**Bài tập lý thuyết về SQL:**

* Có mấy loại khoá của bảng?
* Có 2 loại khóa bảng , khóa chính và khóa ngoại
* Định nghĩa và ý nghĩa của từng loại?
* Khóa chính được sử dụng để định danh duy nhất mỗi bản ghi trong table của CSDL.
* Khóa ngoại của một table được xem như con trỏ để trỏ tới khóa chính của table khác
* Một bảng có thể có nhiều loại khoá không?
* Có nhiều khóa ngoại trong một bảng nhưng chỉ có 1 khóa chính trong một bảng.
* Một bảng có thể có nhiều khoá cùng 1 loại hay không?
* Có nếu trong bảng đó có nhiều khóa phụ , còn lại chỉ được 1 khóa chính trong bảng
* Các giá trị của khoá có thể là NULL hay không?
* Có
* Có mấy loại Join, ý nghĩa của từng loại, lấy ví dụ minh hoạ.
* Có 4 loại join : INNER JOIN, LEFT JOIN, RIGHT JOIN, OUTER JOIN.
* INNER JOIN : Trả tất cả các hàng có ít nhất một giá trị ở cả hai bảng.

Vd:

SELECT a

FROM bang1

INNER JOIN bang2

ON bang1.a = bang2.a;

* LEFT JOIN :Trả lại tất cả các hàng từ bảng bên trái và phải thỏa mãn điều kiện và đúng từ bảng bên phải.
* RIGHT JOIN :Trả lại tất cả các hàng từ bảng bên phải và phải thỏa mãn điều kiện và đúng từ bảng bên trái.
* OUTER JOIN : Trả về tất cả các dòng đúng với 1 trong các bảng.
* Tìm hiểu về từ khoá UNION trong truy vấn dữ liệu? Có mấy loại UNION, cách sử dụng và ví dụ minh hoạ cho từng loại.
* UNION là toán tử để kết hợp tập hợp kết quả của hai hoặc nhiều câu lệnh select
* Có 2 loại toán tử :
* UNION: chỉ lưu giữ các bản ghi duy nhất xóa bỏ các bản ghi trùng lặp

VD: SELECT JOB FROM EMP WHERE DEPTNO = 20

UNION

SELECT JOB FROM EMP WHERE DEPTNO = ‘30’

* UNION ALL: lưu giữ tất cả các bản ghi, kể cả các bản sao

VD: SELECT JOB FROM EMP WHERE DEPTNO = 20

UNION ALL

SELECT JOB FROM EMP WHERE DEPTNO = ‘30’

* Tìm hiểu về Nested Query, Correlated Query. Phân biệt, lấy ví dụ minh hoạ.
* Nested Query được gọi là truy vấn con.
* Correlated Query được gọi là truy vấn con tương quan.
* Khác biệt
* Tìm nạp bởi truy vấn bên ngoài, thực hiện truy vấn bên trong bằng cách sử dụng giá trị của truy vấn bên ngoài, sử dụng các giá trị thu được từ truy vấn bên trong để đủ điều kiện hoặc loại bỏ truy vấn bên ngoài (tương quan)
* Truy vấn bên trong thực hiện đầu tiên và tìm một giá trị, truy vấn bên ngoài thực thi một lần bằng cách sử dụng giá trị từ truy vấn bên trong (truy vấn con)
* Vd:
* Nested Query :

SELECT ENAME FROM EMP

WHERE JOB IN (SELECT JOB FROM EMP WHERE JOB LIKE ‘CLERK’)

* Correlated Query:

SELECT ENAME FROM EMP

WHERE EXISTS (SELECT JOB FROM EMP WHERE JOB LIKE ‘CLERK’)

* Tìm hiểu về Index? Phân biệt global index và local index? Khi viết câu lệnh truy vấn, muốn các index được sử dụng phải chú ý gì? Lấy ví dụ minh hoạ.
* Là một cấu trúc dữ liệu để tăng hiệu suất truy vấn của cơ sở dữ liệu. Index cho phép cơ sở dữ liệu thực hiện một số câu truy vấn có điều kiện nhanh hơn so với thông thường
* Global index chỉ mục toàn cục là mối quan hệ một-nhiều, cho phép một phân vùng chỉ mục ánh xạ tới nhiều phân vùng bảng
* Local index chỉ mục cục bộ là ánh xạ 1-1 giữa phân vùng chỉ mục và phân vùng bảng.
* Muốn các index được sử dụng phải chú ý :
* Dùng Index để query cho nhanh
* Dùng Index khi mà việc lấy dữ liệu <15% số row trong bảng.
* Index những column nào dùng để nối giữa các bảng lẫn nhau.
* Không nên dùng Index cho các bảng nào chỉ có vài row.
* Khóa chính và khóa duy nhất tự động có index, nhưng nên có index cho khóa phụ
* Quá nhiều Index sẽ gây ảnh hưởng đến số liệu khi mà dữ liệu trong table bị thay đổi thứ tự theo index

Vd : CREATE INDEX E\_NAME ON EMP (ENAME);

1. Tìm hiểu về partition của bảng, nên tạo partition cho bảng trong trường hợp nào? Lấy ví dụ minh hoạ?

* Khái niệm : là việc phân chia một table thành những phần nhỏ theo một logic nhất định, được phân biệt bằng key, key này thường là tên column trong table.
* Trường hợp tạo partition
* Bảng lớn hơn 2 GB luôn được coi là ứng viên cho phân vùng.
* Bảng chứa dữ liệu lịch sử, những dữ liệu mới sẽ được thêm vào phân vùng mới.
* Khi nội dung của bảng cần được phân phối trên nhiều loại thiết bị storage khác nhau.
* Vd:

1. Phân biệt câu lệnh delete và truncate? Cách sử dụng?

* Delete xóa toàn bộ dữ liệu trong bảng tương ứng.Khi dùng Delete cần chú ý đến các điều kiện lọc sau Where. Có thể rollback lại dữ liệu sau khi xoá (với điều kiện thực hiện trên cùng một session).Nhưng không giải phóng không gian lưu trữ và tốc độ của delete là rất chậm .
* Truncate xóa toàn bộ dữ liệu trong bảng, với các bảng có ràng buộc dữ liệu, kể cả trong trường hợp không có dữ liệu ràng buộc cũng không thể truncate. Không thể rollback lại dữ liệu đã xóa. Nhưng có thể xóa dữ liệu và giải phóng không gian lưu trữ của bảng và tốc độ của truncate là nhanh hơn delete.