYEES
where Initcap(First_Name)='Jicc Center';
```

وكل هذه الحالات ستظهر النتيجة ذاتها.

انتهت المحاضرة.

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