اي الفرق بين ال authentication و authorization ؟؟؟

اول حاجه ال authentication ده اللي بيعمل validation للداتا و يتأكد هل البيانات ده صح وليك حق الوصول للسيستم ولا

زي مثلاً لمه تدخل ال username وال password

بيحصل validate للداتا ده لو صح

يبقا تمام هيدخل ع صفحه ال home للسيستم

خلاص دخلت ع السيستم الحمدلله و اتأكدنا إن ليك حق الوصول

طب هل دخلت كاادمن أو كيوزر عادي ؟؟

هنا هيجي دور ال authorization هو اللي بيحدد اي ال resource اللي تقدر توصلها و اي ال action اللي تقدر تعملها

يعني مثلاً الادمن ليه حق أنه يضيف أو يحذف يوزر

لكن اليوزر مايقدرش يعمل ده

الفرق بين ال sql و ال Nosql واي ال mysql ؟؟

1️⃣ sql

اختصار ل structure query language ده لغه اللي بنقدر ندير بيها الداتا بيز اللي هيا الأوامر اللي بنستخدمها عادي زي ال insert وال select وال update وده تعبر relational database management system (RDBMS) بيكون في قواعد ونظام محدد لازم تمشي عليها وانت بتستخدم اللغه ده

بتستخدم ال sql لمه يكون في relations بين الداتا اللي عايزه اخزنها وبتخزن الداتا في tables بطريقه منظمه

طب اي بقا اللي بيدعم لغه ال sql عشان الاومر ده تتنفذ ؟ هنا هيجي دور ال my sql أو oracle

◀ ال my sql ده software عادي بنزله عشان نقدر نستخدم أوامر ال sql

2️⃣ No sql

ده تعبر database management system (DBMS) بتدير الداتا بيز بردو وبتخزن بيانات لكن ده عكس ال sql مالهاش قيود معينه لازم تمشي عليها وبتخزن الداتا بشكل عشوائي مش لازم تكون table ممكن يكون document أو column أو graph بنستخدمها لمه يكون الداتا مافيش بينهم relations

زي ال mongodb أو ال Cassandra ودول بي support ال nosql

-- من مميزات ال nosql أنها بتقدر تخزن داتا كتير وتتعامل مع ال big data

لان لو الضغط زاد ع السيرفر تقدر تزود وتستخدم كذا سيرفر ويتوزع عليهم الضغط ده اللي هو نظام ال distributed system

عكس ال sql لو سيرفير واحد وزاد الضغط عليه بتزود ال ram أو ssd وده كلها حاجات ليها حد

-- من مميزات ال sql انها افضل في ال complex query عكس ال nosql ده مش هتكون احسن حاجه لو ال query معقدة وفي

Drop vs truncate vs delete

Drop: يستخدم لحذف جدول كامل من قاعدة البيانات بما فيه من بيانات وهيكل.

Truncate: يستخدم لحذف جميع الصفوف الموجودة في جدول مع الاحتفاظ بالهيكل الخارجي للجدول.

Delete: يستخدم لحذف صفوف محددة من جدول وليس الجدول بأكمله.

What is meaning crud?

CRUD هو اختصار لـ Create, Read, Update, Delete وهو مصطلح يستخدم في تطوير البرمجيات لوصف العمليات الأساسية التي يمكن أن يقوم بها نظام معين على قاعدة بيانات. وتشمل هذه العمليات:

- Create: إنشاء سجل جديد في قاعدة البيانات.

- Read: قراءة بيانات موجودة في قاعدة البيانات.

- Update: تحديث بيانات موجودة في قاعدة البيانات.

- Delete: حذف بيانات موجودة في قاعدة البيانات.

تعتبر هذه العمليات الأساسية الأربعة هي العمليات الأساسية التي يمكن أن يقوم بها أي نظام يتفاعل مع قاعدة البيانات. ويتم تطبيق مفهوم CRUD في العديد من تقنيات تطوير البرمجيات مثل RESTful APIs وقواعد البيانات العلاقية وغيرها.

Self join?

الـ Self Join هو نوع من الـ Join يتم استخدامه في قواعد البيانات لربط جدول بنفسه. يتم استخدام Self Join عندما تحتاج إلى استعلام بيانات من نفس الجدول وتحتاج إلى ربط الصفوف في الجدول بنفسه.

على سبيل المثال، لدينا يحتوي على معلومات حول الموظفين، ونريد عرض اسماء الموظفين وأسماء مديريهم. في هذه الحالة، يمكننا استخدام Self Join لربط الجدول بنفسه واستعلام البيانات بشكل مناسب.

مثال على Self Join:

SELECT e1.employeename AS employee, e2.employeename AS manager

FROM employees e1

JOIN employees e2 ON e1.managerid = e2.employeeid

في هذا المثال، تم استخدام Self Join لربط جدول الموظفين بنفسه عن طريق ربط حقل managerid في الجدول بحقل employeeid في نفس الجدول. يتم استعلام البيانات لعرض اسماء الموظفين وأسماء مديريهم.

الـ indexing هو عملية تنظيم البيانات في قاعدة البيانات بطريقة تسهل وتسرع عمليات البحث والاستعلام عن البيانات. تقوم عملية الـ indexing بإنشاء هيكل بيانات فرعي يحتوي على قيم معينة من البيانات وروابط للسجلات الأصلية في قاعدة البيانات.

هناك عدة أنواع من الـ indexing في قواعد البيانات، منها:

1. الـ Clustered Indexing: يقوم بترتيب البيانات في الجدول بناءً على قيمة معينة، مما يسرع عمليات البحث والفرز.

2. الـ Non-Clustered Indexing: يقوم بإنشاء هيكل بيانات فرعي يشير إلى مواقع السجلات الأصلية في الجدول، مما يسهل عمليات البحث والاستعلام.

3. الـ Unique Indexing: يتيح تخزين قيم فريدة لحقول معينة في الجدول، مما يضمن عدم تكرار القيم وتسهيل عمليات البحث.

4. الـ Composite Indexing: يسمح بإنشاء فهرس يشمل أكثر من حقل واحد في الجدول، مما يسرع عمليات البحث التي تتضمن هذه الحقول.

تستخدم عمليات الـ indexing في قواعد البيانات لتحسين أداء الاستعلامات وتقليل الوقت اللازم لاسترجاع البيانات المطلوبة، وتعتبر أساسية لتحسين أداء قواعد البيانات وتحسين تجربة المستخدم.