**Bài tập lý thuyết về SQL:**

1. Có mấy loại khoá của bảng? Định nghĩa và ý nghĩa của từng loại? Một bảng có thể có nhiều loại khoá không? Một bảng có thể có nhiều khoá cùng 1 loại hay không? Các giá trị của khoá có thể là NULL hay không?
2. Có mấy loại Join, ý nghĩa của từng loại, lấy ví dụ minh hoạ.
3. 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.
4. Tìm hiểu về Nested Query, Correlated Query. Phân biệt, lấy ví dụ minh hoạ.
5. 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.
6. Tìm hiểu về partition của bảng, nên tạo partition cho bảng trong trường hợp nào? Lấy ví dụ minh hoạ?
7. Phân biệt câu lệnh delete và truncate? Cách sử dụng?

Nguyễn Danh Vinh

Câu 1:

* Có 2 loại khóa của bảng: Khóa chính, khóa ngoại.
* Khóa chính: là 1 column hoặc một tập các column xác định tính duy nhất của cá rows ở trong bảng.
* Khóa ngoại: là 1 column hoặc một tập hợp các column có tham chiếu tới tính chất của bảng đó hoặc một bảng khác. Khóa ngoại xác định mối quan hệ giữa các bảng.
* Một bảng chỉ có duy nhất 1 khóa chính và có thể có nhiều khóa ngoại
* Một bảng cso thể có nhiều khóa cùng 1 loại
* Các giá trị của khóa có thể là Null

Câu 2:

Có 4 loại join trong SQL

* INNER JOIN
* Trả về tất cả các hàng giữa bảng 1 và bảng 2

SELECT ENAME, LOC,SAL  
FROM EMP join DEPT D on EMP.DEPTNO = D.DEPTNO

* LEFT OUTER JOIN
* Trả về tất cả giá trị từ bảng bên trái

SELECT ENAME, LOC,SAL  
FROM EMP left join DEPT D on EMP.DEPTNO = D.DEPTNO

* RIGHT OUTER JOIN
* Trả về tất cả giá trị từ bảng bên phải

SELECT ENAME, LOC,SAL  
FROM EMP right join DEPT D on EMP.DEPTNO = D.DEPTNO

* FULL OUTER JOIN
* Trả về tất cả giá trị của 2 bảng

SELECT ENAME, LOC,SAL  
FROM EMP full outer join DEPT D on EMP.DEPTNO = D.DEPTNO

Câu 3: Từ khóa UNION trong truy vấn dữ liệu là :

* Trong SQL được sử dụng để kết hợp kết quả của hai hoặc nhiều lệnh SELECT mà không trả về bất cứ bản ghi trùng lặp nào.
* Có 2 loại UNION:
* UNION: Kết hợp kết quả của nhiều câu hỏi với nhau, chỉ giữ lại một đại diện cho các mẫu tin trùng nhau

Vd:

Select job,DEPTNO from emp where deptno = 10  
Union  
Select job,DEPTNO from emp where deptno = 30;

Select job from emp where deptno = 30;

* UNION ALL: Kết hợp kết quả của nhiều câu hỏi với nhau, các mẫu tin trùng nhau cũng được lặp lại

Vd:

* Select job,DEPTNO from emp where deptno = 10  
  Union all  
  Select job,DEPTNO from emp where deptno = 30;

Câu 4:

* Nested Query: truy vấn lồng là một câu truy vấn được định nghĩa bên trong một câu truy vấn khác. Câu truy vấn được đặt bên trong một câu truy vấn khác được gọi là câu truy vấn con.

Vd: SELECT ENAME, DEPTNO, *MAX*(SAL)  
FROM EMP WHERE SAL IN (SELECT *MAX*(SAL) FROM EMP GROUP BY DEPTNO)  
GROUP BY DEPTNO,ENAME  
ORDER BY DEPTNO;

* Correlated Query :Truy vấn con tương quan không độc lập với truy vấn bên ngoài. Một truy vấn con tương quan là một truy vấn con sử dụng các giá trị từ truy vấn bên ngoài trong mệnh đề WHERE của nó. Quá trình thực hiện như sau: các truy vấn bên ngoài được thực hiện trước tiên và sau đó thi hành truy vấn con bên trong cho mỗi dòng kết quả của truy vấn bên ngoài

Vd:

Câu 5:

* Index trong SQL là bảng tra cứu đặc biệt mà công cụ tìm kiếm cơ sở dữ liệu có thể sử dụng để tăng nhanh thời gian và hiệu suất truy xuất dữ liệu.
* 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.   Tài liệu nói rằng "chỉ mục toàn cầu có thể được phân vùng bằng phương thức phạm vi hoặc hàm băm và nó có thể được xác định trên bất kỳ loại bảng nào được phân vùng hoặc không phân vùng".
* Local index: Chỉ mục cục bộ là ánh xạ 1-1 giữa phân vùng chỉ mục và phân vùng bảng.   Nói chung, các chỉ mục cục bộ cho phép một cách tiếp cận "phân chia và chinh phục" rõ ràng hơn để tạo ra các kế hoạch thực thi SQL nhanh với việc cắt bỏ phân vùng
* Khi viết câu lệnh truy vấn, muốn các index được sử dụng phải chú ý:

Có 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 i-ENAME ON EMP (ENAME);

Câu 6:

* Partition của bảng 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.
* Nên tạo partition cho bảng trong trường hợp:
* Khi bảng lớn hơn 2GB
* Bảng chứa nhiều partition phân tán trên các phân vùng đĩa khác nhau (như SSD, FC, SAS, SATA) khi đó sẽ được lưu trên các tablespace tương ứng (như DATA2019, DATA2018, DATA2017,...)
* Bảng có dữ liệu lịch sử: Chỉ cần DML trong tháng hiện tại (VD tháng 10/2019), còn lại trước đó Read Only

-Vd : CREATE TABLE invoices

(invoice\_no NUMBER NOT NULL,

invoice\_date DATE NOT NULL,

comments VARCHAR2(500))

PARTITION BY RANGE (invoice\_date)

(PARTITION invoices\_q1 VALUES LESS THAN (TO\_DATE('01/04/2001', 'DD/MM/YYYY')) TABLESPACE users,

PARTITION invoices\_q2 VALUES LESS THAN (TO\_DATE('01/07/2001', 'DD/MM/YYYY')) TABLESPACE users,

PARTITION invoices\_q3 VALUES LESS THAN (TO\_DATE('01/09/2001', 'DD/MM/YYYY')) TABLESPACE users,

PARTITION invoices\_q4 VALUES LESS THAN (TO\_DATE('01/01/2002', 'DD/MM/YYYY')) TABLESPACE users);

Câu 7 : Phân biệt câu lệnh delete và truncate

* DELETE cung cấp các lựa chọn để xóa những dòng dữ liệu thỏa mãn các điều kiện nhất định, như WHERE hoặc JOIN với các bảng khác.
* TRUNCATE không có lựa chọn nào, mà luôn cắt bỏ toàn bộ dữ liệu của bảng. Nói cách khác, ta không thể TRUNCATE 1 nửa hay 1 phần của bảng.

Cách sử dụng Delete :

* Lệnh DELETE trong SQL-SERVER (Transact-SQL) được dùng để xóa một hoặc nhiều bản ghi (hàng) từ một bảng trong SQL Server.

Bạn có thể xóa những bản ghi thỏa mãn điều kiện nhất định hoặc xóa toàn bộ bản ghi trong bảng một cách dễ dàng với DELETE.

Cách sử dụng Truncate

TRUNCATE được dùng để xóa tất cả bản ghi từ 1 bảng trong SQL Server Lệnh này hoạt động tương tự như lệnh Delete nhưng không có mệnh đề where