

# قواعد البيانات وطرق تصميمها

Databases and how to design it

---

المحاضر: حسام يونسو

- المخرجات المتوقعة من الدرس
- مقدمة
- تثبيت بيئة قواعد البيانات على الحاسوب
- تشغيل نظام قاعدة البيانات MySQL
- مفهوم phpMyAdmin
- واجهة phpMyAdmin
- فائدة استخدام phpMyAdmin
- أنواع البيانات في MySQL
- إنشاء قاعدة البيانات
- § إنشاء جدول في قاعدة البيانات
- § تصدير قواعد البيانات
- § استيراد قواعد البيانات
- § العمل في محرر MySQL
- § شرح واستخدام الأمر SELECT
- § التعامل مع بيانات متراابطة وجداول علائقية JOIN
- § أنواع JOIN في MySQL
- § أمثلة عملية
- § الخاتمة

## المَخْرُجَاتُ المُتَوقَّعَةُ مِنَ الدُّرْسِ

في هذه الدورة سنتعلم كيفية استخدام MySQL لإدارة البيانات بطريقة احترافية باستخدام بيئة محلية تعتمد على برنامج XAMPP، والذي يوفر خادم MySQL وواجهة phpMyAdmin التي تسهل إنشاء قواعد البيانات والجداول وإدارتها دون الحاجة إلى أوامر معقدة. سنبدأ بفهم المفاهيم الأساسية لقواعد البيانات، ثم ننتقل إلى إنشاء قواعد بيانات وربط الجداول، وكتابة استعلامات SQL أساسية مثل SELECT وINSERT وUPDATE وDELETE، بالإضافة إلى استعلامات شرطية باستخدام WHERE، مع التركيز على الجانب العملي.



تُعد قواعد البيانات من أهم مكونات أنظمة المعلومات الحديثة، إذ

ستخدم لتخزين البيانات وتنظيمها واسترجاعها بطريقة فعالة وآمنة.

في هذا الملف، نقدم دليلاً عملياً لتعلم تصميم قواعد البيانات

باستخدام MySQL من خلال أداة phpMyAdmin، ضمن بيئة

محلية مبنية على XAMPP، لتوفير تجربة تعليمية سهلة وقابلة

للتطبيق.

# تثبيت بيئة قواعد البيانات على الحاسوب

XAMPP هو برنامج مجاني يقوم بتنصيب حزمة من الأدوات على جهازك، أهمها:



المكون	وظيفته
Apache	خادم ويب محلي (يشغل phpMyAdmin)
MySQL	قاعدة البيانات
phpMyAdmin	واجهة رسومية لإدارة قواعد البيانات بسهولة

# خطوات تثبيت XAMPP

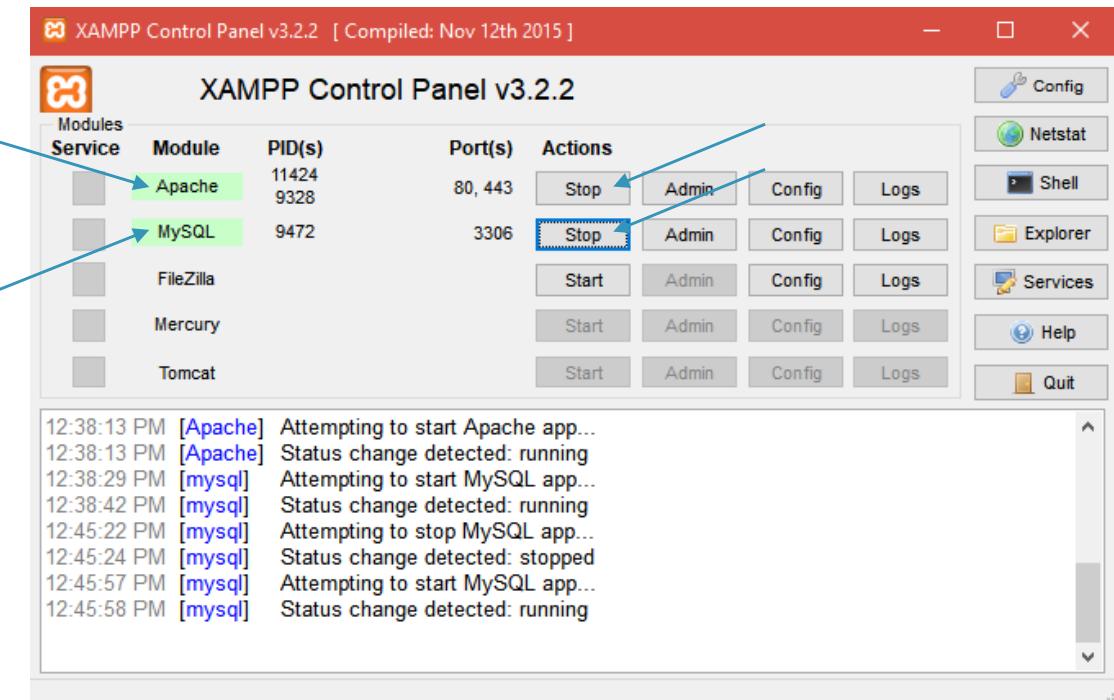
## 2. تثبيت XAMPP

- بعد التحميل، شغّل ملف التثبيت.
- أثناء التثبيت، تأكد من تحديد  هذه الخيارات:
  - Apache •
  - MySQL •
  - phpMyAdmin •
- اضغط على زر **Finish** حتى النهاية، ثم **Next**

## 1. تحميل XAMPP

- ادخل إلى الموقع الرسمي:  
<https://www.apachefriends.org>
- اختر نظام التشغيل الخاص بك ( )  
Windows / .(macOS / Linux)
- اضغط على زر **Download**

# تشغيل XAMPP



## افتح .XAMPP Control Panel

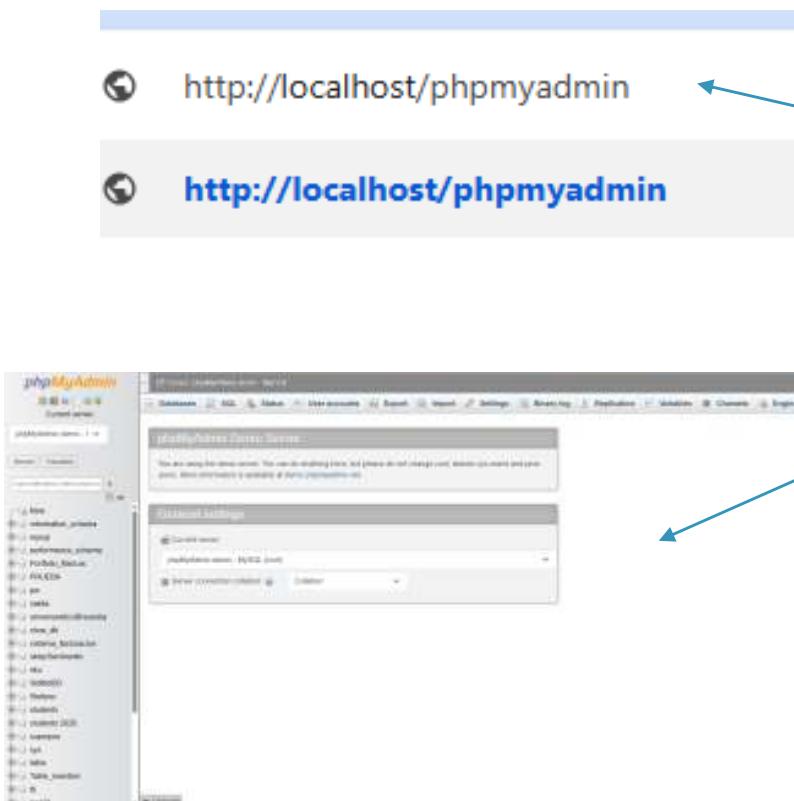
- اضغط على Start أمام:

- Apache

- MySQL

- إذا تحول اللون إلى أخضر، فهذا يعني أن الخدمة تعمل.

# تشغيل phpMyAdmin



- بعد تشغيل MySQL و Apache و Apache:
  - افتح المتصفح واكتب:  
<http://localhost/phpmyadmin>
  - ستظهر لك واجهة phpMyAdmin لإدارة قواعد البيانات.
- ملاحظة مهمة:
  - XAMPP يعمل على جهازك المحلي فقط (localhost).
  - قاعدة البيانات MySQL تُدار من خلال phpMyAdmin دون الحاجة إلى أوامر معقدة.

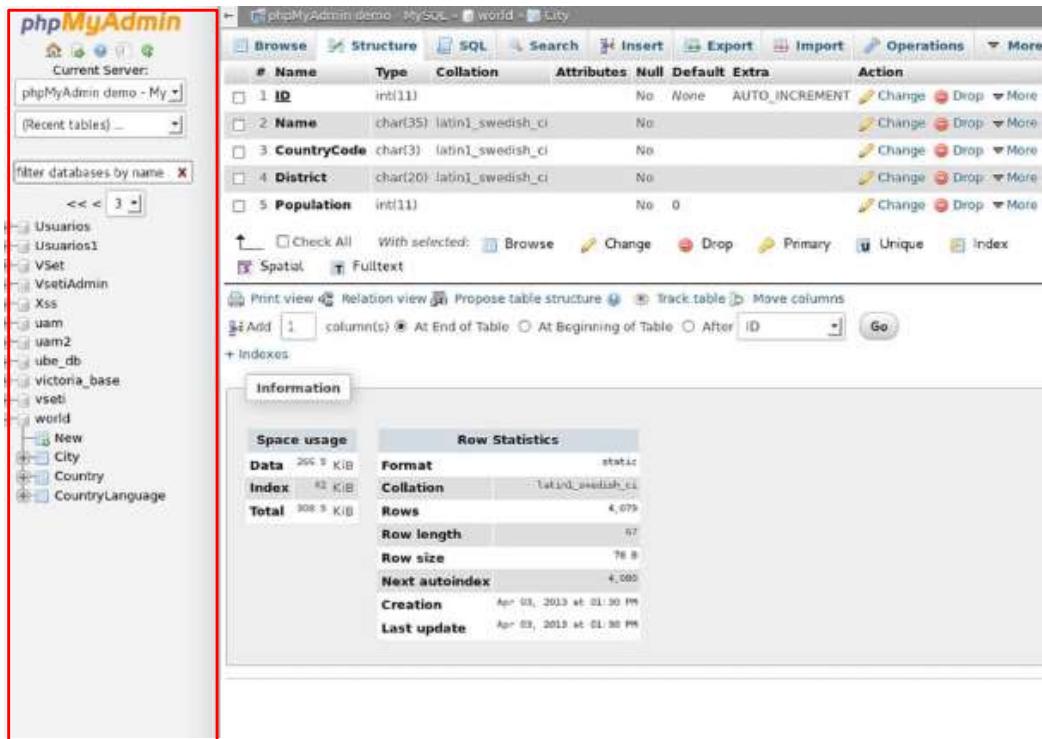
# شرح واجهة phpMyAdmin (UI)



ما هي phpMyAdmin هي واجهة رسومية على المتصفح تتيح لك إدارة قواعد بيانات MySQL بسهولة دون الحاجة لكتابة أوامر برمجية.

تأتي phpMyAdmin مدمجة تلقائياً مع XAMPP، وتُستخدم لتسهيل إنشاء قواعد البيانات، الجداول، إدخال البيانات، تنفيذ الاستعلامات، وغير ذلك.

# أقسام واجهة phpMyAdmin

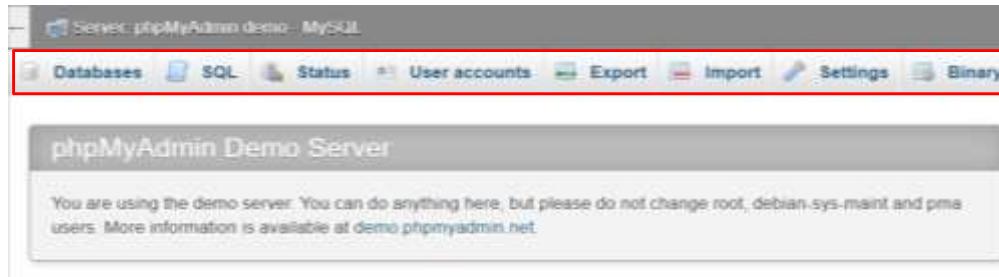


The screenshot shows the phpMyAdmin interface. On the left, there is a sidebar with a red border containing a tree view of databases and tables. The main area displays the structure of a table named 'City' with columns: ID, Name, CountryCode, District, and Population. Below the table structure, there is an 'Information' panel showing space usage and row statistics for the 'City' table.

## 1. الشريط الجانبي (Sidebar)

- يعرض جميع قواعد البيانات الموجودة.
- يمكنك النقر على أي قاعدة بيانات لرؤيتها الجداول داخلها.
- هناك زر + New لإنشاء قاعدة بيانات جديدة.

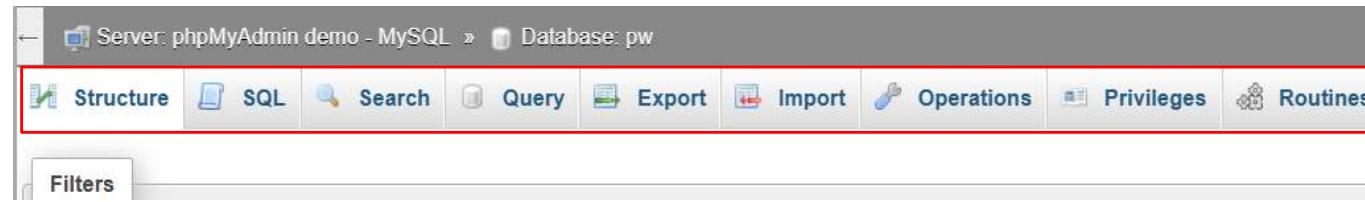
# أقسام واجهة phpMyAdmin



## 2. الشريط العلوي (Tabs)

الوظيفة	التبوب
استعراض وإنشاء قواعد بيانات.	Database
كتابة وتنفيذ أوامر SQL يدوياً.	SQL
يعرض إحصائيات السيرفر.	Status
استيراد قواعد بيانات (عادة من ملفات .sql).	Import
تصدير قاعدة بيانات إلى ملف.	Export
إنشاء وتعديل المستخدمين (غالباً لا نستخدمه كمبتدئين).	User Accounts

# أقسام واجهة phpMyAdmin



## الوظيفة

عرض تصميم الجدول (أسماء الأعمدة، الأنواع، المفاتيح).

عرض بيانات الجدول.

إدخال بيانات جديدة إلى الجدول.

تشغيل استعلامات على هذا الجدول بالذات.

البحث داخل الجدول.

لتعديل اسم الجدول، النسخ، الحذف، أو تغيير إعداداته.

## 3. عند الدخول إلى قاعدة بيانات:

ستظهر لك تبويبات جديدة، أهمها:

### التبوب

Structure

Browse

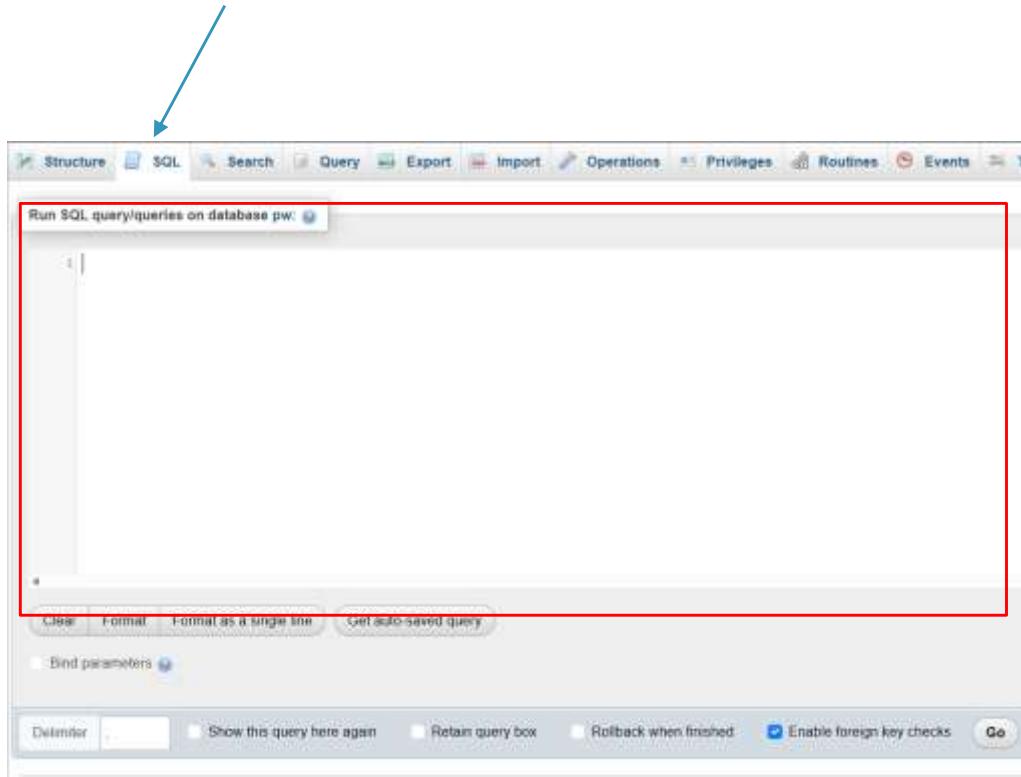
Insert

SQL

Search

Operations

# أقسام واجهة phpMyAdmin



4. محرر SQL (SQL Editor) من خلال تبويب SQL يمكنك كتابة أي استعلام (مثل SELECT, ...INSERT...) وتشغيله، والنتائج ستظهر أسفل الصفحة.

مثال:

سيعرض كل البيانات الموجودة في جدول العملاء.

```
SELECT * FROM Customers;
```

# فوائد استخدام phpMyAdmin



لماذا phpMyAdmin مفيدة جداً للمبتدئين؟

- لا تحتاج إلى حفظ أوامر كثيرة في البداية.
- يمكنك تنفيذ استعلامات SQL ومشاهدة النتائج مباشرة.
- تعرض الجداول بشكل رسومي واضح.
- مثالية للتعلم والتجريب، خاصة عند استخدام قواعد بيانات .Northwind مثل.

# أنواع البيانات الأكثر استخداماً في MySQL

## 1. الأنواع العددية (Numeric Types)

تُستخدم لتخزين الأرقام، سواء كانت صحيحة أو تحتوي على فاصلة عشرية.

النوع	الوصف	مثال استخدام
INT	عدد صحيح (رقم بدون فاصلة عشرية)	رقم الطالب، عدد الطلبات
TINYINT	عدد صغير (من -128 إلى 127)	للحالة (0 = موقوف، 1 = نشط)
BIGINT	عدد صحيح كبير جداً	معرفات ضخمة (مثل رقم حساب)
DECIMAL(x,y)	= العدد الكلي، = x رقم بفاصلة عشرية ثابتة، بعد الفاصلة	أسعار، رواتب دقيقة
FLOAT, DOUBLE	أرقام عشرية بدقة تقريبية	النسب، معدلات الأداء

# أنواع البيانات الأكثر استخداماً في MySQL

## 2. الأنواع النصية (String Types)

تُستخدم لتخزين النصوص مثل الأسماء والعناوين والإيميلات.

النوع	الوصف	مثال استخدام
VARCHAR(n)	سلسلة نصية بطول متغير	اسم الطالب، البريد الإلكتروني
CHAR(n)	سلسلة نصية بطول ثابت	رموز ثابتة (مثل رمز دولة)
TEXT	نص طويل	وصف منتج، ملاحظات
ENUM	قائمة خيارات محددة	الجنس (ذكر/أنثى)، الحالة

# أنواع البيانات الأكثر استخداماً في MySQL

## 3. الأنواع الزمنية (Date and Time Types)

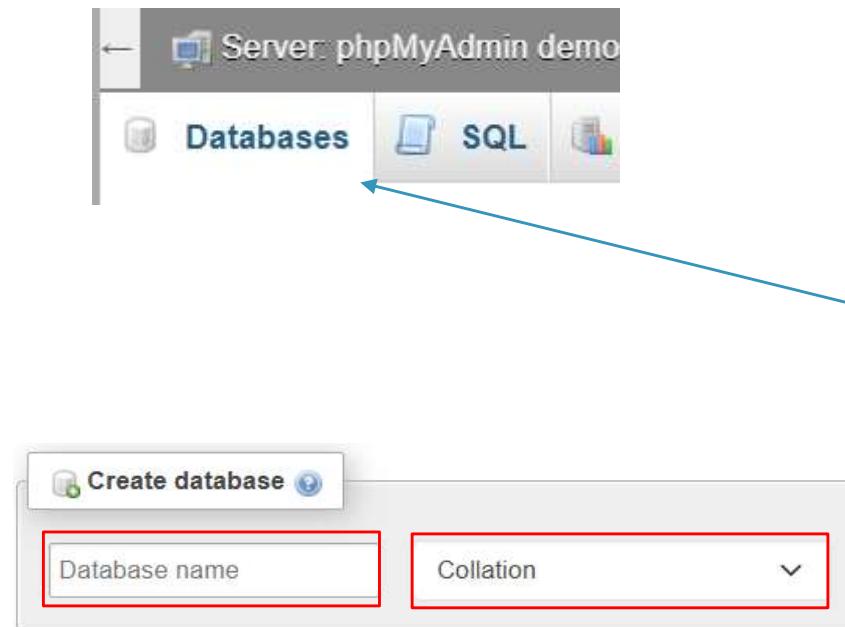
تُستخدم لتخزين التواريخ والأوقات.

النوع	الوصف	مثال استخدام
DATE	تاريخ فقط (YYYY-MM-DD)	تاريخ الميلاد، تاريخ الطلب
DATETIME	تاريخ ووقت معًا	وقت التسجيل، وقت الدخول
TIME	الوقت فقط (HH:MM:SS)	مدة، توقيت معين
TIMESTAMP	ويحدث تلقائياً DATETIME مثلًّا	وقت آخر تعديل مثلاً

## نصائح عند اختيار نوع البيانات

1. استخدم VARCHAR بدلاً من TEXT عندما تعرف الحد الأقصى للطول.
2. استخدم ENUM للحالات المحدودة (ذكر/أنثى، نشط/موقوف).
3. استخدم DECIMAL بدلاً من FLOAT للبيانات المالية.
4. حدد NOT NULL إذا كان العمود لا يجب أن يقبل القيمة الفارغة.
5. استخدم AUTO\_INCREMENT للأعمدة الأساسية الرقمية.

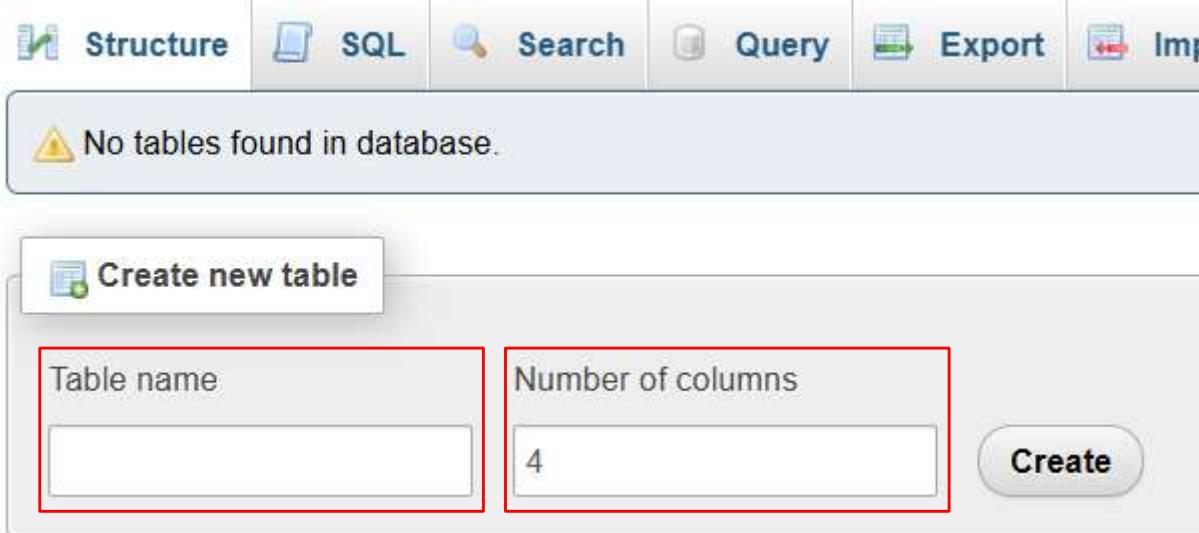
# إنشاء قاعدة بيانات في phpMyAdmin



- أولاً: إنشاء قاعدة بيانات
- الخطوات:
  - افتح المتصفح واذهب إلى:  
<http://localhost/phpmyadmin>
  - من الشريط العلوي اضغط على تبويب **Databases**
  - ستجد حقلًا بعنوان **Create database**
  - أدخل اسم القاعدة، مثلاً: **my\_school**
  - من القائمة الجانبية اختر الترميز: **ic\_lareneg\_4mb8utf**
  - اضغط على زر **Create**.

# إنشاء جداول في phpMyAdmin

## ثانياً: إنشاء جدول داخل قاعدة البيانات



- بمجرد إنشاء القاعدة، سيتم تحويلك تلقائياً إلى صفحة إضافة جدول.
- أدخل اسم الجدول، مثلاً: students
- عدد الأعمدة (مثلاً 3 أعمدة) → ثم اضغط .create (مثلاً 3 أعمدة) → ثم اضغط .create

# إنشاء جداول في phpMyAdmin

خصائص إضافية	الطول/القيمة	النوع	الاسم
Primary Key - Auto Increment -		INT	id
	100	VARCHAR	name
		DATE	birth_date

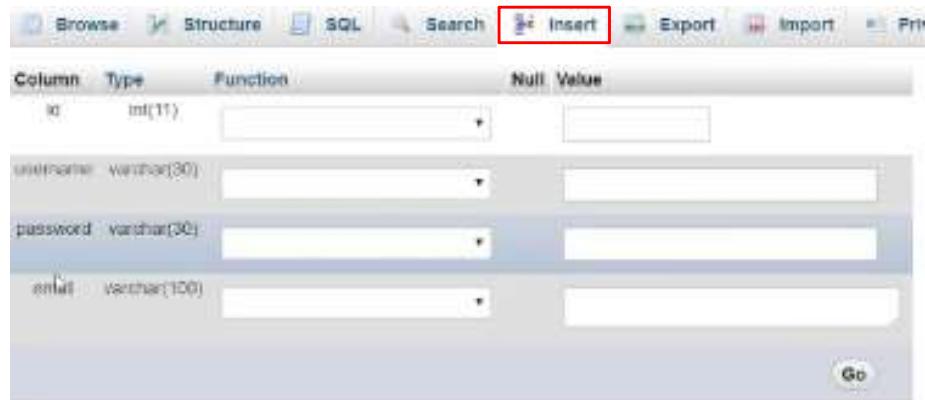
## ثالثاً: تعبئة تفاصيل الأعمدة

- في عمود `:id`
- فّعل  ليكون رقم **A\_I (Auto Increment)** متسلسل تلقائي.
- اختر **PRIMARY** كمفتاح أساسي.
- ثم اضغط على **Save** في الأسفل.

❖ الآن أصبح لدينا جدول `students` يحتوي على أعمدة منظمة ويمكننا البدء في إدخال بيانات .

# إنشاء جداول في phpMyAdmin

## رابعاً: إدخال بيانات داخل الجدول



- من القائمة على اليسار اختر القاعدة my\_school، ثم اضغط على الجدول students.
- اختر التبويب Insert.
- أدخل مثلاً:

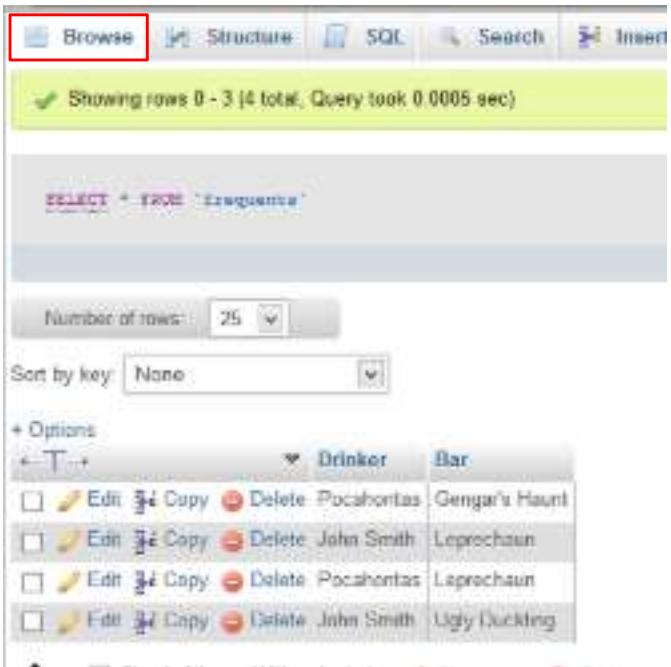
  - name: Ali
  - birth\_date: 10-05-2002

- اضغط Go.

❖ تمت إضافة الطالب "Ali" إلى الجدول بنجاح.

# إنشاء جداول في phpMyAdmin

## خامساً: عرض البيانات



The screenshot shows the phpMyAdmin interface with the 'Browse' tab selected. The top menu bar includes 'Browse', 'Structure', 'SQL', 'Search', and 'Insert'. A green status bar at the top indicates 'Showing rows 0 - 3 (4 total; Query took 0.0005 sec)'. Below the status bar is a SQL query editor containing the command 'SELECT \* FROM `Frequent`'. The main area displays a table with four rows of data. The table has columns labeled 'Drinker' and 'Bar'. The data is as follows:

Drinker	Bar
Pocahontas	Gengar's Haunt
John Smith	Leprechaun
Pocahontas	Leprechaun
John Smith	Ugly Duckling

Below the table, there are buttons for 'Edit', 'Copy', 'Delete', and 'Structure' for each row.

- من نفس الصفحة اضغط على تبويب **Browse**.
- سترى الصف الذي أضفته مع رقم تلقائي في عمود **.id**.

# التصدير (Export)

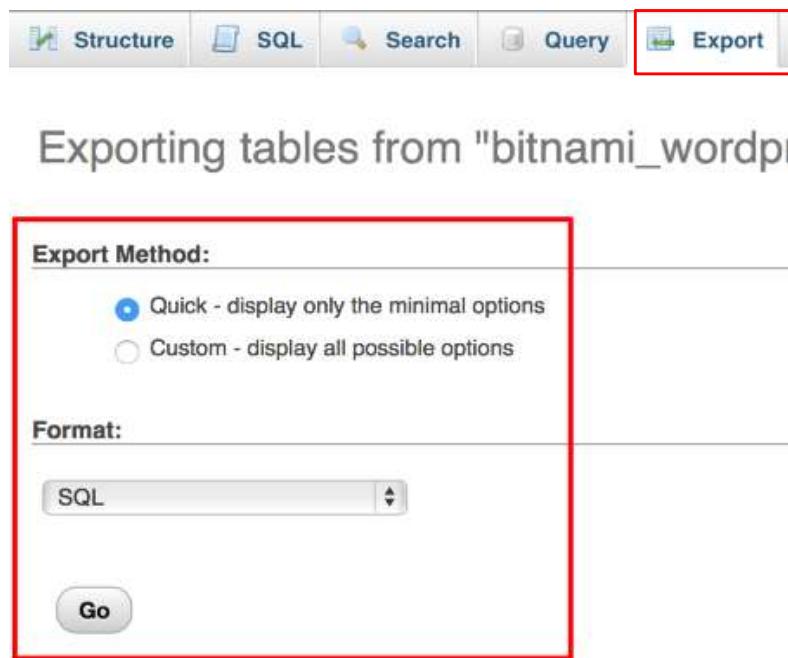
متى تستخدمه؟



- لحفظ نسخة احتياطية من قاعدة بيانات.
- لنقل القاعدة إلى جهاز آخر.
- لمشاركة القاعدة مع زميل.

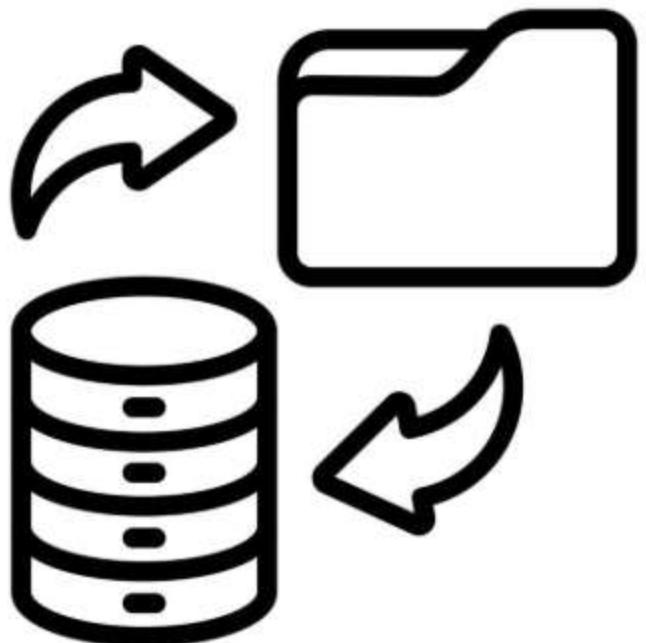
# التصدير (Export)

الخطوات:



- من صفحة phpMyAdmin، اختر قاعدة البيانات من القائمة الجانبية (مثلاً my\_school).
- اضغط على تبويب Export في الأعلى.
- اختر أحد الوضعين:
  - الإعدادات التقائية (سرع): Quick ✓
  - مخصص): لتحديد جداول معينة، أو تعديل الصيغة. Custom
- اختر الصيغة SQL واضغط على زر Go.
- سيتم تحميل ملف sql إلى جهازك.
- الملف يحتوي على كل أوامر إنشاء الجداول والبيانات داخلها. ✨

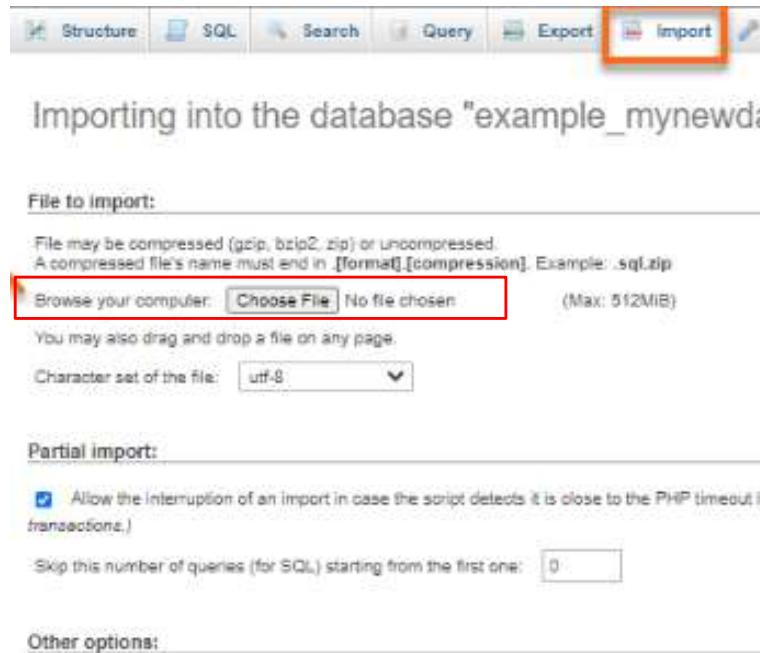
## الاستيراد (Import)



متى تستخدمه؟

- لاسترجاع قاعدة بيانات من نسخة محفوظة.
- لتحميل قواعد جاهزة مثل Northwind.

# الاستيراد (Import)



- الخطوات:**
- من صفحة **phpMyAdmin**، اضغط على تبويب **Databases**، اضغط على تبويب **Import**.
  - أنشئ قاعدة بيانات جديدة بالاسم الذي تريده (مثلاً **northwind**).
  - بعد الإنشاء، اضغط على القاعدة من القائمة الجانبية.
  - اختر تبويب **Import** من الأعلى.
  - اضغط على **Choose File** واختر ملف **.sql** من جهازك (مثل **northwind.sql**).
  - اترك باقي الإعدادات كما هي، واضغط على **Go**.
  - ★ بعد ثوانٍ قليلة، سيتم استيراد جميع الجداول والبيانات تلقائياً إلى القاعدة.

# استخدام محرر SQL في phpMyAdmin

ما هو محرر SQL؟

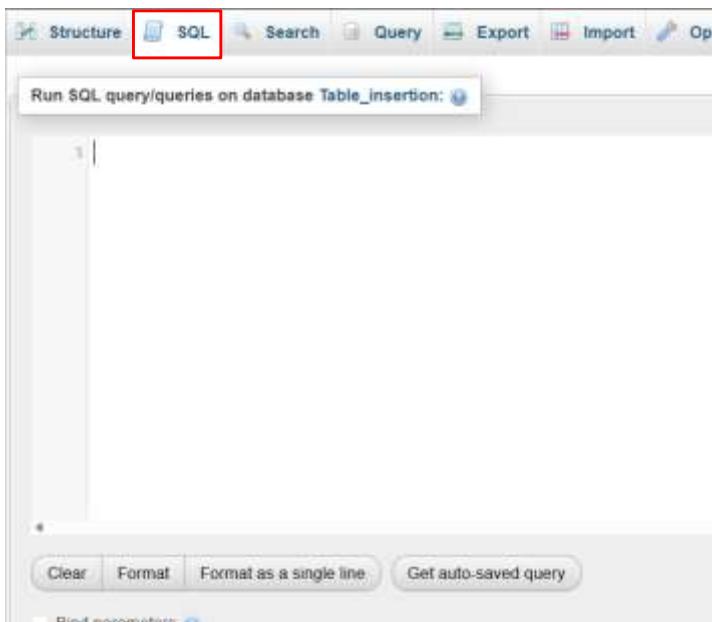
محرر SQL هو أداة مدمجة داخل phpMyAdmin تتيح لك كتابة وتشغيل أوامر SQL يدوياً.

تستخدمه عند الحاجة إلى:

- عرض بيانات (SELECT)
- إدخال بيانات (INSERT)
- تعديل بيانات (UPDATE)
- حذف بيانات (DELETE)
- إنشاء أو حذف جداول (CREATE, DROP)
- تشغيل استعلامات مخصصة مثل JOIN, WHERE, ORDER BY

# استخدام محرر SQL في phpMyAdmin

أين تجد محرر SQL؟



• افتح <http://localhost/phpmyadmin>

• اختر قاعدة البيانات التي تريده العمل عليها من القائمة الجانبية (مثلاً: northwind)

• في الأعلى اضغط على تبويب SQL

سيظهر مربع كبير يمكنك الكتابة فيه، وزر يسمى Go أو تنفيذ.

كيف تستخدمه؟

• اكتب الاستعلام الذي تريده في المربع.

• اضغط على زر Go.

• النتائج ستظهر في الأسفل.

# مقدمة عن MySQL في SELECT

ما هو **SELECT** ؟

أمر **SELECT** يُستخدم لاستخراج البيانات من جدول أو أكثر في قاعدة البيانات.

يمكنك أن تحدد:



- الأعمدة التي تريد عرضها
- من أي جدول
- تحت شروط معينة ( باستخدام WHERE )
- وكيفية ترتيب النتائج ( ORDER BY )
- وعدد النتائج ( LIMIT )

# الشكل الأساسي لأمر SELECT

```
SELECT column1, column2  
FROM table_name;
```

□: تعني "اختر" SELECT

□: أسماء الأعمدة التي تريد عرضها 2nmuloc , 1column

□: اسم الجدول الذي سستخرج منه البيانات table\_name

## الشكل الأساسي لأمر SELECT

استخدام \* لاستعراض كل الأعمدة

☞ النتيجة: يعرض كل الأعمدة الموجودة في جدول المنتجات.

```
SELECT * FROM Products;
```

## الشكل الأساسي لأمر SELECT

□ تغيير اسم العمود في النتيجة : (Alias)

❖ النتيجة: يعرض الأعمدة بأسماء مخصصة في النتيجة.

```
SELECT ProductName AS 'المتّج', UnitPrice AS 'السعر'  
FROM Products;
```

## الشكل الأساسي لأمر SELECT

□ العمليات الحسابية داخل SELECT

❖ النتيجة: يعرض اسم المنتج، السعر الأصلي، والسعر بعد إضافة 20% ضريبة (كمثال).

```
SELECT ProductName, UnitPrice, UnitPrice * 1.2 AS 'PriceWithTax'  
FROM Products;
```

# الشكل الأساسي لأمر SELECT

□ استعلام مع شرط:

➤ النتيجة: يعرض فقط العملاء من البرازيل.

```
SELECT * FROM Customers  
WHERE Country = 'Brazil';
```

## الشكل الأساسي لأمر SELECT

□ استعلام مع ترتيب (ORDER BY):

➤ النتيجة: يعرض المنتجات مرتبة من الأعلى إلى الأدنى.

```
SELECT ProductName, UnitPrice  
FROM Products  
ORDER BY UnitPrice DESC;
```

## الشكل الأساسي لأمر SELECT

- استعلام مع تحديد عدد النتائج (LIMIT):
- النتيجة: يعرض أول 3 موظفين فقط.

```
SELECT * FROM Employees  
LIMIT 3;
```

# الشكل الأساسي لأمر SELECT

□ متى تستخدم ?SELECT

الأمر المناسب

السيناريو

SELECT \* FROM table\_name;

عرض كل البيانات من جدول معين

SELECT ... WHERE ...;

عرض بيانات مع شرط معين

SELECT ... JOIN ...;

عرض بيانات من جداول متعددة

ORDER BY column ASC/DESC

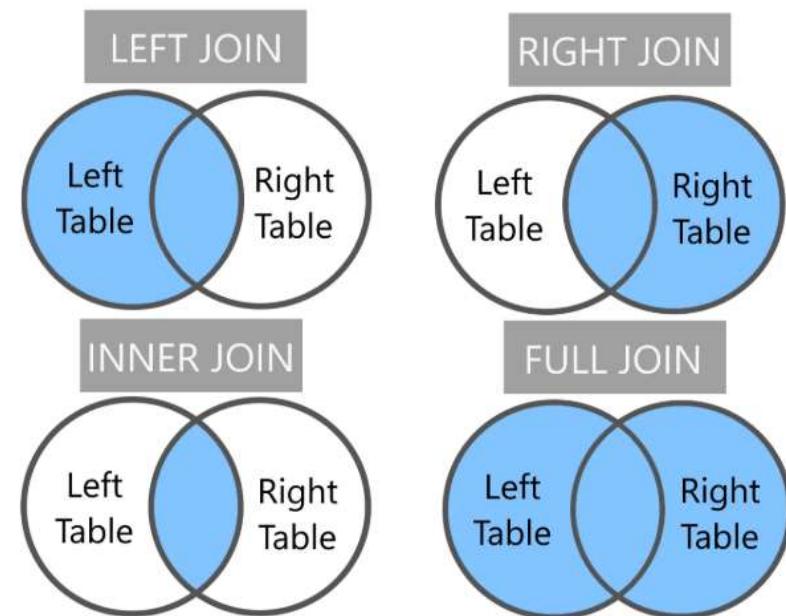
ترتيب النتائج

LIMIT number

تحديد عدد النتائج المعروضة

# استخدام MySQL في SELECT مع JOIN

ما هو JOIN؟

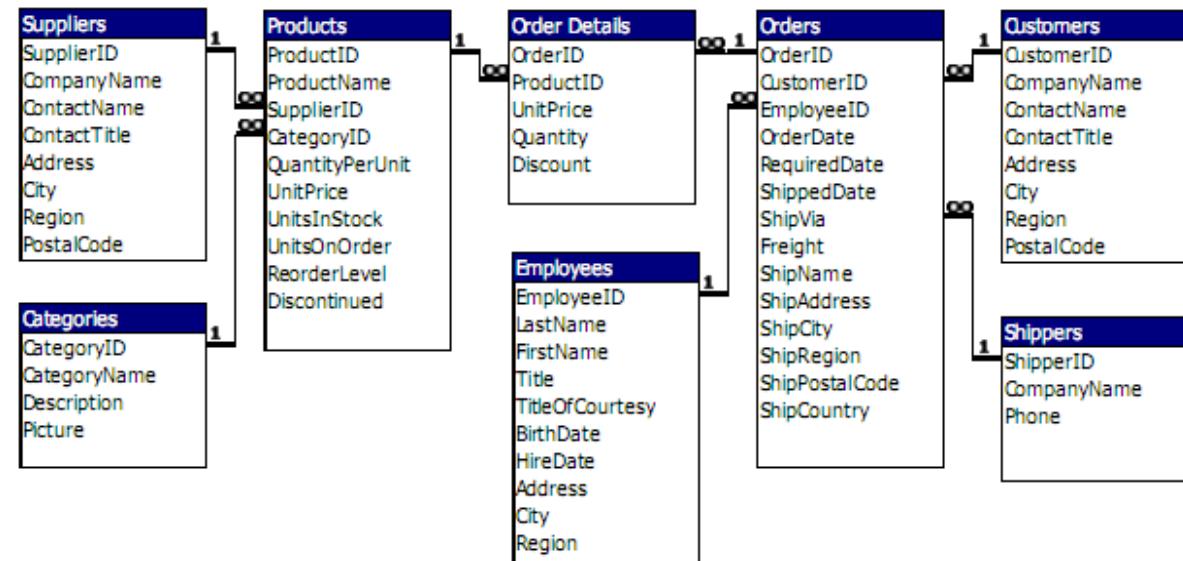


كلمة JOIN في SQL تُستخدم لربط جداول متعددة بناءً على علاقـة منطقـية بين الأعمـدة المشـترـكة، غالـباً عبر مفتـاح أسـاسـي في جـدول و مفتـاح أجـنبـي في جـدول آخر.

الشكل العام لاستخدام JOIN:

```
SELECT table1.column, table2.column  
FROM table1  
JOIN table2 ON table1.key = table2.key;
```

# استخدام MySQL في JOIN مع SELECT



- في قاعدة **Northwind** لدينا علاقات كثيرة بين الجداول، مثل:
- العميل له عدة طلبات. **Orders ↔ Customers**
  - كل طلب يحتوي على تفاصيل منتجات. **OrderDetails ↔ Orders**
  - كل منتج يمكن أن يوجد في عدة طلبات. **OrderDetails ↔ Products**

# أنواع JOIN في MySQL

ماذا يفعل؟

النوع

يعرض فقط الصفوف التي لها تطابق في كلا الجداولين.

INNER JOIN

يعرض كل صفوف الجدول الأيسر، ويُكمل من الجدول الأيمن إذا وجد تطابق.

LEFT JOIN

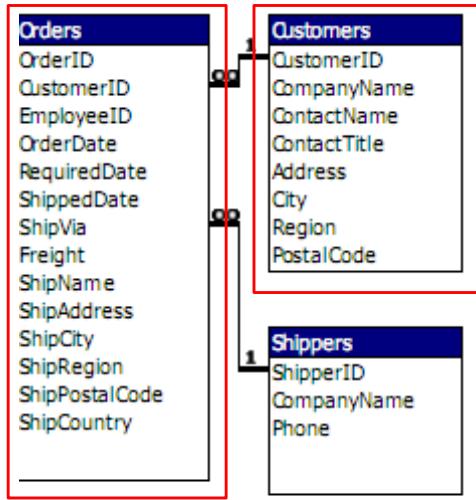
يعرض كل صفوف الجدول الأيمن، ويُكمل من الجدول الأيسر إذا وجد تطابق.

RIGHT JOIN

لماذا JOIN مهم جداً؟

- لأنه يتيح لك العمل مع بيانات مترابطة من جداول مختلفة.
- من المستحبيل عمل أنظمة حقيقة (مثل متجر أو نظام تسجيل) بدون ربط الجداول.

# مثال عملي



مثال عملي: ربط جدول Orders مع Customers

الهدف:

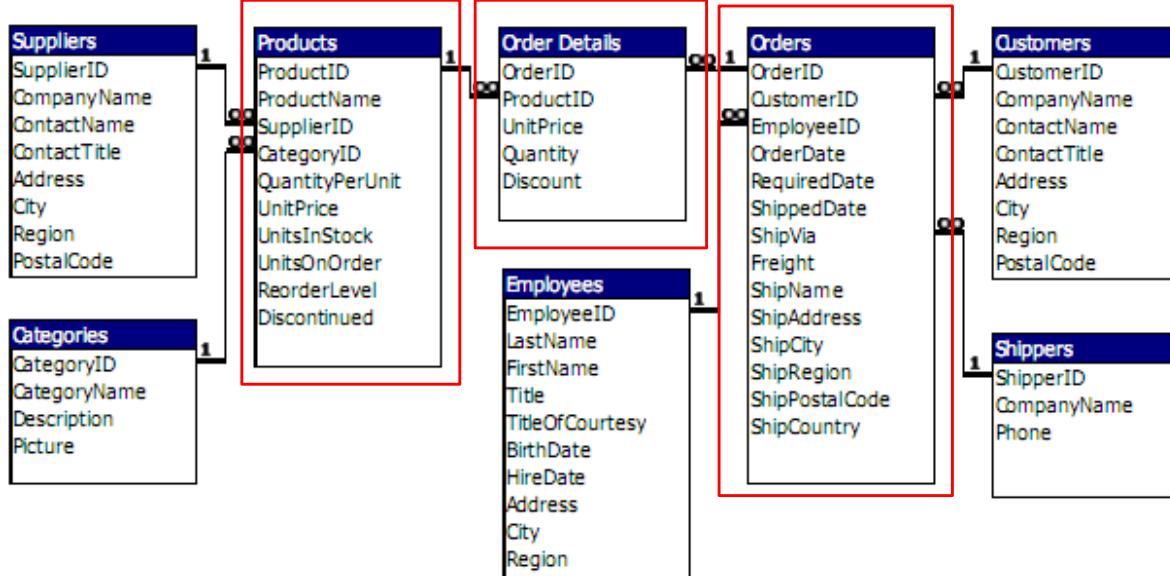
عرض رقم الطلب، اسم العميل، وتاريخ الطلب.

الشرح:

- نربط جدول Orders مع Customers باستخدام العمود المشترك CustomerID.
- نعرض رقم الطلب، اسم الشركة، وتاريخ الطلب.

```
SELECT Orders.OrderID, Customers.CompanyName, Orders.OrderDate
FROM Orders
JOIN Customers ON Orders.CustomerID = Customers.CustomerID;
```

# مثال عملي



مثال ثاني: ربط 3 جداول معًا  
الهدف:

عرض اسم المنتج، رقم الطلب، وكمية المنتج في الطلب.

الشرح:

- بدأنا من جدول **OrderDetails** لأنه جدول العلاقة.
- ربطنا بـ **Products** لجلب اسم المنتج.
- ربطنا بـ **Orders** لجلب رقم الطلب.

```
SELECT Products.ProductName, Orders.OrderID, OrderDetails.Quantity
FROM OrderDetails
JOIN Products ON OrderDetails.ProductID = Products.ProductID
JOIN Orders ON OrderDetails.OrderID = Orders.OrderID;
```

## مثال عملي

مثال باستخدام :LEFT JOIN

الهدف:

عرض كل العملاء حتى لو لم يقدموا أي طلب.

❖ النتيجة: العملاء الذين لا يملكون طلبات ستظهر لديهم NULL كـ OrderID .

```
SELECT Customers.CompanyName, Orders.OrderID
FROM Customers
LEFT JOIN Orders ON Customers.CustomerID = Orders.CustomerID;
```

• مؤلف الكتاب: المهندس حسام يونسو

• مراجع: schools3w

• روابط قد تفيد الطالب:

<https://www.wschools.com>

شكراً لكم