

ربط الجداول

Joining Tables

فكرة المحاضرة:

سنناقش في هذا الدرس مبادئ وتقنيات إنشاء استعلامات متعددة الجداول، بما في ذلك ربط استعلامين فرعيين ضمن عبارة **From**. إذ يمكن لاستعلام **SQL** التعامل مع عدد كبير من الجداول، لكن يجب على كل جدول أن يتشارك بحقل مع جدول واحد على الأقل ليتم تشكيل ما يسمى سلسلة العلاقات أو **Relationship chain**. يوجد العديد من أشكال الربط بين الجداول، وتختلف الصيغة باختلاف نظام إدارة قواعد البيانات.

مواضيع المحاضرة:

كيف يقوم **SQL** بربط الجداول؟
الربط البسيط أو الجداء الديكارتي **Cartesian Product**
الربط بالتساوي **Equi Join**
الربط باللامساواة **Non-Equi Join**
الربط الداخلي **Inner Join**
الربط الخارجي **Outer Join**
الربط الذاتي **Self Join**
ربط استعلامين فرعيين **Join two sub-queries**

مقدمة:

وجدنا فيما سبق أن الاستعلام عن أكثر من جدول باستخدام الاستعلامات الفرعية قد وفر قدرات جيدة على معالجة البيانات. ولكن هذه القدرات قد لا تمكننا دائماً من الحصول على كل النتائج التي نحتاجها. كما أن الصيغة قد تصبح صعبة الفهم بعض الشيء وقد تؤدي إلى انخفاض مستوى الأداء أحياناً.

توفر **SQL** إمكانية الاستعلام عن جداول متعددة في وقت واحد باستخدام صيغة أبسط ندعوها الربط.

لا تستطيع الصيغة الجديدة استبدال كل التقنيات التي تؤمنها الاستعلامات الفرعية، ولكنها تمثل الحل الأمثل في بعض الحالات، وخاصة في الاستعلامات التي تربط بين سجلات من جداول مختلفة.

نبدأ فيما يلي بعرض حالات الربط البسيطة لنصل بعد ذلك إلى الحالات الأكثر تعقيداً:

- الربط البسيط أو الجداء الديكارتي **Cartesian product**
- الربط بالتساوي **Equi join**
- الربط بعدم المساواة **Non-Equi join**
- الربط الخارجي **Outer join**

كيف يقوم **SQL** بربط الجداول؟

ليكن لدينا الجدولين التاليين:

Department	
DeptID	DeptName
Actg	Accounting
Admn	Administration
Fin	Finance
Mktg	Marketing

Employeee				
EmpID	FirstName	LastName	Gender	DeptID
1	عمر	السهلي	Male	Actg
2	طارق	عواد	Male	Mktg
3	سامية	سمان	Female	Mktg
4	حنان	آغا	Female	NULL

Children				
ChildID	ChildName	Gender	Birthdate	EmpID
1	سحر	Female	11/02/1990	1
2	طوني	Male	19/02/1990	2
3	رهام	Female	30/08/1992	1
4	لميس	Female	19/09/1980	3
5	وسيم	Male	04/03/1999	4
6	حسن	Male	02/07/2000	1
7	محمد	Male	29/05/1991	3
8	عبير	Female	09/01/2001	2

تذكير ببعض التعريفات:

- **المفتاح الرئيسي Primary key:**
حقل أو أكثر تعرّف فيه قيمة أو قيم فريدة لكل سجل في الجدول. في الجداول السابقة يمثل الحقل EmpID مفتاحاً رئيسياً للجدول Employeee، ويمثل الحقل DeptID مفتاحاً رئيسياً للجدول Department. ويمثل الحقل ChildID مفتاحاً رئيسياً للجدول Children.

- **المفتاح الخارجي Foreign key:**
حقل أو أكثر من حقول الجدول يشير إلى حقل أو حقول المفتاح الرئيسي في جدول آخر. في الجداول السابقة يمثل الحقل DeptID الموجود في الجدول Employeee مفتاحاً خارجياً للجدول Department.

الربط البسيط أو الجداء الديكارتي Cartesian Product

تتم عملية الربط بمقابلة كل سجل من الجدول الأول مع جميع سجلات الجدول الثاني. وهذا ما يدعى بالجداء الديكارتي أو *Cartesian product*. وفي هذه الحالة يكون عدد السجلات الناتجة عن الاستعلام هو عدد سجلات الجدول الأول × عدد سجلات الجدول الثاني.

مثال: نتيجة الربط البسيط بين الجدولين: Employeee و Department هي:

```
select * from Employee , Department
```

Cartesian Product

EmpID	FirstName	LastName	Gender	DeptID	DeptID	DeptName
1	عمر	السهلي	male	actg	actg	accounting
2	طارق	العواد	male	mktg	actg	accounting
3	سامية	سمان	female	mktg	actg	accounting
4	حنان	آغا	female	NULL	actg	accounting
1	عمر	السهلي	male	actg	admn	administration
2	طارق	العواد	male	mktg	admn	administration
3	سامية	سمان	female	mktg	admn	administration
4	حنان	آغا	female	NULL	admn	administration
1	عمر	السهلي	male	actg	fin	finance
2	طارق	العواد	male	mktg	fin	finance
3	سامية	سمان	female	mktg	fin	finance
4	حنان	آغا	female	NULL	fin	finance
1	عمر	السهلي	male	actg	Mktg	marketing
2	طارق	العواد	male	mktg	Mktg	marketing
3	سامية	سمان	female	mktg	Mktg	marketing
4	حنان	آغا	female	NULL	Mktg	marketing

يمكن التعبير عن صيغة الربط بالصيغة التالية:

```
SELECT Table1.Column1, Table2.Column2
FROM Table1, Table2;
```

يسمى استعلام الربط البسيط أيضاً **بالربط المتصالب** أو **Cross join**. ويمكن التعبير عن نفس صيغة الربط السابقة، بالصيغة:

```
SELECT Table1.Column1, Table2.Column2
FROM Table1 CROSS JOIN Table2;
```

الربط بالتساوي Equi Join

يُعرّف الربط بالتساوي على أنه الربط البسيط بين سجلات جدول أول، وسجلات جدول ثان اعتماداً على مساواة بين قيمة حقل في سجل من الجدول الأول (عادة المفتاح الخارجي) وقيمة حقل في سجل من الجدول الثاني (عادة المفتاح الرئيسي).

يُعبّر عن الربط بالتساوي بالصيغة:

```
SELECT Table1.Column1, Table1.Column2, Table2.Column3
FROM Table1, Table2
WHERE Table1.Column1 = Table2.Column2;
```

عموماً، لا تستخدم عملية الربط بالضرورة نفس الحقول التي يجب أن يعيدها الاستعلام.

مثال:

لنفرض أننا نبحث عن أسماء الموظفين الذين يعملون في قسم المحاسبة. عندها سيكون الاستعلام على الشكل التالي:

```
SELECT DeptName, FirstName, LastName
FROM Employeee, Department
WHERE Employeee.DeptID = Department.DeptID
AND DeptName = 'Accounting'
```

وستكون النتيجة هي:

DeptName	FirstName	LastName
Accounting	عمر	السهلي

Equi Join	
<pre>SELECT Table1.Column1, Table1.Column2, Table2.Column3 FROM Table1, Table2 WHERE Table1.Column1 = Table2.Column2; SELECT Table1.Column1, Table1.Column2, Table2.Column3 FROM Table1 Join Table2 ON Table1.Column1 = Table2.Column2;</pre>	Syntax الصيغة
<p>1. أسماء الموظفين الذين لديهم ولد اسمه وسيم.</p> <pre>SELECT FirstName, LastName FROM Employeee, Children WHERE Employeee.EmpID = Children.EmpID AND ChildName = 'وسيم'</pre>	أمثلة
<p>2. أسماء الأبناء الذكور للموظفين العاملين في قسم التسويق.</p> <pre>SELECT ChildName FROM Department, Employeee, Children WHERE Department.DeptID = Employeee.DeptID AND Employeee.EmpID = Children.EmpID AND Children.Gender = 'Male' AND DeptName = 'Marketing'</pre>	

3. أسماء الموظفين وأعداد أولاد الموظفين (في جميع الأقسام) (جميع الموظفين).

```
SELECT FirstName, LastName,
       Count(ChildID) As ChildrenCount
FROM Employeee, Children
WHERE Employeee.EmpID = Children.EmpID
GROUP BY FirstName, LastName
```

انتبه:

عند إضافة العلاقة Primary key – Foreign key، يتم استعمال اسم الجدول ملحقاً بنقطة '، لمنع الالتباس عند تشابه أسماء الحقول الموجودة في عدة جداول. كما يجب استخدام اسم الجدول ملحقاً بنقطة ضمن التعبير Select أو Where أيضاً عند إمكانية حدوث التباس بالحقول. انظر المثال 2 في الجدول السابق.

الربط باللامساواة Non-Equi Join

يعتمد الربط باللامساواة على استخدام المساواة في شرط التعبير WHERE ولكن هذا لا يعني أننا لا نستطيع استخدام عمليات المقارنة الأخرى (أكبر، أصغر، وغيرها) كما في الصيغة التالية:

```
SELECT Table1.Column1, Table2.Column2
FROM Table1, Table2
WHERE Table1.Column1 < Table2.Column2;
```

مثال:

تحديد الاستعلام الذي يعطي أسماء الأقسام وأسماء الموظفين الذين لا يعملون فيها. عندها سيكون الاستعلام على الشكل التالي:

```
SELECT DeptName, FirstName, LastName
FROM Employeee, Department
WHERE Employeee.DeptID <> Department.DeptID
```

وستكون النتيجة هي:

accounting	طارق	العواد
accounting	سامية	سمان
administration	عمر	السهلي
administration	طارق	العواد
administration	سامية	سمان
finance	عمر	السهلي
finance	طارق	العواد
finance	سامية	سمان
marketing	عمر	السهلي

الربط الداخلي Inner Join

يعطي الربط الداخلي نفس النتيجة التي يعطيها الربط بالتساوي، الفرق فقط بالصيغة. إذ لا تزودنا جميع أنواع أنظمة إدارة قواعد المعطيات بالربط الداخلي. فنسخ Oracle ما قبل 9 لا تدعم الربط الداخلي بالنسبة للصيغة، يوجد اختلافان:

- يفصل بين أسماء الجداول الكلمات Inner Join بدلاً من الفواصل '،'.
- يتغير موضع تحديد العلاقة بين الجداول من Where إلى On، تاركين بذلك التعبير Where للشروط التقليدية.

الأمثلة الواردة في الجدول التالي هي نفسها الواردة في فقرة الربط بالتساوي، وبالتالي يمكنك المقارنة.

Inner Join	
<pre>SELECT * FROM Table1 INNER JOIN Table2 ON Table1.Field = Table2.Field</pre>	Syntax الصيغة
<p>1. أسماء الموظفين الذين لديهم ولد اسمه وسيم.</p> <pre>SELECT FirstName, LastName FROM Employeee Inner Join Children ON Employeee.EmpID = Children.EmpID WHERE ChildName = 'وسيم'</pre>	مثال
<p>2.أ. أسماء الأبناء الذكور للموظفين العاملين في قسم التسويق.</p> <pre>SELECT ChildName FROM Department Inner Join Employeee ON Department.DeptID = Employeee.DeptID Inner Join Children ON Employeee.EmpID = Children.EmpID WHERE Children.Gender = 'Male' AND DeptName = 'Marketing'</pre>	
<p>2.ب. أسماء الأبناء الذكور للموظفين العاملين في قسم التسويق.</p> <pre>SELECT ChildName FROM (Department Inner Join Employeee ON Department.DeptID = Employeee.DeptID) Inner Join Children ON Employeee.EmpID = Children.EmpID WHERE Children.Gender = 'Male' AND DeptName = 'Marketing'</pre>	

3. أسماء وأعداد أولاد الموظفين في جميع الأقسام.	مثال
<pre> SELECT FirstName, LastName, Count(ChildID) As ChidrenCount FROM Employeee Inner Join Children ON Employeee.EmpID = Children.EmpID GROUP BY FirstName, LastName </pre>	

ملاحظة:

رأينا فيما سبق كيفية تغيير اسم حقل أو عمود ضمن التعبير Select في أي استعلام، وذلك عبر ما يسمى Alias. يمكننا أيضاً تغيير أسماء الجداول ضمن SQL، بهدف تقصير طول الاستعلام أو تسهيل قراءته. لاستعمال alias ضمن التعبير From، أتبع الاسم الحقيقي للجدول بفراغ ثم الاسم الجديد الذي نرغب بالتعامل معه. ويمكنك إضافة As بين الاسم الحقيقي للجدول والاسم الجديد له. وعند تغيير تسمية جدول، لا يمكن استخدام الاسم الحقيقي في بقية الاستعلام. يبين الجدول التالي عدة أمثلة على استخدام إعادة تسمية الجداول.

Table Aliases	
<pre> SELECT * FROM Table1 Alias1 Inner Join Table2 Alias2 On Alias1.Field = Alias2.Field SELECT * FROM Table1 Alias1, Table2 Alias2 Where Alias1.Field = Alias2.Field </pre>	Syntax الصيغة
<p>..1</p> <pre> SELECT FirstName, LastName FROM Employeee E, Children C WHERE E.EmpID = C.EmpID AND ChildName = 'وسيم' </pre>	أمثلة
<p>2. أسماء الأبناء الذكور للموظفين العاملين في قسم التسويق.</p> <pre> SELECT ChildName FROM Department D Inner Join Employeee E ON D.DeptID = E.DeptID Inner Join Children C ON E.EmpID = C.EmpID WHERE Children.Gender = 'Male' AND DeptName = 'Marketing' </pre>	

الربط الخارجي Outer Join

في حالة INNER JOIN، كانت السجلات التي أرجعها الاستعلام، هي السجلات التي تحقق شرط الربط الذي يظهر بعد تعبير ON، حيث تم إسقاط السجلات غير المتطابقة من جدول النتائج. أما في حالة الربط الخارجي Outer Join فلا يتم إسقاط السجلات غير المتطابقة.

للربط الخارجي ثلاثة أنواع: LEFT, RIGHT, FULL.

1. Left Outer Join:

لأخذ جميع السجلات من الجدول الأول Table1 وفقط السجلات من الجدول الثاني Table2 التي تتطابق فيها قيمة الحقل Column1 من الجدول Table1 مع قيمة الحقل Column2 من الجدول الثاني Table2، نكتب الصيغة:

```
SELECT *  
FROM Table1  
LEFT OUTER JOIN Table2  
ON Table1.Column1 = Table2.Column2;
```

2. Right Outer Join:

لأخذ جميع السجلات من الجدول الثاني Table2 وفقط السجلات من الجدول الأول Table1 التي تتطابق فيها قيمة الحقل Column1 من الجدول Table1 مع قيمة الحقل Column2 من الجدول الثاني Table2، نكتب الصيغة:

```
SELECT *  
FROM Table1  
RIGHT OUTER JOIN Table2  
ON Table1.Column1 = Table2.Column2;
```

3. Full Outer Join:

لأخذ جميع السجلات من الجدول الثاني Table2 وجميع السجلات من الجدول الأول Table1 بحيث تتوضع السجلات التي تتطابق فيها قيمة الحقل Column1 من الجدول Table1 مع قيمة الحقل Column2 من الجدول الثاني Table2 في نفس السجل من جدول القيم المعادة، نكتب الصيغة:

```
SELECT *  
FROM Table1  
FULL OUTER JOIN Table2  
ON Table1.Column1 = Table2.Column2;
```

ملاحظة:

ينتج عن عمليات الربط الخارجي، في الحالة العامة، سجلات تحتوي في حقول معينة القيمة NULL بسبب اختلاف عدد السجلات التي نريد ربطها، وهذا ما سنوضحه بالتفصيل لاحقاً مع مثال مناسب لكل نوع من أنواع الربط الخارجي .

لا تعتمد جميع أنظمة إدارة قواعد البيانات DBMS نفس الصيغة في التعبير عن أشكال Outer Join المختلفة

الصيغة المستخدمة في sql server:

هي الصيغة المطابقة تقريباً لصيغة Inner Join (وهي ما استخدمناه فيما سبق من هذه الفقرة). أي يتم وضع الكلمات Left Outer Join (أو اختصاراً Left Join) أو الكلمات Right Outer Join (أو اختصاراً Right Join) أو الكلمات Full Outer Join (أو اختصاراً Full Join) بين أسماء الجداول، يلي اسم الجدول الثاني تعليمة ON لتحديد الحقول الرابطة بين الجداول.

Outer Join (first syntax form)	
<pre>SELECT Table1.Column1, Table1.Column2, Table2.Column3 FROM Table1 LEFT RIGHT FULL JOIN Table2 ON Table1.Field = Table2.Field</pre>	Syntax الصيغة
<p>1. اعرض أسماء جميع الأقسام مع ما يقابلها من موظفين يعملون في هذه الأقسام.</p> <pre>SELECT DeptName, FirstName, LastName FROM Department LEFT JOIN Employeee ON Department.DeptID = Employeee.DeptID Where employee.name = null</pre>	مثال

ستكون النتيجة هي:

accounting	طارق	العواد
administration	عمر	السهلي
administration	null	null
finance	null	null
marketing	طارق	عواد
marketing	عمر	السهلي

تمرين:

أعد كتابة الاستعلام التالي مستخدماً Outer Join بدلاً من Not IN.

```
Select Artistname
From Artists
Where ArtistID NOT IN (Select ArtistID From Movies);
```

الحل:

```

Select Artistname
From Artists A Left Join Movies M
ON A.ArtistID = M.ArtistID
WHERE M.ArtistID Is NULL;

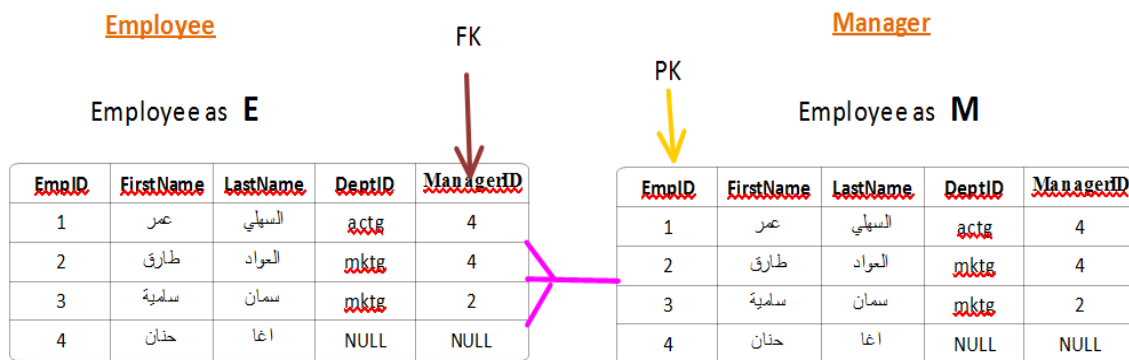
```

الربط الذاتي Self Join

الربط الذاتي هو ربط جدول مع نفسه. ويمكن ذلك عبر Inner Join أو Outer Join. لنأخذ بنية معدلة عن الجدول Employee الذي عرّفناه في بداية الجلسة، وذلك بإضافة ManagerID إليه كما هو موضح في الجدول التالي.

Employee				
EmpID	FirstName	LastName	DeptID	ManagerID
1	عمر	السهلي	actg	4
2	طارق	العواد	mktg	4
3	سامية	سمان	mktg	2
4	حنان	أغا	NULL	NULL

يُعبّر الحقل الجديد ManagerID عن الرقم EmpID لمدير الموظف. وإذا كانت قيمته NULL فهذا يعني أن الموظف الحالي هو المدير العام أو ليس له مدير.



تمرين:

اكتب الاستعلام الذي يعطي أسماء جميع الموظفين مع اسم المدير (إن كان موجوداً).

ملاحظة: نستخدم الاسم المستعار M ونستخدم معه حقل المفتاح الرئيسي، لأن العلاقة من طرف جدول المدير هي واحد وبالتالي هو الجدول الرئيسي، بينما هي من طرف الموظفين متعدد .. وبالتالي معه نستخدم المفتاح الثانوي ManagerID

الحل:

```

Select E.FirstName, E.LastName,
      M.FirstName, M.LastName
From Employee E LEFT JOIN Employee M
      ON E.ManagerID = M.EmpID;

```

لا حظ الاستخدام الإجباري لإعادة التسمية (Aliasing) ضمن عبارة From في التمرين السابق.

Self Join	
<pre> SELECT Alias1.Field1, Alias2.Field2 FROM Table Alias1 Inner Left Right Full Join Table Alias2 On Alias1.Field1 = Alias2.Field2 SELECT Alias1.Field1, Alias2.Field2 FROM Table Alias1, Table Alias2 Where Alias1.Field1 = Alias2.Field2 </pre>	الصيغة Syntax
<p>1.أ. اكتب الاستعلام الذي يعطي أسماء جميع الموظفين مع اسم المدير (إن كان موجوداً).</p> <pre> Select E.FirstName, E.LastName, M.FirstName, M.LastName From Employee E LEFT JOIN Employee M ON E.ManagerID = M.EmpID; </pre>	
<p>1.ب. اكتب الاستعلام الذي يعطي أسماء الموظفين مع اسم المدير (يعطي فقط اسم الموظف مع اسم المدير المرتبط به)</p> <pre> Select E.FirstName, E.LastName, M.FirstName, M.LastName From Employee E, Employee M WHERE E.ManagerID = M.EmpID; </pre>	

ربط استعلامين فرعيين Join two sub-queries

في الكثير من الأحيان نضطر لاستحصال معلومات موجودة في جداول ضخمة جداً. وتؤدي عملية ربط هذه الجداول مع بعضها للحصول على كميات هائلة من السجلات المتقابلة، والتي يتم اختصارها وفق شروط معينة. أما إذا تم اختصار هذه الجداول وفق الشروط المعطاة قبل البدء بعملية الربط، سيؤدي ذلك إلى تسريع الاستعلام المطلوب، وبالتالي إلى تسريع النظام ككل.

وهنا يأتي دور الاستعلامات الفرعية الموجودة ضمن عبارة FROM في اختصار عدد السجلات التي ستربط مع سجلات أخرى.

لا يوجد مبرر في مثالنا البسيط لاستخدام الاستعلام الفرعي سوى التمرين والتدريب على المفهوم.

ملاحظات:

- تعمل الاستعلامات الفرعية بشكل جيد في SQL Server

Subqueries in FROM Clause	
SQL Server	
<p>SELECT Field Field, Field, Field *</p> <p>FROM Table1 Inner Left Right Join</p> <p>(SELECT Field, Field FROM Table Where condition) Alias</p> <p>On Table1.Field = Alias.Field</p> <p>SELECT Field Field, Field, Field *</p> <p>FROM Table1 ,</p> <p>(SELECT Field, Field</p> <p>FROM Table</p> <p>Where condition) Alias</p> <p>Where Table1.Field = Alias.Field</p>	الصيغة Syntax
<p>1. ترغب المؤسسة بتقديم معونات مالية لموظفيها. لذلك ترغب بمعرفة أسماء العاملين وأعداد أولادهم الذين تزيد أعمارهم عن 5 سنوات، وذلك في كل قسم من أقسامها. اكتب الاستعلام الذي يؤمن للمؤسسة هذه المعلومات.</p> <p>(الاستعلام الفرعي: يعطي رقم القسم مع اسم الموظف وكنيته مع عدد أولاده الذين يزيد عددهم عن خمس سنوات وذلك لكل الأقسام)،</p> <p>نقوم بربط البنية الجدول الناتج من الاستعلام الفرعي مع جدول الأقسام من أجل الحصول على اسم كل قسم.. لأن الاستعلام الفرعي لا يحتوي إلا على رقم القسم....</p> <pre> Select d.DeptName, sub.FirstName, sub.LastName, sub.ChildrenCount FROM Department D INNER JOIN (SELECT DeptID, FirstName, LastName, COUNT(ChildID) AS ChildrenCount FROM Employeee INNER JOIN Children ON Employeee.EmpID = Children.EmpID WHERE DATEDIFF(yy, Children.Birthdate, GETDATE()) > 5 GROUP BY DeptID, FirstName, LastName) AS sub ON D.DeptID = sub.DeptID </pre>	

النتيجة هي:

accounting	عمر	السهلي	3
marketing	سامية	سمان	2
marketing	طارق	العواد	2