الباب الثاني

أساسيات Query

٤. أساسيات لغة الإستعلام QUERY

٥. استخدام المعاملات (Operators)

٦. استخدام الدوال المعرفة مسبقا Built-in Functions

٧. ترتيب السجلات وتجميعها

الفصل الرابع

أساسيات لغة الاستعلام Query

*الترجمة الحرفية لكلمة **Query** سوف تحصر استخدامات **Query** في الاستعلام أو البحث أو استخراج معلومات من قواعد البيانات . الحقيقة أن لها استخداما أوسع من ذلك كما ستعرف في هذا الكتاب.*
*في هذا الفصل سوف نتفهم أساسيات أو استخدامات **Query**.*
بالانتهاء من هذا الفصل ستكتسب المعارف وتتدرب على المهارات التى تجعلك قادرا على:

- *أقسام لغة **SQL***
- *فهم ماهية **Query***
- *استخدام جملة **Select***
- *الشكل العام الجملة **SQL***
- *كتابة أول جملة **Query***

في هذا الكتاب سوف نستخدم العديد من الأمثلة ، ولكي تستطيع تنفيذ هذه الأسئلة فانك تحتاج إلى تثبيت نظام قواعد البيانات الذي تستخدمه ثم تقوم بإضافة بعض الملفات من خلال الـ **CD-ROM** إلى النظام بعد تثبيته.
الطريقة الأخرى التي أحبذها لك عزيزي القارئ أن تبحث في هذا الكتاب عن المعلومة التي تريدها، فإذا احتجت إلى تكوين قاعدة بيانات ، ابحث عن كيفية تكوينها في الفصل المناسب وهكذا أيضا عندما تحتاج لإضافة جدول **Table** ابحث في الكتاب عن إضافة الجداول ، وهذا سوف يعودك على طريقة البحث والتجربة والخطأ.
تعودنا فى كتب قواعد البيانات التى قدمناها للقارئ العزيز أن نبدأ بإنشاء الجداول ثم إدخال البيانات ثم معالجة البيانات سواء بالإضافة أو الحذف أو التعديل ثم نشرح كيفية الاستعلام عن البيانات المخزنة بقاعدة البيانات

وفى هذا الكتاب نستسمحك عزيزى القارئ فى الخروج على هذا النمط وذلك للضرورة التى تمليها طبيعة المادة التى يشرحها هذا الكتاب.

لقد وقفت طويلا أمام أمثل طريقة لترتيب فصول هذا الكتاب وكان أمامى خياران:

إما أن أبدأ بشرح كيفية إنشاء قواعد البيانات ابتداء من تكون الجداول وإدخال بياناتها قبل شرح كيفية الاستعلام عن بياناتها. أى قبل شرح استخدام أوامر **Query** وهى العمود الفقرى فى هذا الكتاب. أو أن أبدأ بشرح كيفية الاستعلام عن بيانات موجودة (كيفية استخدام **Query**) وفى هذه الحالة لابد أن يكون هناك قاعدة بيانات وجداول لكى نستعلم عن بياناتها.

وقد أخذت الأخير على أن أضع لك قواعد بيانات وجداولها فى مجلد خاص نجده على القرص المرفق مع هذا الكتاب على أن تجرب الأمثلة الواردة في الشرح باستخدام قواعد البيانات والجداول المعدة سلفا.

ولكن لكى تستطيع متابعة الأمثلة وتجربتها لابد أن تنسخ قواعد البيانات إلى جهازك ليمكنك التعامل معها. (راجع الفصل السابق)

أقسام لغة SQL

من المعروف أن لغة **SQL** عبارة عن مجموعة من الأوامر التي تتعامل مع قواعد البيانات. وتمكنك من إنشاء كائنات قاعدة البيانات كالجداول ومعالجة بياناتها وعرض محتوياتها وتتكون لغة **SQL** من ثلاثة أقسام:

١. لغة تعريف البيانات **Data Definition Language (DDL)**.

وتختص بإنشاء أو تعديل أو حذف كائنات قاعدة البيانات، مثل الجداول وما تحتويه من أعمدة وفهارس، ومثل العلاقات والقيود **Constrains**.

ومن أشهر الأوامر التي تستخدمها الأمر

- **Create**
- **ALTER**
- **DROP**

٢. لغة معالجة البيانات **Data Manipulation Language (DML)**.

تشتمل علي الأوامر التي تتيح عرض البيانات والاستعلامات ومن أشهر هذه الأوامر الأمر **Select** وكما تشتمل علي مجموعة من الأوامر التي تتيح إضافة وحذف وتعديل البيانات ومن أشهرها

- **Select**
- **Insert**
- **Update**
- **Delete**

٣. لغة التحكم في البيانات **Data Control Language**

وتختص بالتحكم في العمليات **Transactions** وحقوق المستخدمين ومن أشهر الأوامر التي تستخدمها:

- **SET TRANSACTION**
- **BEGIN**
- **COMMIT TRANSACTION**
- **ROLL BACK TRANSACTION**

فهم ماهية Query

عندما تريد الحصول على بيانات من قواعد البيانات عن الموظفين الذين تجاوز عمرهم ٤٠ سنة وتم تعيينهم منذ أكثر من ١٠ سنوات بشرط أن يكونوا متزوجين فماذا تفعل ؟ يمكن أن يقوم البعض بطباعة كشوف كل الموظفين ثم تبدأ رحلة البحث داخل الحقول الخاصة بالسن وتاريخ التعيين والحالة الاجتماعية وكل من تنطبق عليه الشروط تقوم بوضع علامة على اسمه في الكشف ثم تقوم بنقل هذه المعلومات باليد في ورق خارجي ؟ بدلا من ذلك ، وبسرعة وبجملة بسيطة من خلال **SQL** يمكنك استخراج تلك البيانات بدقة وذلك باستخدام **Query** أي أن **Query** عبارة عن سؤال بسيط لقواعد البيانات بما تريده وبسرعة تتم الاجابة عليك .

استخدام جملة Select

سنشرح فيما يلي مفردات جملة **Select** باعتبارها هى قلب لغة **SQL** وبمعنى آخر سوف نتعلم كيف نكتب صيغة أو نص **(Syntax)** صحيح يمكن أن يفهمه **SQL** ويقوم بتنفيذه؟

ولنأخذ مثالا عن طريقة كتابة الاستعلام. هب أننا نريد أن نبحث في أحد جداول قواعد بيانات المدرسين عن مدرس أسمه "Saeed" فماذا نكتب ؟

```
SELECT NAME
FROM TEACHERS
WHERE NAME = 'Saeed';
```

سوف نشرح بالتفصيل فيما بعد المصطلحات الواردة في هذه الجملة وإنما أوردناها هنا لتوضيح قواعد كتابة جملة SQL

ماذا نلاحظ ؟ في المثال السابق نلاحظ أن كل الحروف كبيرة فيما عدا كلمة Saeed ، ولأن الشكل العام لجملة (SQL) لا يفرق بين الحروف الكبيرة والصغيرة فيمكن كتابة نفس المثال كما يلي :

```
select name
from teachers
where name = 'Saeed';
```

ماذا نلاحظ ؟ نفس الجملة ولكن الحروف صغيرة ، أيضا فيما عدا كلمة Saeed ، فماذا عن هذه الكلمة ؟ أود أن تلاحظ عزيزي القارئ أن الجملة كلها يمكن كتابتها بأي حروف سواء كان كبيرة أم صغيرة ولكن كلمة Saeed هي كلمة مخزنة داخل قواعد البيانات ونحن نريد أن نبحث عن هذه الكلمة فيجب أن يتم البحث بنفس الطريقة التي تم تخزينها بها داخل قواعد البيانات وبالتالي فان 'SAEED' أو 'saeed' أو 'Saeed' تختلف اختلافا بينا فيما بينها ، لذلك عند البحث عن معلومة داخل قواعد البيانات يجب تحري الدقة في الطريقة المكتوبة بها هذه المعلومة لكي يتم البحث صورة صحيحة .

الأمر الآخر هو المسافات بين أوامر أو جمل Query ، فرغم أنها غير هامة مثل :

```
SELECT NAME  FROM TEACHERS WHERE NAME = 'Saeed';
```

فان هذه الجملة صحيحة النص وسوف يتم تنفيذها بدون مشاكل ، ولكن هذه جملة بسيطة فهل لك أن تخبرني كم من الوقت سوف تأخذه لكي تبحث عن كل جزئية ، فما بالنا لو كانت ال Query المكتوبة هي عدد كبير من الجمل ، لذلك أنصحك عزيزي أن تستخدم الأسلوب الأسهل تنظيما بحيث تترك مسافات وتكتب كل أمر في سطر مستقل .

ولتحليل الجملة السابقة نجدها تبدأ بأمر **select** وتنتهي بالفاصلة المنقوطة ولنر مدلول الجملة كما يلي :

SELECT اختر

NAME حقل اسمه

FROM من

TEACHERS جدول اسمه

WHERE بشرط أن يكون

NAME = 'Saeed' حقل**Name** يساوي كلمة **" Saeed "**

لاحظ أن النصوص المكتوبة في جملة **Query** السابقة هي نصوص بسيطة ومقروءة بدلالة واضحة .

استخدمنا في المثال السابق أشهر وأعم ثلاثة كلمات أساسية في **SQL** وهي **Select, from, where** وسوف تجد أن هذه الكلمات دائمة الاستخدام مهما اختلف نظام قواعد البيانات المستخدم.

الشكل العام الجملة SQL

الشكل العام أو **Syntax** لجملة **Query** باستخدام كل من **select** و **From** ببساطة كما يلى :

```
Select <columns names>
From <table>
```

حيث لابد من ذكر أسماء الحقول التى نريد الاستعلام عنها أو إظهارها بعد كلمة **Select** مباشرة. ولابد أيضا من ذكر اسم الجدول الذي يحتوى على هذه الحقول بعد كلمة **From** تسمى كل من **Select** و **From** كلمات أساسية أو **Keywords** واستخدامهما معا، يسمى جملة **SQL** كما فى المثال السابق.

قبل أن نتوسع في كتابة **Query** وفي استخدام أهم كلمتين وهما **Select** و **From** دعنا نتوسع في فهم استخداماتهم أو طريقة كتابتهم ، لعلنا في المثال السابق استخدمنا جدول المدرسين **Teachers** السابق تكوينه كما أننا سوف نستخدم في الأمثلة التالية

جداول سبق تكوينها existing tables . وسوف نتعلم طريقة تكوين الجداول ضمن طريقة تكوين قواعد البيانات نفسها في الفصل العاشر ولكننا لكي نستفيد من الأمثلة التالية قمنا بتكوين الجدول التالي واسمه Teachers ويحتوي على المعلومات التالية :

NAME	AGE	GYEAR	SPECIALITY	REM
Ali Hassan	30	1995	Math	Senior
Hassan Ali	35	1990	English	Senior
Mohssen	28	1997	Arabic	Prof
Amir	27	2000	Computer	
	Engineer			
Mohammad	25	2003	Physics	Junior
Assad	26	2000	French	Level II

هكذا يبدو الجدول عند تخزينه في نظام قواعد البيانات Oracle

بينما يبدو شكله كالتالي في نظام قواعد البيانات MySql

NAME	AGE	GYEAR	SPECIALITY	REM
Ali Hassan	30	1995	Math	Senior
Hassan Ali	35	1990	English	Senior
Mohssen	28	1997	Arabic	Prof
Amir	27	2000	Computer	Engineer
Mohammad	25	2003	Physics	Junior
Assad	26	2000	French	Level II

من الواضح أن الجدول في أوراكل أكثر بساطة وبدون تسطير بينما الجدول في MySql يتم اظهارة في شكل عملي أكثر ، لا تنسى أن هذا الجدول خاص بالمدرسين Teachers Table وسوف نستخدم هذا الجدول في الأمثلة التالية .

الأمثلة التالية سوف نكتب نصوص SQL مرة باستخدام oracle والأخرى باستخدام MySql والفرق بينهما هو تلك العلامة في حالة أوراكل

SQL>

أما في حالة MySql فسوف تجد علامة النظام التالية

MySql>

وذلك لعموم الفائدة .

كتابة أول جملة Query :

الجملة في أوراكل

إذا كان المطلوب إظهار البيانات الموجودة في جدول Teachers فإن الجملة المناسبة تكون كما يلي:

```
SQL> SELECT * FROM TEACHERS;
```

النتيجة

NAME	AGE	GYEAR	SPECIALITY	REM
Ali Hassan	30	1995	Math	Senior
Hassan Ali	35	1990	English	Senior
Mohssen	28	1997	Arabic	Prof
Amir	27	2000	Computer	Engineer
Mohammad	25	2003	Physics	Junior
Assad	26	2000	French	Level II

6 Rows selected <--- هذه الرسالة تظهر في نهاية تنفيذ الأمر

الجملة في *MySql*

```
MySql> SELECT * FROM TEACHERS;
```

النتيجة

NAME	AGE	GYEAR	SPECIALITY	REM
Ali Hassan	30	1995	Math	Senior
Hassan Ali	35	1990	English	Senior
Mohssen	28	1997	Arabic	Prof
Amir	27	2000	Computer	Engineer
Mohammad	25	2003	Physics	Junior
Assad	26	2000	French	Level II

6 Rows in set (0.19 sec) <--- هذه الرسالة تظهر في نهاية تنفيذ الأمر

لاحظ أن استخدام النجمة ' * ' في جملة SELECT تعني اختيار كافة حقول وبيانات الجدول بدون تحديد لأي معلومة أو حقل لذلك فان النتيجة كانت نفس الجدول السابق تخزينه مع اختلاف الشكل وفقا لنظام قواعد البيانات المستخدم .

إنهاء جملة SQL

في معظم نظم قواعد البيانات يتم البحث عن الفاصلة المنقوطة (;) في نهاية أي جملة قبل تنفيذها وإلا كانت النتيجة خطأ كما في نظام أوراكل ، بينما لا تهتم بعض قواعد البيانات بهذه الفاصلة المنقوطة كدلالة على انتهاء الجملة مثل Microsoft Query أو Borland حيث أنك تقوم بكتابة الجملة داخل محرر نصوص Editor خاص بتلك النوعية من Query .

تغيير ترتيب الحقول في Query

يمكننا عند كتابة الجملة أن ننص على ترتيب معين للمخرجات Output أو ما نسميه (النتيجة) بعد تنفيذ الجملة ومثال على ذلك :

أوراكل

```
SQL> SELECT NAME, SPECIALITY, REM, GYEAR, AGE
FROM TEACHERS;
```

النتيجة

NAME	SPECIALITY	REM	GYEAR	AGE
Ali Hassan	Math	Senior	1995	30
Hassan Ali	English	Senior	1990	35
Mohssen	Arabic	Prof	1997	28
Amir	Computer	Engineer	2000	27
Mohammad	Physics	Junior	2003	25
Assad	French	Level II	2000	26

6 Rows selected <--- هذه الرسالة تظهر في نهاية تنفيذ الأمر

MySql

```
MySql> SELECT SPECIALITY, REM, GYEAR, AGE, NAME
    ->FROM TEACHERS;
```

النتيجة Output

SPECIALITY	REM	GYEAR	AGE	NAME
Math	Senior	1995	30	Ali Hassan
English	Senior	1990	35	Hassan Ali
Arabic	Prof	1997	28	Mohssen
Computer	Engineer	2000	27	Amir
Physics	Junior	2003	25	Mohammad
French	Level II	2000	26	Assad

6 Rows in set (0.19 sec) <--- هذه الرسالة تظهر في نهاية تنفيذ الأمر

اختيار بعض الحقول فقط

عندما نريد اختيار بعض الحقول فقط دون باقي حقول الجدول Table ماذا نفعل ؟ لاحظ الأمثلة التالية :

في حالة نظام أوراكل

```
SQL> SELECT NAME, SPECIALITY FROM TEACHERS;
```

ومعني هذه الجملة استخراج البيانات الموجودة في حقل NAMEوحقل SPECIALITY فقط النتيجة

```
NAME          SPECIALITY
----------    ----------
Ali Hassan    Math
Hassan Ali    English
Mohssen       Arabic
Amir          Computer
Mohammad      Physics
Assad         French
```

6 Rows selected <--- هذه الرسالة تظهر في نهاية تنفيذ الأمر

*في حالة نظام **MySql***

```
MySql> SELECT AGE, NAME FROM TEACHERS;
```

النتيجة

AGE	NAME
30	Ali Hassan
35	Hassan Ali
28	Mohssen
27	Amir
25	Mohammad
26	Assad

6 Rows in set **(0.19 sec)** <--- هذه الرسالة تظهر في نهاية تنفيذ الأمر

لاحظ أن جملة **Select** باستخدامها العام يمكن أن تكون مساعدة على فهم **SQL** وكيفية عمله كما في الجمل التالية :

```
SQL> select * from teachers
SQL> select * from employees
SQL> select * from students
```

ما يجب علينا ملاحظته الآن أن الاختلاف بين الجمل السابقة هو فقط في اسم الجدول **TABLE** الذي نختار منه البيانات بينما الاختيار يكون لكامل محتويات الجدول **SELECT ***
فالجملة هي **SELECT** والنجمة " * " تشير الى كل محتويات الجدول **TABLE-NAME** مثل **Teachers** أو **STUDENTS** كما في الجمل السابقة .

قبل أن نبدأ

تعلمنا في هذا الفصل أشهر جملة استعلام أو استخراج معلومات من قواعد البيانات وذلك باستخدام **SQL**. وفي هذا الفصل تعلمنا عدداً من أشكال الجملة والنتائج التي نحصل عليها في كل مرة .والسؤال الذي يتبادر إلى الذهن حتى الآن هو :
نحن تعلمنا كيف نستعلم عن بيانات في قاعدة بيانات موجودة ، ولكن كيف يتم انشاء قاعدة البيانات وتخزين البيانات فيها ؟

سوف نتعلم كيفية إنشاء الجداول في الفصول التالية من هذا الكتاب وسوف نتدرب على إنشاء الجداول Create Tables وعلى إدخال بيانات داخل الجداول في فصول تالية ولذلك فإن كل ما تعلمناه في السابق هي أمثلة نظرية ولا نعلم اذا كانت ستعمل بطريقة صحيحة أو أننا أخطأنا في كتابتها .

في الفصول التالية سوف نتعلم كيف نكتب الجمل المختلفة سواء داخل قاعدة البيانات أو بالاتصال بقاعدة البيانات عن طريق أوامر أخرى .

والسؤال الآخر هل ينحصر استخدام SQL في كتابة الأوامر والتعليمات مباشرة من خلال نافذة في قاعدة البيانات ؟

لا ، فهذه احدى الطرق ، ولكن بالطبع هناك العديد من الطرق العملية لاستخدام قواعد البيانات من خلال البرامج بحيث نكتب جمل أو نصوص SQL من داخل البرامج التي نقوم بكتابتها بلغات مختلفة . فيمكن كتابة تعليمات SQL والارتباط بقاعدة بيانات من خلال لغات عديدة مثل C, C++, C#, Visual Basic كما أن هناك مكتبات برامج تأتي مع قواعد بيانات مثل ORACLE, SYBASE بحيث يمكنك كتابة التعليمات من خلال قواعد البيانات . وقد قامت ميكروسوفت بتطوير تطبيقات لوضع تعليمات SQL وتنفيذها من خلال لغات البرمجة ويطلق على هذا النظام من ميكروسوفت Application Programming Interface API وقد سهل هذا كثيرا على المبرمجين .