



## الفصل العاشر:

### لغة الاستعلامات المهيكلية

### (SQL) (3)

الصفحة	العنوان
4	<b>1. مقدمة</b>
4	<b>2. شروط التكامل المرجعي (Integrity constraints)</b>
4	<b>1.2</b> المفتاح الأساسي Primary key
4	<b>2.2</b> المفتاح المستورد Foreign key
5	<b>3.2</b> الوحدانية Unique
5	<b>4.2</b> اجباري Not Null
6	<b>3. القيم الافتراضية Default</b>
6	<b>4. شروط التحقق Check</b>
7	<b>5. التوابع في SQL</b>
7	<b>1.5</b> التوابع التجميعية
9	<b>2.5</b> التوابع الدرجية
11	<b>6. الأسماء البديلة Aliases</b>
13	<b>7. تعليمة Select المتداخلة (nested select statement)</b>
15	<b>8. تعليمات Select المترابطة (Correlated select statement)</b>
15	<b>9. تطبيق</b>
18	<b>10. المراجع</b>
19	<b>11. تدريبات</b>

## الكلمات المفتاحية:

جدول، سجل، عمود، تابع، قيمة، بيانات، حقل، جدول، رقمية، سلاسل محارف، تاريخ، صيغة، قاعدة بيانات.

## ملخص:

هذه الوحدة هي توسع في لغة الإستعلام المُهيكلية وتعرض مفاهيم متقدمة في SQL وتدريبات تطور مهارة الطالب في استخدام SQL.

## أهداف تعليمية:

يهدف هذا الفصل التعريف بالمفاهيم التالية:

- شروط التكامل المرجعي في SQL:
- 1. المفاتيح الأساسية Primary Keys
- 2. المفاتيح المستوردة Foreign Keys
- 3. الوحداية Unique
- 4. الالزامية Not null
- القيم الافتراضية Default
- شروط التحقق Check
- التوابع في SQL
- 5. التوابع التجميعية
- 6. التوابع الدرجية
- الأسماء البديلة Aliases
- تعليمة Select المتداخلة (Nested Select statement)
- تعليمات Select المترابطة Correlated Select statement
- تطبيق

## المخطط:

- مقدمة
- شروط التكامل المرجعي (Integrity constraints)
- القيم الافتراضية Default
- شروط التحقق Check
- التتابع في SQL
- الأسماء البديلة Aliases
- تعليمة Select المتداخلة (nested select statement)
- تعليمات Select المترابطة (Correlated select statement)
- تطبيق

## 1. مقدمة

في هذا الفصل سنعرض حالات متقدمة لاستخدام تعليمة Select موضوع الفصل السابق، وتتمت أساسية في لغة الاستعلام المهيكل.

## 2. شروط التكامل المرجعي (Integrity constraints):

شروط التكامل المرجعي هي مجموعة من القواعد، تطبق على نمط وقيم البيانات التي يمكن إدخالها إلى أعمدة الجداول، والغاية منها ضمان صحة وتكامل البيانات في مجموعة جداول قاعدة البيانات. شروط التكامل المرجعي هي: مفتاح أساسي (Primary Key)، مفتاح مستورد (Foreign Key)، وحيد (Unique)، اجباري (Not Null).

### المفتاح الأساسي Primary key

المفتاح الأساسي هو حقل يعرف بشكل وحيد كل تسجيلة من الجدول، وقد شرحنا سابقاً كيفية اختياره من المفاتيح الأعظمية والمفاتيح المرشحة (راجع الفصل: النموذج العلائقي لقواعد البيانات)، أما كيفية تعريفه فهي على الشكل التالي:

```
Create table Student (s_id int PRIMARY KEY, Name varchar(60), Age int);
```

الحقل s\_id هو المفتاح الأساسي للجدول Student، لا يمكن ادخال قيم مكررة أو Null في هذا الحقل. ومعرفة s\_id تحدد حتماً قيم بقية حقول الجدول.

### المفتاح المستورد Foreign key

المفتاح المستورد هو وسيلة ربط جدولين، بحيث تكون قيم الحقل المستورد في الجدول الابن هي حتماً موجودة في حقل المفتاح الأساسي في الجدول الأب، إن لم تكن Null.

يمكن فهم المفتاح المستورد من خلال المثال التالي:

Department

Id	Name
1	HR
2	Sales

## Database Architecture And Design\_CH10

```
Create table Emp (id number primary key, name
varchar(30), dept foreign key references Department(id));
```

Emp

Id	Name	dept
1	A	1
2	B	2
3	C	2
4	D	1

القيم المتاحة للحقل dept في الجدول Emp هي 1 (HR) و 2 (Sales) و Null، وأي محاولة لإدخال قيمة أخرى ستفشل مع رسالة خطأ (تم خرق قيد تكامل مرجعي).

```
Insert into Emp values (5,'E',3);
```

خطأ - فشل في تطبيق التعليمة بسبب خرق قيد تكامل مرجعي.

### الوحدانية Unique

تعني الوحدانية، عدم امكانية إدخال نفس القيمة للحقل أكثر من مرة:

```
Create table Student (s_id int Primary Key, Name
varchar(60) Unique, Age int);
```

لا يمكن ادخال قيم مكررة للحقل Name، و يمكن ادخال Null في هذا الحقل.

### اجباري Not Null

في حال كان الحقل اجباري، لا يمكن ادخال قيمة Null فيه بشكل صريح أو بشكل ضمني:

```
Create table Student (s_id int Primary Key, Name
varchar(60) not null, Age int);
```

```
Insert into Student values (6,Null,2);
```

خطأ: لا يمكن ادخال Null للحقل بشكل صريح.

```
Insert into Student (id, age) values (6,2);
```

خطأ: لا يمكن ادخال Null للحقل بشكل ضمني.

### 3. القيم الافتراضية Default

يمكن تعريف قيمة افتراضية لأحد الحقول عند انشاء الجدول، وفي هذه الحالة يمكن ادخال قيمة صريحة في هذا الحقل وإلا فإن القيمة الافتراضية بدلاً من Null ستدخل في الحقل عند إضافة تسجيلية .

```
Create table Student (s_id int Primary Key, Name
varchar(60), Age int default 30 );
```

S_id	Name	age
------	------	-----

=====

```
Insert into student(id, name) values(1,'x');
Insert into student values(2,'y',45);
```

S_id	Name	age
1	x	30
2	y	45

في التسجيلة الأولى تم ادخال القيمة الافتراضية (30) في حقل العمر، بدلاً من Null.

### 4. شروط التحقق Check:

يفيد شرط التحقق في وضع قيد على القيم المدخلة لأحد الحقول، كأن يكون ضمن مجال معين، أو أكبر من قيمة محددة.

```
Create table Student (s_id int Primary Key, Name
varchar(60), Age int check (age>18) );
```

S_id	Name	age
------	------	-----

```
Insert into student values(1,'x',12);
```

خطأ - فشل في تطبيق التعليمة بسبب خرق قيد تكامل مرجعي.

## 5. التوابع في SQL

التابع هو عبارة عن تعبير رياضي يأخذ مجموعة من قيم الدخل التي ندعوها مُعاملات، ويعيد قيمة خرج وحيدة ندعوها قيمة التابع. تتعلق قيمة التابع (أي الخرج) بمُعاملاته (أي بالدخل)، كحال التابع الذي يقوم بحساب مجموع قيم عددية.

تصنّف توابع SQL إلى:

- **التوابع التجميعية** وهي التوابع التي تأخذ كمُعاملات مجموعة من القيم وتعيد قيمة وحيدة، مثل التابع الذي يحسب مجموع أعداد حقيقية.
- **التوابع الدرجية** وهي التوابع التي تأخذ مُعاملاً وحيداً وتُعيد قيمة وحيدة، مثل تابع القيمة المطلقة لعدد حقيقي.

### التوابع التجميعية

أهم التوابع التجميعية والأكثر استخداماً هي:

التابع	استخدامه
AVG	يقوم بحساب معدل القيم لحقل معين
COUNT	يقوم بحساب عدد البيانات الخاصة بحقل معين
MIN	يقوم بإعادة القيمة الصغرى من قيم حقل معين
MAX	يقوم بإعادة القيمة العظمى من قيم حقل معين
SUM	يقوم بحساب مجموع قيم حقل معين

فيما يلي أمثلة على استخدام التوابع التجميعية:

*Employee*

Id	Name	Location	Age	Salary
1	X	1	32	15000
2	Y	1	44	12000

*Dept*

Id	Name
1	HR
2	Sales



```
Select      avg (age)
from        Employee,Dept
where       Employee.location=Dept.id
           and Dept.name='Sales';
```

النتيجة معدل أعمار موظفي المبيعات.

```
Select Dept.name, Sum (Employee.salary)
from Employee,Dept
where      Employee.location=Dept.id
group by  Dept.name
having     Count (Employee.id)>10;
```

النتيجة مجاميع رواتب الأقسام التي فيها أكثر من عشرة موظفين.

### ملاحظات:

- التابع Avg يمكن أن يأخذ أحد دخليين بالإضافة إلى الحقل المراد حساب الوسطي له، الدخليين هما All وDistinct، ويفيد الثاني منهما باستبعاد القيم المكررة عند حساب الوسطي، أما الأول فهو الخيار الافتراضي ولا يتم فيه استبعاد القيم المكررة.
- التابع Count يأخذ إحدى ثلاثة قيم للدخل هي (\*, All, Distict) بالإضافة لاسم الحقل، وتعني:
  - \*: أي عدد التسجيلات بما فيها التسجيلات التي تحوي Null كقيمة للحقل الذي نحسب مجموع ادخالاته.
  - All: تستبعد هنا القيم Null من العدد.
  - Distinct: نستبعد هنا القيم Null والقيم المكررة للحقل.

## التوابع الدرجية

التوابع الدرجية هي أربعة أنواع:

التوابع الرقمية	وهي التوابع الخاصة بالعمليات على الأرقام، مثل تابع التقريب إلى أقرب فاصلة عشرية
توابع سلاسل المحارف	وهي التوابع الخاصة بالعمليات على سلاسل المحارف، مثل تابع تحديد طول سلسلة محرفية
توابع التاريخ والوقت	وهي التوابع الخاصة بالعمليات على التاريخ والوقت، مثل تابع حساب الزمن الفاصل بين تاريخين
توابع التحويل	هي التوابع الخاصة بعملية تحويل مُعامل الدخل، من نمط بيانات إلى آخر.

- **التوابع الرقمية:** وهي التوابع الدرجية الخاصة بالعمليات على القيم الرقمية ومن أهمها التوابع التالية:

Floor	وهو التابع الذي يُقَرَّب مُعامل الدخل إلى أقرب عدد صحيح أصغر من مُعامل الدخل
Ceiling	وهو التابع الذي يُقَرَّب مُعامل الدخل إلى أقرب عدد صحيح أكبر من مُعامل الدخل
Round	وهو التابع الذي يُقَرَّب مُعامل الدخل ذو الفاصلة العشرية إلى أقرب عدد صحيح أو عدد حقيقي بدقة محددة
Abs	وهو التابع الذي يعيد القيمة المطلقة لمُعامل الدخل
Sin, Cos, Tan, Atan	وهي التوابع التي تحسب قيم ظل، تظل، جب، تجب الزاوية التي نأخذها كمُعامل دخل.
SQRT	هو التابع الذي يُعيد قيمة الجذر التربيعي لمُعامل الدخل
RAND	وهو التابع الذي يُعيد رقم عشوائي بين 0 و 1 و يستخدم مُعامل الدخل كأساس لتوليد الرقم العشوائي

- **توابع سلاسل المحارف:** هي التوابع الدرجية الخاصة بالعمليات على السلاسل المحرفية، ومن أهمها التوابع التالية:

Left()	وهو التابع الذي يُعيد جزء من السلسلة، يبتدئ من بدايتها حتى عدد محدد من المحارف
Right()	وهو التابع الذي يُعيد جزء من السلسلة، يبتدئ من نهايتها حتى عدد محدد من المحارف
Substr()	وهو التابع الذي يُعيد جزء من السلسلة، يبتدئ من موقع محدد فيها وبطول عدد محدد من المحارف
Length()	وهو التابع الذي يُعيد طول السلسلة المحرفية
Concat()	وهو التابع الذي يُستخدم لدمج أكثر من سلسلة محرفية
Lower() / Upper()	وهو التابع الذي يحول جميع محارف السلسلة إلى أحرف كبيرة أو صغيرة
Trim()	وهو التابع الذي يلغي الفراغات من بداية ونهاية السلسلة المحرفية
Instr()	وهو التابع الذي يُستخدم لتحديد موقع سلسلة جزئية ضمن سلسلة رئيسية

- توابع التاريخ والوقت: هي التوابع الخاصة بالعمليات على التاريخ والوقت ومن أهمها التوابع التالية:

التابع	استخدامه
DateDiff()	يُعيد الفرق بين تاريخين
GetDate()	يُعيد السنة، والتاريخ، واليوم، والساعة، والدقيقة، والثانية، وأجزاء الثانية
CURRENT_DATE	يعيد التاريخ الحالي الخاص بنظام إدارة قاعدة البيانات
CURRENT_TIME	يُعيد التوقيت الخاص بنظام إدارة قاعدة البيانات
CURRENT_TIMESTAMP	يُعيد التاريخ والتوقيت الخاصين بنظام إدارة قاعدة البيانات

- **توابع التحويل:** هي التوابع الخاصة بالتحويل من نمط بيانات إلى نمط آخر ومن أهمها التوابع التالية:

التابع	استخدامه
Str()	يحول قيمة الدخل العددية إلى سلسلة محرفية.
To_Number()	يحول سلسلة المحارف المارة كمعامل دخل إلى عدد.
Cast()	يحول قيمة الدخل إلى قيمة من أي نمط آخر من البيانات
Convert()	يحول قيمة الدخل إلى قيمة من أي نمط آخر من البيانات

أمثلة:

*Employee*

Id	Name	Location	Age	Salary
1	X	HR	32	15000
2	Y	Sales	44	12000

```
Select Left (Name, 1)
from Employee;
```

النتيجة الحرف الأول من اسم كل موظف.

=====

```
Select Concat (name, location)
from Employee;
```

النتيجة قائمة بأسماء الموظفين مدموجة مع أسماء الأقسام التي يعملون فيها.

## 6. الأسماء البديلة Aliases

قد تتضمن الجداول حقول متشابهة في الاسم، فمثلاً يمكن أن نجد حقل باسم Id في جدول Employee وفي جدول Dept (مثالنا السابق)، ولتمميز حقلين بنفس الاسم من جدولين مختلفين في نفس عبارة Select نستخدم الأسماء البديلة على الشكل التالي.

*Employee*

Id	Name	Location
1	X	1
2	Y	1

*Dept*

Id	Name
1	HR
2	Sales

```
Select Employee.id as e_id,Employee.name as e_name,
Dept.name as department
from Employee, Dept
where Employee.location=Dept.id;
=====
```

E_id	E_Name	Department
1	X	HR
2	Y	HR

يمكن أيضاً استخدام نفس الجدول في تعليمة Select أكثر من مرة (nested, correlated, self-join)، وفي هذه الحالة يجب تمييز ورودات الجدول عن بعضها باستخدام الأسماء البديلة، كما في المثال التالي:

*Person*

Id	Name	Gender	Age
1	X	1	32
2	Y	1	24
3	A	0	30
4	B	0	28

```
Select A.name as m1, B.name as m2
from Person A, Person B
where A.gender + B.Gender=1 and abs (A.age-B.age)<5;
=====
```

النتيجة: هي الأزواج المحتملة من ذكور واثبات بفارق عمر لا يزيد عن 5 سنوات

M1	M2
X	A
X	B
Y	B
A	X
B	X
B	y

نلاحظ أن الأزواج مكررة (ما الشرط الذي يجب أن نضيفه لإلغاء التكرار؟)

المثال السابق<sup>1</sup> يدعى بالارتباط الذاتي (self-join)، الحالات الأخرى التي نضطر فيها لاستخدام أسماء بديلة للجداول هي حالات تداخل select (nested select statement)، وحالة select المترابطة (Correlated select statement)، وسنناقش الحالتين في الفقرات التالية.

## 7. تعليمة Select المتداخلة (nested select statement)

يمكن استخدام نتيجة تعليمة Select كمصدر للبيانات تبنى عليه تعليمة Select أخرى، تساعد هذه التقنية في تسهيل فهم وقراءة تعليمات Select إلا أنها ليست الخيار الأفضل في حال كان تنفيذ المطلوب بعبارة Select واحدة مع الضم والاختيار، وذلك لدواعي سرعة التنفيذ. فيما يلي سنحلل كيفية حل مسألة باستخدام تعليمات Select متداخل، وسنرى أننا يمكن أن نحقق نفس النتيجة بتعليمة Select واحدة:

---

```
Select A.name as m1, B.name as m21
from Person A, Person B
where A.gender + B.Gender=1 and abs (A.age-B.age)<5;
```

*Company*

Id	Name	City
401	x	Damascus

*Product*

Id	Name	Producer
1	x	401

*Order*

Id	Product	Client
10	1	John
20	1	Joe

المطلوب هو المدن التي فيها شركات تنتج مواد تم طلبها من قبل John.

المواد التي طلبها John هي:

```
Select Product.id
from Product, Order
where Product.id=Order.product and Order.Client='John';
```

باستبدال أرقام المواد بأرقام مصنعها، يبقى أن نختار اسم كل شركات رقمها ضمن قائمة أرقام المصنعين التي نتجت معنا.

```
Select City from Company where Company.Id in
(
    Select Product.Producer
    from Product, Order
    where Product.id=Order.product and
    Order.Client='John'
);
```

=====

نفس النتيجة يمكن تحقيقها بتعليمة Select واحدة هي:

```
Select    Company.City
from      Product, Order, Company
where     Product.id=Order.product
          and Product.Producer=Company.id
          and Order.Client='John';
```

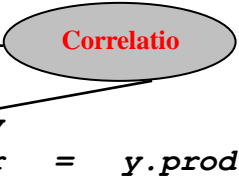
## 8. تعليمات Select المترابطة (Correlated select statement):

Product

Name	Price	Category	Producer	Year

المطلوب قائمة بالمنتجات الأعلى من أية منتجات أنتجها نفس المصنّع قبل عام 2014

```
SELECT DISTINCT name, producer
FROM      Product AS x
WHERE     price > ALL (SELECT price
                     FROM      Product AS y
                     WHERE     x.producer = y.producer AND
y.year < 2014);
```



## 9. تطبيق:

في المثال التالي سنعرض بعض الاستخدامات الخاصة لتعليمة Select:

Person

Id	FName	LName	Gender	Nationality	Age
1	John	Smith	1	2	32
2	Adam	Sandler	1	1	45
3	Mary	Clair	0	1	40
...	...	...	...	...	...



=====

### Nationality

Id	Name
1	American
2	Frensh
...	...

=====

### ContactMedia

Id	Media
1	Mobile
2	E-Mail
3	Phone
...	...

=====

### PersonContact

Id	Person	ContactMedia	ContactValue
1	1	1	00963944111222
2	1	2	<a href="mailto:xxx@gmail.com">xxx@gmail.com</a>
...	...	...	...

=====

في مثالنا الجنسية Nationality تصدر مفتاحها إلى الجدول Person (ارتباط واحد لعدة)، وطريقة الاتصال ContactMedia ترتبط مع جدول Person من خلال جدول وسيط هو PersonContact (يمكن أن يكون لنفس الشخص أكثر من طريقة للاتصال به).

المطلوب:

- قائمة بفرق العمل المحتملة، علماً أن كل فريق يتألف من شخصين من جنسية واحدة و فارق العمر بينهما لا يتجاوز 5 سنوات ؟
- قائمة بالجنسيات التي يحملها على الأقل شخصين مسجلين في قاعدة البيانات ؟
- وسيلة الاتصال الأكثر استخداماً (المدخل بياناتها للأشخاص المسجلين) ؟
- وسطي أعمار الأشخاص المسجلين في كل جنسية ؟

## 10. المراجع:

- <http://www.studytonight.com/dbms/select-query>

## 11. تدریبات:

## Invoice

<b>Id</b>	<b>Serial</b>	<b>Date</b>	<b>Client</b>
1	1	1-1-2015	20
2	2	2-1-2015	30
3	3	1-1-2015	10
4	4	4-2-2015	20
5	5	4-2-2015	10

## Client

<b>Id</b>	<b>Name</b>
10	C 1
20	C 2
30	C 3

## Material

<b>Id</b>	<b>Name</b>	<b>Price</b>
1	Printer	15000
2	Monitor	30000
3	Case	2500
4	MB	22000

## Invoice\_Item

<b>Id</b>	<b>Invoice</b>	<b>Material</b>	<b>QTY</b>
1	1	1	1
2	1	2	1
3	1	4	1
4	2	2	2
5	2	3	2

- نتيجة تطبيق جملة التعليمات التالية:

```

Select      Id,Name
From        Material
Where       Id not in ( Select material from invoice, invoice_item
                        where invoice.id=invoice_item.invoice
                        and invoice.date <>'1-1-2015');
```

هي:

1. المواد التي تم شراؤها في 1-1-2015.
2. المواد التي لم يتم شراء كميات منها.
3. المواد التي لم يتم شراء أي كمية منها بتاريخ 1-1-2015.
4. المواد المتضمنة في الفاتورة رقم 1 و الفاتورة رقم 3.

الإجابة: (3)

- نتيجة التعليمة التالية:

```

Select      Id,Name
From        Material
Where       Price > ( Select Max(price) from material, invoice_item
                        where invoice_item.material=material.id
                        );
```

هي:

1. المادة الأعلى سعراً.
2. المواد التي سعرها أعلى من سعر أية مادة ظهرت في إحدى الفواتير.
3. المادة المباعة و الأعلى سعراً.
4. المواد التي لم يتم البيع منها.

الإجابة: (2)

• نتيجة التعليلة التالية:

```
SELECT      Name, Sum(price*qty) AS TotalSales
FROM        Material, Invoice, Invoice_item
WHERE

            Material.id=Invoice_item.material
            And invoice_item.invoice=invoice.id
            And date > '1/1/2015'

GROUP BY    name
```

هي:

1. إجمالي الفواتير بعد تاريخ 2015-1-1
2. إجمالي المبيعات حسب المادة بعد تاريخ 2015-1-1.
3. مجموع أسعار المواد المباعة بتاريخ 2015-1-1
4. مبيعات المواد مجمعة حسب التاريخ.

الإجابة: (2)