قواعد البيانات

تاريخ التحديث: يونيو 9, 2024

لغة SQL الدليل الشامل الأول

يأخذك هذا الدليل في رحلة تعريفية إلى عالم لغة SQL، المستخدمة لإدارة وتنظيم البيانات في قواعد البيانات. هذا الشرح مناسب للمبتدئين ويشرح بلغة سهلة ومبسطة كيف يمكن استخدام SQL لإنشاء واستعلام وتحديث البيانات داخل قواعد البيانات.

اسم الدليل: لغة SQL

رقم الدليل: 300

المجال: قواعد البيانات SQL

المتطلبات: لا يوجد

مسار تعلم متعلق بهذا الدليل: -

قائمة فيديوهات الدليل

تنزيل الدليل كملف الكتروني PDF

تاريخ تحديث الدليل: يونيو 23, 2024

ما هي SQL؟

SQL أو **Structured Query Language** هي لغة معيارية تُستخدم لإدارة قواعد البيانات العلائقية. تُمكِّنك من إجراء عمليات متعددة مثل الاستعلام عن البيانات، تحديثها، إدخالها، وحذفها من الجداول.

شاهد مقدمة حول هذا الدليل كفيديو تعليمي: رابط الفيديو

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

ماقبل أساسيات لغة SQL

هذه الأمور عليك أن تعرفها قبل البدء بتعلم لغة SQL

ما هي قواعد البيانات؟

قاعدة البيانات هي مجموعة منظمة من البيانات يمكن الوصول إليها وإدارتها وتحديثها بسهولة. تُستخدم قواعد البيانات لتخزين المعلومات بطريقة تجعلها سهلة البحث والفرز والاسترجاع.

مثال بسيط

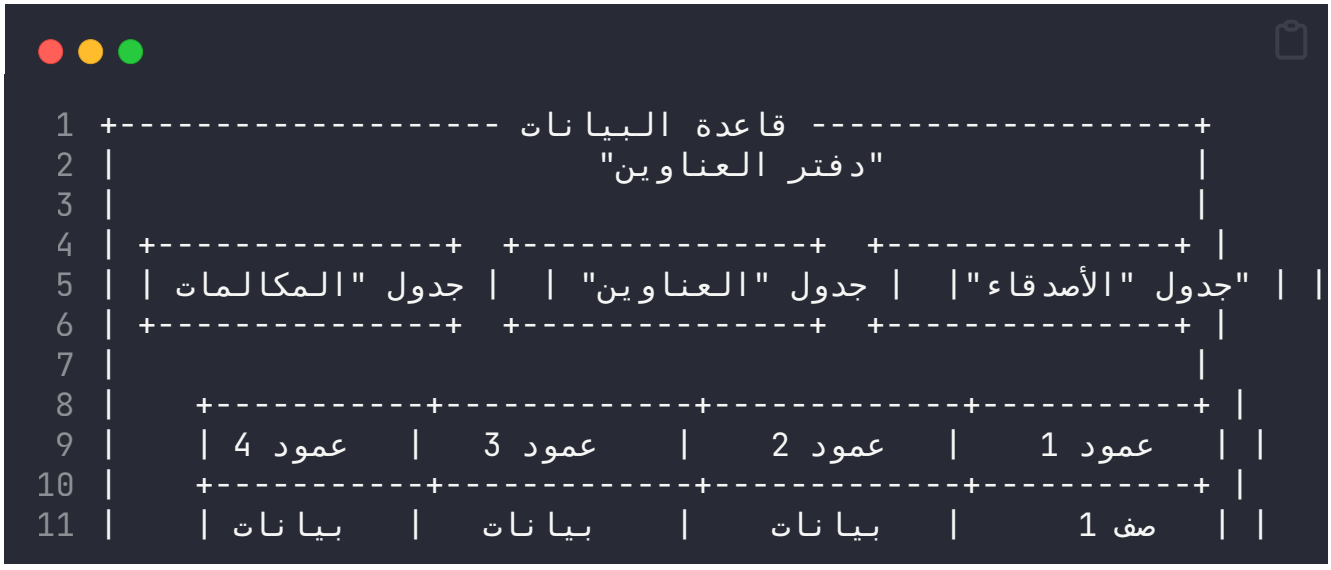
الاسم الأول	الاسم الأخير	رقم الهاتف	البريد الإلكتروني
علي	الأحمد	123456789	ali@example.com
سارة	أحمد	987654321	sara@example.com
محمد	الطيب	456123789	mohamed@example.com

جدول بيانات الأصدقاء

- **اسم الجدول**: "الأصدقاء".
- **الصفوف**: كل صف يمثل سجلاً (Record) أو مدخل بيانات. في المثال، الصف الأول يحتوي على معلومات صديق يدعى "علي الأحمد".
- **الأعمدة**: كل عمود يمثل نوعًا معينًا من البيانات. في المثال، الأعمدة هي "الاسم الأول"، "الاسم الأخير"، "رقم الهاتف"، و"البريد الإلكتروني".

باختصار، قاعدة البيانات هي مثل نظام تصنيف للمعلومات حيث يتم تخزين المعلومات في جداول (مثل ملفات على الرف)، وكل جدول يتكون من أعمدة (مثل التصنيفات)، وصفوف (مثل الأوراق داخل الملفات) لتخزين البيانات المنظمة.

توضيح بصري بسيط:



Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

```
| | صف 2 | بيانات | بيانات | بيانات | بيانات |
+-----------+-----------+-----------+-----------+-----------+
```

Plaintext

ما هي الأعمدة؟

الأعمدة هي مكونات الجدول التي تمثل خصائص معينة أو نوعًا معينًا من البيانات. كل عمود في الجدول له اسم ويخزن نوعًا معينًا من المعلومات.

مثال بسيط

في جدول "الأصدقاء" أعلاه:

- **العمود "الاسم الأول"**: يحتوي على الأسماء الأولى للأصدقاء.
- **العمود "الاسم الأخير"**: يحتوي على الأسماء الأخيرة للأصدقاء.
- **العمود "رقم الهاتف"**: يحتوي على أرقام هواتف الأصدقاء.
- **العمود "البريد الإلكتروني"**: يحتوي على عناوين البريد الإلكتروني للأصدقاء.

ما هي الصفوف؟

الصفوف هي مكونات الجدول التي تحتوي على البيانات الفعلية. كل صف يمثل سجلًا (Record) واحدًا ويخزن مجموعة من القيم في كل عمود من أعمدة الجدول.

مثال بسيط

في جدول "الأصدقاء" أعلاه، كل صف يحتوي على معلومات عن صديق واحد. الصف الأول يحتوي على بيانات "علي الأحمد"، بما في ذلك الاسم الأول، الاسم الأخير، رقم الهاتف، والبريد الإلكتروني.

الملخص

تخيل قاعدة بيانات كحافظة ملفات كبيرة:

- **قاعدة البيانات**: الحافظة.
- **الجدول**: كل مجلد في الحافظة يمثل جدولًا. مثلا، مجلد يحتوي على معلومات الأصدقاء.
- **الأعمدة**: الفئات في كل مجلد مثل "الاسم الأول"، "الاسم الأخير"، إلخ.
- **الصفوف**: كل ورقة داخل المجلد تحتوي على المعلومات عن شخص واحد.

الحاجة للغة استعلام

قواعد البيانات الحديثة غالبًا ما تكون معقدة وتتضمن العديد من الجداول والعلاقات بينها. هذا يعني أنه من الضروري أن يكون هناك واجهة قوية وفعالة لإدارة وتحليل هذه البيانات بكفاءة، وهنا تأتي دور لغة SQL.

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

لغة SQL تقدم مجموعة من الأدوات والأوامر التي تسمح للمستخدمين بالتعامل مع قواعد البيانات بسهولة وفعالية. من خلال SQL، يمكن للمستخدمين القيام بالعديد من الأنشطة مثل:

1. استعلام البيانات: باستخدام عبارات SELECT يمكن للمستخدمين استعلام البيانات من مجموعة متنوعة من الجداول باستخدام مجموعة متنوعة من الشروط والمعايير.
2. تحديث وحذف البيانات: يمكن للمستخدمين استخدام عبارات UPDATE و DELETE لتحديث البيانات الموجودة بالفعل في الجداول أو حذفها.
3. إدخال البيانات الجديدة: يمكن للمستخدمين استخدام عبارة INSERT لإدخال بيانات جديدة إلى الجداول.
4. إدارة هيكل قاعدة البيانات: يمكن للمستخدمين إنشاء وتعديل وحذف الجداول والفهارس والقيود باستخدام عبارات DDL (Data Definition Language).

بفضل هذه الأدوات والأوامر، يمكن للمستخدمين إدارة وتحليل البيانات في قواعد البيانات بكفاءة، بغض النظر عن حجمها أو تعقيدها. وبذلك، تكون لغة SQL أداة قوية وحيوية في مجال إدارة البيانات الحديثة.

كيف تتبع هذا الدليل؟

سيتم توضيح كل التعليمات من خلال تطبيق عملي.

> سنقوم بتنفيذ كود وتعليمات SQL على موقع:
> https://www.db-fiddle.com

1. اذهب لرابط الموقع: https://www.db-fiddle.com
2. سيكون هنالك مربعان: الأول خاص بإضافة جداول قاعدة البيانات تحت اسم schema SQL
3. المربع الثاني: يتم اضافة فيه الاستعلامات تحت اسم Query SQL
4. اكتب أوامر التمرينات في المربع الذي يحدده التمرين، ثم قم بالنقر على تنفيذ RUN موجودة في الأعلى.
5. سأقوم بمشاركة رابط الكود الذي أكتبه في كل مرة معكم، وسيتم اضافة الكود لنفس الموقع السابق.
6. يمكنك مقارنة الكود الذي تكتبه مع الكود الموجود في التمرين لتتأكد من صحة الكود الذي كتبته أو تتبع الأخطاء
7. لا تيأس إذا وجدت صعوبة بفهم فكرة أو تمرين ما، فالعقل يحتاج وقت للاعتياد على المعلومات الجديدة.
8. استمر! حاول! استمر!

ماذا تعني Schema SQL؟

هنا سيتم اضافة قاعدة البيانات نفسها أي بمعنى آخر البيانات التي لدينا مجمعة بجداول ومعبر عنها بصفوف وجداول.

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

هذا هو من أهم مزايا SQL، أنها تسمح لنا بتنظيم البيانات بجداول التي بدورها تحتوي على أعمدة وصفوف.

ماذا تعني SQL Query ؟

بعد إنشاء وتنظيم البيانات بقاعدة البيانات، سيكون هنالك حاجة لطلب هذه البيانات.

اذا كانت هذه بيانات الأصدقاء كما في المثال في القسم الأول، فالآن نريد أن نحصل على رقم هاتف سارة، ففي هذه الحالة يتم القيام بعملية استعلام أو استرجاع أو ما تسمى باللغة الانكليزية Query.

فعندما نريد استرجاع أو الاستعلام عن معلومة ما سواء كان هذا الاستعلام معقد أو بسيط، فهذه العملية تسمى Query

ما هذا الموقع db-fiddle.com

للأسف، عملية تثبيت أدوات قواعد البيانات على جهاز الحاسوب قد تكون عملية صعبة للمبتدئين.

ولذلك، وجدت عن موقع يساعدنا على التدرب على أوامر لغة SQL دون الحاجة فعلياً لتثبيت أي أداة على جهاز الحاسوب.

هذا سيسهل عملية التعلم ولاحقاً يمكن أن تتعلم كيفية تثبيت لغة SQL على جهازك.

لا، هذا ليس إعلان أو مقالة دعائية لموقع db-fiddle.com. وجدت أنه موقع مفيد ومجاني، وأحببت مشاركة الفائدة معكم.

أساسيات لغة SQL

هنا سيتم توضيح أهم الأوامر التي عليك تعلمها في لغة SQL.

سيتم توضيح الأوامر الأساسية بالتدريبات العملية

تدريب 301: إنشاء جدول وإدخال بيانات والاستعلام عنها.

شاهد تطبيق التمرين كفيديو تعليمي: لغة SQL الدليل الشامل الأول - التمرين 301

تفاصيل التمرين

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

1. المطلوب: قم بإنشاء جدول في قاعدة بيانات يحتوي معلومات الأصدقاء في الجدول أدناه.
2. رابط الأمر النهائي - الخطوة الأولى: تمرين 301 - الكود النهائي 1
3. رابط الأمر النهائي - الخطوة الثانية: تمرين 301 - الكود النهائي 2
4. رابط الأمر النهائي - الخطوة الثالثة: تمرين 301 - الكود النهائي 3

الاسم الأول	الاسم الأخير	رقم الهاتف	البريد الإلكتروني
علي	الأحمد	123456789	ali@example.com
سارة	أحمد	987654321	sara@example.com
محمد	الطيب	456123789	mohamed@example.com

نعتبر أن لدينا قاعدة بيانات منشئة وجاهزة، ونبدأ من نقطة إنشاء جداول.

الخطوة الأولى

الحل مع التوضيح

نكتب الكود الآتي في موقع db-fiddle.com في قسم Schema

```sql
CREATE TABLE Friends (
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    PhoneNumber VARCHAR(15),
    Email VARCHAR(100)
);

```

هذا الكود يقوم بإنشاء جدول في قاعدة البيانات يُسمى "Friends"، وهذا الجدول يحتوي على أربعة أعمدة:

1. FirstName (الاسم الأول): هذه العمود يحتوي على الأسماء الأولى لأصدقائك.
2. LastName (الاسم الأخير): هذه العمود يحتوي على الأسماء الأخيرة لأصدقائك.
3. PhoneNumber (رقم الهاتف): هذه العمود يحتوي على أرقام هواتف أصدقائك.
4. Email (البريد الإلكتروني): هذه العمود يحتوي على عناوين البريد الإلكتروني لأصدقائك.

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

كل عمود له نوع محدد للبيانات التي يمكن تخزينها فيه. في هذا الجدول، يتم استخدام أنواع بيانات "VARCHAR"، وهي تعني سلسلة محارف بعدد معين. على سبيل المثال، "FirstName" و "LastName" يمكنهما أن يحتويا على نص يصل إلى 50 محرفاً، و "PhoneNumber" يمكنه أن يحتوي على نص يصل إلى 15 محرفاً، و "Email" يمكنه أن يحتوي على نص يصل إلى 100 محرفاً.

ملاحظات في دفترك

```SQL
يتم إنشاء الجدول من خلال الأمر
Create Table
```

varchar يعبر أن قيمة العمود قد تكون بين 0 لقيمة قصوى يتم تحديدها. مثلاً:
varchar(50) يعني أن قيمة العمود لا يمكن أن تكون أكثر من 50 محرف (حرف أو رقم أو رمز خاص)

سؤال: هل يجب أن أكتب كل كلمات الأمر بأحرف كبيرة (upper case)؟

لا، في SQL، ليس من الضروري كتابة الأوامر أو الكلمات الرئيسية بأحرف كبيرة. يعمل الكود بشكل متماثل سواء كتبت الكلمات بأحرف كبيرة أو صغيرة. عادةً ما يكتب الأشخاص بعض الكلمات بأحرف كبيرة لتسهيل قراءة الكود وتمييز الكلمات الرئيسية من الأسماء العادية في الجداول والأعمدة. ومع ذلك، يمكن استخدام أحرف صغيرة أيضًا دون مشاكل في تنفيذ الأوامر في SQL.

الخطوة الثانية

الحل مع التوضيح

نكتب الكود الآتي في موقع db-fiddle.com في قسم Schema

```SQL
CREATE TABLE Friends (
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    PhoneNumber VARCHAR(15),
    Email VARCHAR(100)
);

INSERT INTO Friends (FirstName, LastName, PhoneNumber, Email)
```

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

```sql
VALUES
    ('Ali', 'Al-Ahmad', '123456789', 'ali@example.com'),
    ('Sara', 'Ahmad', '987654321', 'sara@example.com'),
    ('Mohamed', 'Al-Tayeb', '456123789', 'mohamed@example.com');


```

SQL

هذا الكود هو أمر INSERT في SQL، ويتم استخدامه لإضافة سجلات جديدة (صفوف) إلى جدول. دعني أقدم لك تفسيرًا بسيطًا للكود:

1. **INSERT INTO Friends**: هذا الجزء يحدد اسم الجدول الذي نريد إدراج السجلات فيه، وهو "Friends". سنقوم بإضافة السجلات إلى جدول "Friends".
2. **(FirstName, LastName, PhoneNumber, Email)**: هذا الجزء يحدد الأعمدة التي نريد إدراج البيانات فيها. تُسمى الأعمدة FirstName، LastName، PhoneNumber وEmail.
3. **VALUES**: هذه الكلمة الرئيسية تُستخدم لتحديد القيم التي نريد إدراجها في الأعمدة المحددة.
4. **('علي', 'الأحمد', '123456789', 'ali@example.com')**: هذه هي مجموعة القيم الأولى التي سنقوم بإدراجها في الجدول. كل مجموعة من القيم تتوافق مع صف واحد في الجدول. القيم تُقدم في نفس ترتيب الأعمدة المحددة سابقًا. لذا، 'علي' تتوافق مع FirstName، 'الأحمد' تتوافق مع LastName، '123456789' تتوافق مع PhoneNumber، و'ali@example.com' تتوافق مع Email.
5. **('سارة', 'أحمد', '987654321', 'sara@example.com')**: هذه هي مجموعة القيم الثانية التي سنقوم بإدراجها. مرة أخرى، كل قيمة تتوافق مع العمود المقابل لها بنفس الترتيب.
6. **('محمد', 'الطيب', '456123789', 'mohamed@example.com')**: هذه هي مجموعة القيم الثالثة التي سنقوم بإدراجها، وتتبع نفس النمط.

لذا، يقوم هذا الكود SQL بإدراج ثلاثة سجلات جديدة في جدول "Friends"، حيث يحتوي كل سجل على اسم،

ملاحظات في دفترك

```sql
اضافة سجلات للجدول
INSERT INTO اسم الجدول
```

SQL

يجب تحديد اسماء الاعمدة التي تريد اضافة قيم لها، ووضعها ضمن أقواس عادية، وتفصل بينها بفاصلة.

سؤال: هل استخدم اشارات اقتباس أحادية ' أو ازدواجية "؟

استخدم اشارات الاقتباس الأحادية للقيم النصية لتفادي أي التباس. ستتعلم أكثر عن هذا في المستقبل.

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

لمزيد من المعلومات:
شاهد فيديو: لن تخطأ باستخدام ‘ , ” في SQL بعد الآن.

الخطوة الثالثة

الحل مع التوضيح

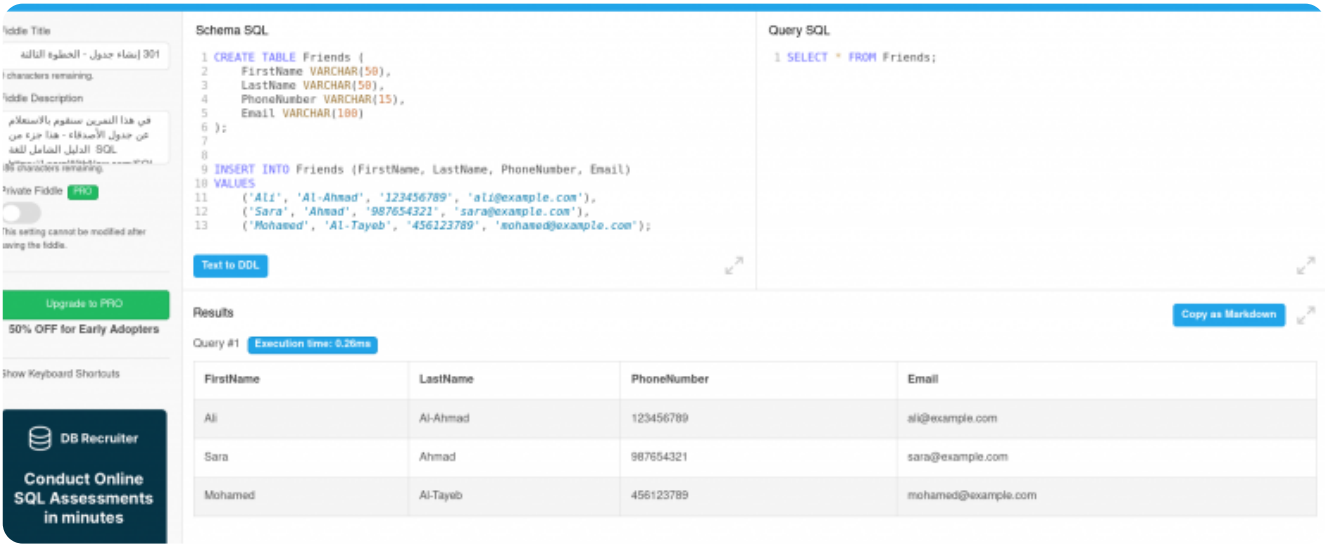
نكتب الكود الآتي في موقع db-fiddle.com في قسم Query

```sql
SELECT * FROM Friends;

```

يقوم هذا الأمر بالاستعلام لاسترجاع جميع البيانات المخزنة في جدول معين. دعنا نوضحه بشكل مبسط:

- **SELECT**: تعني “اختر”. تُستخدم لتحديد الأعمدة التي تريد استرجاعها من قاعدة البيانات.
- *: تعني “كل الأعمدة”. عندما نستخدم *، فنحن نطلب استرجاع جميع الأعمدة الموجودة في الجدول.
- **FROM**: تعني “من”. تُستخدم لتحديد الجدول الذي نريد استرجاع البيانات منه.
- **Friends**: اسم الجدول الذي نريد استرجاع البيانات منه. في هذا المثال، الجدول اسمه “Friends”.



نتيجة الاستعلام عن الأصدقاء

ملاحظات في دفترك

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

```sql
الاستعلام عن السجلات
SELECT
```

سؤال: ماذا تعني *؟

في SQL، الرمز * يُعرف باسم "علامة النجمة" أو "النجمة"، وهو يستخدم لاسترجاع جميع الأعمدة من الجدول المحدد.

تدريب 302: إنشاء جدول الأفلام والاستعلام عنه

تفاصيل التمرين

1. المطلوب: الاستعلام عن الأفلام من جدول الأفلام.
2. يمكنك محاولة إنشاء الجدول الآتي وإدخال البيانات فيه.
3. أو يمكنك الاكتفاء بقراءة أمر إنشاء الجدول وإدخال بيانات فيه.
4. والمواصلة لتمرين الاستعلام والتصفية.
5. رابط الأمر النهائي - الخطوة الأولى: تمرين 302 - الكود النهائي 1
6. رابط الأمر النهائي - الخطوة الثانية: تمرين 302 - الكود النهائي 2
7. رابط الأمر النهائي - الخطوة الثالثة: تمرين 302 - الكود النهائي 3

المعرّف	عنوان الفيلم	النوع	سنة الإصدار	التقييم
1	Inception	خيال علمي	2010	8.8
2	The Godfather	جريمة	1972	9.2
3	The Dark Knight	أكشن	2008	9.0
4	Pulp Fiction	جريمة	1994	8.9
5	The Shawshank Redemption	دراما	1994	9.3
6	The Matrix	خيال علمي	1999	8.7
7	Forrest Gump	دراما	1994	8.8
8	The Avengers	أكشن	2012	8.0
9	Gladiator	أكشن	2000	8.5
10	Titanic	رومانسية	1997	7.8

جدول الأفلام

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

نعتبر أن لدينا قاعدة بيانات منشئة وجاهزة، ونبدأ من نقطة إنشاء جداول.

الخطوة الأولى

الحل مع التوضيح

نكتب الكود الآتي في موقع db-fiddle.com في قسم Schema

```sql
-- إنشاء جدول باسم Movies
CREATE TABLE Movies (
    ID INT PRIMARY KEY,                -- المعرف الفريد للفيلم
    Title VARCHAR(255),                -- عنوان الفيلم
    Genre VARCHAR(50),                 -- نوع الفيلم
    ReleaseYear INT,                   -- سنة إصدار الفيلم
    Rating DECIMAL(3, 1)               -- نظام النقاط حتى عشري واحد
);
```

1. **CREATE TABLE Movies :)**
   - هذا الجزء يعني أننا نقوم بإنشاء جدول جديد بإسم "Movies".
2. **ID INT PRIMARY KEY:**
   - يعرّف عمودًا بإسم "ID" من نوع INT (أي عدد صحيح)، ويتم تعيينه كمفتاح أساسي PRIMARY KEY للجدول. الفهرس الأساسي يضمن أن قيمة كل سجل في هذا العمود فريدة ولا تتكرر، وهو يُستخدم عادة لتعريف كيفية تمثيل كل سجل بشكل فريد.
3. **Title VARCHAR(255):**
   - يعرّف عمودًا بإسم "Title" من نوع VARCHAR، وهو نوع بيانات يمكنه تخزين سلسلة نصية بحد أقصى يصل إلى 255 حرفًا. يُستخدم هنا لتخزين عناوين الأفلام.
4. **Genre VARCHAR(50):**
   - يعرّف عمودًا بإسم "Genre" من نوع VARCHAR، يمكنه تخزين سلسلة نصية بحد أقصى يصل إلى 50 حرفًا. يُستخدم هذا العمود لتخزين نوع أو تصنيف الفيلم مثل "Action" أو "Comedy".
5. **ReleaseYear INT:**
   - يعرّف عمودًا بإسم "ReleaseYear" من نوع INT (أي عدد صحيح). يُستخدم لتخزين سنة إصدار الفيلم.
6. **Rating DECIMAL(3, 1):**

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

- يعرّف عمودًا بإسم "Rating" من نوع DECIMAL، حيث يمثل التقييم الخاص بالفيلم. الـ DECIMAL(3, 1) يعني أن التقييم يمكن أن يحتوي على حد أقصى 3 أرقام كاملة مع رقم عشري واحد، مما يسمح بتخزين تقييمات مثل 8.5 أو 9.0.

بهذا الشكل، يتم إنشاء جدول "Movies" الذي يحتوي على خمسة أعمدة مختلفة تمثل معلومات مختلفة عن الأفلام مثل العنوان، النوع، سنة الإصدار، والتقييم.

ملاحظات في دفترك

```sql
CREATE TABLE Movies (
    ID INT PRIMARY KEY,
```

في قواعد قواعد البيانات، مفتاح Primary Key (المفتاح الأساسي أو المفتاح الرئيسي) يُستخدم لتحديد بشكل فريد كل سجل في جدول قاعدة البيانات. هذا المفتاح له أهمية كبيرة لأنه يوفر وسيلة فعالة للتعرف على سجلات معينة ومنع اختلاط سجلات مع سجلات أخرى. حيث يكون لكل سجل معرّف خاص مميز **ولا يُمكن أبداً لأي سجلين أن يكون لهما نفس المعرّف.**

```sql
CREATE TABLE Movies (
    ------
    ReleaseYear INT,

```

INT هو نوع بيانات يُستخدم لتخزين الأرقام الصحيحة (بدون أي أرقام عشرية)، وهو جزء من أنواع البيانات التي تُستخدم لتعريف حقول الجدول. تعريف الحقل `ReleaseYear INT` يعني أن هذا الحقل سيحتوي على قيم صحيحة تمثل سنة الإصدار.

```sql
CREATE TABLE Movies (
    ------
    Rating DECIMAL(3, 1)

```

DECIMAL(3, 1) في SQL يعني أنه حقل يمكنه تخزين أرقام عشرية بدقة تصل إلى 3 أرقام كليًا، مع 1 رقم بعد الفاصلة العشرية. على سبيل المثال، يمكن تخزين القيم مثل 123.4 أو

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

-7.8، ولكن لا يمكن تخزين القيم مثل 12.345 لأنها تتجاوز الدقة المحددة للأرقام العشرية.

سؤال: ماذا يعني "- -" في المثال السابق؟

هذه العبارة - - تُستخدم لبدء تعليق في سطر واحد في SQL. ويكون أي شيء بعد - - حتى نهاية السطر تعليقًا ولا يتم تنفيذه كجزء من الاستعلام.
التعليق في البرمجة يعني كتابة ملاحظة أو توضيح حول الأمر، وهذا فقط لغايات التوضيح والتنظيم ولا ينفذ أبداً.

الخطوة الثانية

الحل مع التوضيح

نكتب الكود الآتي في موقع db-fiddle.com في قسم Schema

```sql
-- إنشاء جدول باسم Movies
CREATE TABLE Movies (
    ID INT PRIMARY KEY,                 -- المعرف الفريد للفيلم
    Title VARCHAR(255),                 -- عنوان الفيلم
    Genre VARCHAR(50),                  -- نوع الفيلم
    ReleaseYear INT,                    -- سنة إصدار الفيلم
    Rating DECIMAL(3, 1)                -- (نظام النقاط حتى عشري واحد
);


-- إدراج بيانات عينة في الجدول Movies
INSERT INTO Movies (ID, Title, Genre, ReleaseYear, Rating) VALUE
(1, 'Inception', 'Sci-Fi', 2010, 8.8),
(2, 'The Godfather', 'Crime', 1972, 9.2),
(3, 'The Dark Knight', 'Action', 2008, 9.0),
(4, 'Pulp Fiction', 'Crime', 1994, 8.9),
(5, 'The Shawshank Redemption', 'Drama', 1994, 9.3),
(6, 'The Matrix', 'Sci-Fi', 1999, 8.7),
(7, 'Forrest Gump', 'Drama', 1994, 8.8),
(8, 'The Avengers', 'Action', 2012, 8.0),
(9, 'Gladiator', 'Action', 2000, 8.5),
(10, 'Titanic', 'Romance', 1997, 7.8);

```

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

```
SQL
```

هذا الكود يُقوم بإدراج بيانات عينة داخل جدول يسمى `Movies` في قاعدة البيانات.

الخطوة الثالثة

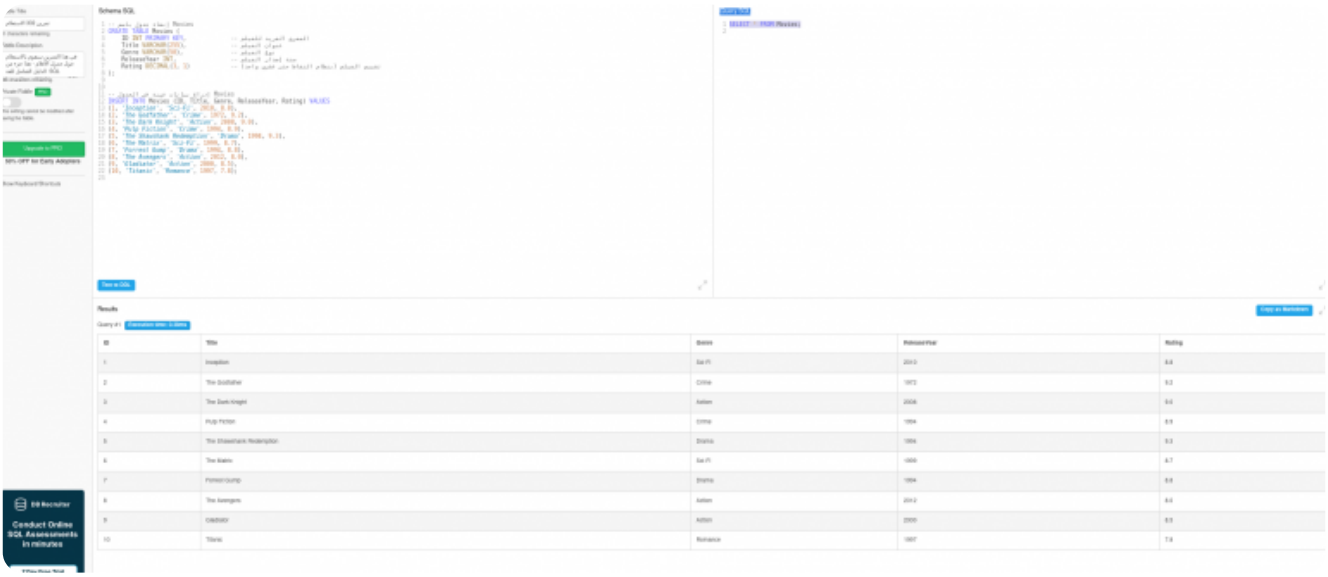
الحل مع التوضيح

نكتب الكود الآتي في موقع db-fiddle.com في قسم Query

```sql
SELECT * FROM Movies;

```

هذا أمر استعلام بسيط، سيعرض لنا كل الأعمدة من جدول الأفلام



نتيجة الاستعلام عن الأفلام

تدريب 303: الاستعلام والتصفية.

تفاصيل التمرين

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

1. يجب أن تقوم بالتدريب السابق 302 قبل الاستمرار بهذا التدريب.
2. سنقوم باستخدام جدول الأفلام للاستعلام عن سجلات معينة.
3. الاستعلام الأول: قم بالاستعلام عن الأفلام التي من نوع Action فقط.
4. الاستعلام الثاني: قم بالاستعلام عن الأفلام التي أصدرت بعد عام 2000.
5. الاستعلام الثالث: الاستعلام عن الأفلام التي قيمت أقل من 8.5.
6. الاستعلام الرابع: الاستعلام عن الأفلام التي صدرت في وبعد عام 2008
7. الاستعلام الخامس: الاستعلام عن الأفلام التي لها تقييم أقل أو يساوي 8.0.
8. الاستعلام السادس: قم بالاستعلام عن الأفلام التي أصدرت بين عامي 1990 و 2000.
9. رابط الأمر النهائي تمرين 303 - الكود النهائي

المعرّف	عنوان الفيلم	النوع	سنة الإصدار	التقييم
1	Inception	خيال علمي	2010	8.8
2	The Godfather	جريمة	1972	9.2
3	The Dark Knight	أكشن	2008	9.0
4	Pulp Fiction	جريمة	1994	8.9
5	The Shawshank Redemption	دراما	1994	9.3
6	The Matrix	خيال علمي	1999	8.7
7	Forrest Gump	دراما	1994	8.8
8	The Avengers	أكشن	2012	8.0
9	Gladiator	أكشن	2000	8.5
10	Titanic	رومانسي	1997	7.8

جدول الأفلام

WHERE =

الاستعلام عن الأفلام من نوع Action

رابط الكود النهائي: الاستعلام عن أفلام Action

نكتب الكود الآتي في موقع db-fiddle.com في قسم Query

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

```sql
-- استعلم عن الأفلام من نوع
-- action
SELECT * FROM Movies WHERE Genre = 'Action';
```

هذا الكود يُستخدم في SQL لاسترجاع جميع البيانات من جدول يسمى `Movies` حيث تكون قيمة العمود `Genre` تساوي `'Action'`. لنشرحه بشكل مبسط:

- **SELECT ***: يعني استرجاع كافة الأعمدة من الجدول.
- **FROM Movies**: يحدد الجدول الذي نريد استخدامه للاستعلام، وفي هذه الحالة هو جدول `Movies`.
- **WHERE Genre = 'Action'**: يعمل كـ"شرط" للبحث، حيث يقوم الاستعلام بتحديد الصفوف التي تحتوي على قيمة `Genre` تساوي `'Action'`.

باختصار، هذا الاستعلام سيُرجع جميع الأفلام التي تنتمي إلى نوع الأكشن (Action) من جدول `Movies`.

ملاحظات في دفترك

```sql
WHERE columnName = 'Value';
```

قم باستعادة السجلات التي يكون فيها قيمة عمود معين يساوي قيمة معينة. الغرض من WHERE هو فلترة وتصفية السجلات بناء على شرط معين.

WHERE >

الاستعلام عن الأفلام التي اصدرت بعد عام 2000

رابط الكود النهائي: الاستعلام عن الأفلام التي أصدرت بعد عام 2000

نكتب الكود الآتي في موقع db-fiddle.com في قسم Query

```sql
SELECT * FROM Movies WHERE ReleaseYear > 2000;
```

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

يقوم الأمر السابق بالتالي:

1. **SELECT * FROM Movies**: يعني استرجاع كافة الأعمدة (* تعني جميع الأعمدة) من جدول يسمى "Movies".
2. **WHERE ReleaseYear > 2000**: يُضيف شرطاً للبحث، يعني استرجاع الصفوف (السجلات) التي تحقق الشرط المحدد، وهو أن قيمة العمود "ReleaseYear" تكون أكبر من 2000.

باختصار، هذا الكود يسترجع جميع الأفلام من جدول "Movies" التي تم إصدارها بعد عام 2000.

ملاحظات في دفترك

```sql
WHERE columnName > 'Value';
```

يقوم بتحديد الصفوف التي تحتوي على قيم في عمود محدد (`columnName`) تكون أكبر من قيمة معينة (`'Value'`).

WHERE <

الاستعلام عن الأفلام التي قيمت أقل من 8.5

رابط الكود النهائي: الاستعلام عن الأفلام التي لها تقييم أقل من 8.5

نكتب الكود الآتي في موقع db-fiddle.com في قسم Query

```sql
SELECT * FROM Movies WHERE Rating < 8.5;
```

يقوم بالتالي:

1. **SELECT * FROM Movies**: يعني استرجاع كافة الأعمدة (* تعني جميع الأعمدة) من جدول يسمى "Movies".
2. **WHERE Rating < 8.5**: يضيف شرطاً للبحث، حيث يتم استرجاع الصفوف (السجلات) التي تحقق الشرط المحدد. في هذه الحالة، يتم استرجاع الأفلام التي تمتلك تقييم (Rating) أقل من 8.5.

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

باختصار، هذا الأمر يعني أننا نريد استرجاع كافة الأفلام من الجدول "Movies" التي لديها تقييم أقل من 8.5.

ملاحظات في دفترك

```sql
WHERE columnName < 'Value';
```

يقوم بتحديد الصفوف في جدول قاعدة البيانات حيث تكون قيمة العمود المحدد (`columnName`) أقل من القيمة المحددة (`'Value'`).

WHERE >=

الاستعلام عن الأفلام التي صدرت في وبعد عام 2008

رابط الكود النهائي: الاستعلام عن الأفلام التي صدرت في وبعد عام 2008

نكتب الكود الآتي في موقع db-fiddle.com في قسم Query

```sql
SELECT * FROM Movies WHERE ReleaseYear ⩾ 2008;
```

هذا الكود يقوم بجلب جميع السجلات من جدول الأفلام (`Movies`) حيث أنّ سنة الإصدار (`ReleaseYear`) أكبر من أو تساوي 2008. بمعنى آخر، يعيد الاستعلام جميع الأفلام التي تم إصدارها في عام 2008 أو بعد ذلك.

بشكل مبسط، يعني هذا أنه سيتم استرجاع كل البيانات من الأفلام التي صدرت في السنوات 2008 وبعد ذلك، وذلك باستخدام الأمر `SELECT *` لجميع الأعمدة (*) من الجدول `Movies`، والشرط `WHERE ReleaseYear >= 2008` لتحديد السجلات التي تنطبق عليها الشرط.

ملاحظات في دفترك

```sql
WHERE columnName ⩾ 'Value';
```

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

```
SQL
```

باستخدام هذا الشرط، يتم اختيار الصفوف حيث تكون قيمة العمود (columnName) أكبر من أو تساوي القيمة ('Value') المحددة. ويتم استخدام هذا النوع من الشروط في العديد من الاستعلامات لتحديد البيانات التي تتوافق مع معايير معينة بناءً على القيم في الجداول.

WHERE <=

الاستعلام عن الأفلام التي لها تقييم أقل أو يساوي 8.0

رابط الكود النهائي: الاستعلام عن الأفلام التي لها تقييم أقل أو يساوي 8.0

نكتب الكود الآتي في موقع db-fiddle.com في قسم Query

```sql
SELECT * FROM Movies WHERE Rating ⩽ 8.0
```

1. **SELECT * FROM Movies**: يعني استرجاع كافة الأعمدة (* تعني جميع الأعمدة) من جدول يسمى "Movies".
2. **WHERE Rating <= 8.0**: يُضيف شرطاً للبحث، حيث يتم استرجاع الصفوف (السجلات) التي تحقق الشرط المحدد. في هذه الحالة، يتم استرجاع الأفلام التي يكون تقييمها (Rating) أقل من أو تساوي 8.0.

باختصار، هذا الأمر يسترجع كل الأعمدة من جدول "Movies" حيث تكون قيمة العمود "Rating" أقل من أو تساوي 8.0.

ملاحظات في دفترك

```sql
WHERE columnName ⩽ 'Value';
```

يستخدم لتحديد الصفوف في جدول قاعدة البيانات حيث تكون قيمة العمود المحدد (columnName) أقل من أو تساوي القيمة المحددة ('Value').

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

WHERE BETWEEN AND

الاستعلام عن الأفلام التي أصدرت بين عام 1990 وعام 2000

رابط الكود النهائي: الاستعلام عن الأفلام التي أصدرت بين عام 1990 وعام 2000

نكتب الكود الآتي في موقع db-fiddle.com في قسم Query

```sql
SELECT * FROM Movies WHERE ReleaseYear BETWEEN 1990 AND 2000;
```

يقوم بالتالي:

1. **SELECT * FROM Movies**: يعني استرجاع كافة الأعمدة (* تعني جميع الأعمدة) من جدول يسمى "Movies".
2. **WHERE ReleaseYear BETWEEN 1990 AND 2000**: يُضيف شرطاً للبحث، حيث يتم استرجاع الصفوف (السجلات) التي تحقق الشرط المحدد. في هذه الحالة، يتم استرجاع الأفلام التي تم إصدارها في الفترة بين عام 1990 وعام 2000، بما في ذلك السنتين الحدودية 1990 و 2000 أيضاً.

باختصار، هذا الأمر يساعد في استرجاع البيانات من جدول "Movies" حيث يكون عام الإصدار (ReleaseYear) في نطاق محدد معين، وفي هذه الحالة هو بين عامي 1990 و 2000.

ملاحظات في دفترك

```sql
WHERE columnName BETWEEN 'Value1' AND 'Value2';
```

يستخدم لتحديد الصفوف في جدول قاعدة البيانات حيث تكون قيمة العمود المحدد (columnName) ما بين القيمتين المحددتين (القيمة الأولى والثانية).

هذا الدليل قيد التحديث المستمر وسيتم إضافة تمارين أسبوعياً.

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

تواصل معي

يسرني التواصل معكم عبر البريد الإلكتروني

mail at LearnWithNaw dot net

كيف أستطيع تسهيل أعمالك؟

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF