

# Difference between DROP, TRUNCATE, DELETE

## 1. \*\*DROP (حذف)\*\*:

- يُستخدم لحذف الجداول بأكملها من قاعدة البيانات -
- عند استخدام هذا الأمر، يتم حذف الجدول بشكل نهائي وتختفي جميع البيانات المرتبطة به -
- يتم استخدام هذا الأمر بشكل أساسي عندما لا يكون هناك حاجة للبيانات في الجدول بعد الآن -

## 2. \*\*TRUNCATE (تفريغ)\*\*:

- يُستخدم لحذف جميع البيانات في الجدول، لكن يحتفظ الجدول بنفس هيكله -
- لأنه لا يسجل كل سجل محذوف في السجلات DELETE أسرع من TRUNCATE يكون -
- ، ولا يمكن استرجاع البيانات التي تم TRUNCATE ، لا يمكن التراجع عن DELETE على العكس من -
- حذفها بهذه الطريقة
- يستخدم عادةً عندما تحتاج إلى حذف كل البيانات ولكن لا تحتاج إلى الحفاظ على سجلات الحذف -

## 3. \*\*DELETE (حذف)\*\*:

- يُستخدم لحذف البيانات من الجداول بناءً على شروط معينة -
- لتحديد السجلات التي تحتاج إلى حذفها WHERE مع شروط DELETE يمكن استخدام -
- عن طريق استخدام اللوغات أو النسخ الاحتياطي إذا كانت متاحة DELETE يمكن التراجع عن عملية -
- عندما تحتاج إلى حذف بيانات معينة داخل الجدول دون حذف الجدول بأكمله DELETE يُفضل استخدام -
- أو بياناته

باختصار:::

تحذف جميع البيانات دون الجدول نفسه TRUNCATE

يحذف البيانات بناءً على شروط محددة: DELETE..

تحذف الجدول بأكمله DROP

## what is the meaning of (CRUD)

اختصار لأربع أفعال أساسية يتم تنفيذها عادة في نظم إدارة البيانات وقواعد البيانات (CRUD)

### 1. \*\*إنشاء (Create)\*\*:

يُشير إلى إنشاء سجل جديد في قاعدة البيانات -

مثال: إضافة مستخدم جديد إلى جدول المستخدمين -

### 2. \*\*قراءة (Read)\*\*:

يُشير إلى استرجاع البيانات من قاعدة البيانات -

مثال: عرض معلومات المستخدم من جدول المستخدمين -

### 3. \*\*تحديث (Update)\*\*:

يُشير إلى تعديل بيانات موجودة في قاعدة البيانات -

مثال: تغيير عنوان البريد الإلكتروني لمستخدم معين -

### 4. \*\*حذف (Delete)\*\*:

يُشير إلى حذف بيانات موجودة في قاعدة البيانات -

مثال: حذف مستخدم معين من جدول المستخدمين -

الوصول إلى وتعديل البيانات في قاعدة البيانات بشكل كامل CRUD تتيح هذه العمليات الأربعة وشامل

ببساطة هو لما تجمع جدول بنفسه. يعني لو عندك جدول فيه معلومات عن الموظفين، مثلاً

وعايز تشوف كل موظف ومديره، تستخدم الـ

"Self Join".

مثلاً، لو عندك جدول فيه معلومات الموظفين ومنها معلومات عن المدير لكل موظف، تقدر  
عشان تجمع بين الموظف والمدير اللي يشرف عليه "Self Join" تستخدم الـ

```
SELECT `employee_name`
```

```
FROM `employees`
```

```
SELF JOIN `employees` ON `employees`.`id` = `employees`.`
```

```
`manager_id`
```

يعرض لك اسم كل موظف بجانب اسم مديره، بناءً على المعلومات "Self Join" هنا الـ  
الموجودة في الجدول نفسه

## difference about join and union

### 1. \*\*JOIN (الانضمام)\*\*:

- لدمج الصفوف من جدولين أو أكثر استنادًا إلى عمود متصل بينهم JOIN يُستخدم الـ -
- يُستخدم أساسًا لاسترداد البيانات ودمجها من الجداول المتصلة في قاعدة البيانات -
- RIGHT JOIN ، (LEFT OUTER JOIN أو LEFT JOIN) ، INNER JOIN تشمل JOIN العمليات الـ -
- ، كل منها لها سلوك محدد (FULL OUTER JOIN أو FULL JOIN) ، و (RIGHT OUTER JOIN أو) بالنسبة للصفوف المتطابقة وغير المتطابقة

### 2. \*\*UNION (الاتحاد)\*\*:

- من جداول مختلفة في نتيجة واحدة SELECT لدمج نتائج تنفيذ عدة عبارات UNION يُستخدم الـ -
- يُستخدم لدمج البيانات من جداول مختلفة أو حتى من نفس الجدول باستخدام شروط مختلفة -
- نفس عدد الأعمدة، ويجب أن تكون الأعمدة المقابلة SELECT يتطلب أن تحتوي عبارات UNION الـ -
- على أنواع بيانات متوافقة SELECT في كل عبارة
- يتضمن UNION ALL بإزالة الصفوف المكررة من النتيجة، بينما الـ UNION افتراضيًا، يقوم الـ -
- جميع الصفوف، بما في ذلك الصفوف المكررة

لدمج الصفوف من جداول مختلفة استنادًا إلى العمود المتصل بينها، بينما JOIN باختصار، يُستخدم الـ  
من جداول مختلفة في نتيجة واحدة SELECT لدمج نتائج تنفيذ عدة عبارات UNION يُستخدم الـ

