Chương 2.

1. Ràng buộc toàn vẹn là gì?

Là các ràng buộc mà toàn bộ dữ liệu trong cơ sở dữ liệu phải tuân theo để đảm bảo tính toàn vẹn và đúng đắn của dữ liệu

1. Tại sao các quan hệ phải có ràng buộc toàn vẹn?

Để đảm bảo tính đúng đắn và toàn vẹn của dữ liệu

1. Khóa chính (primary key) của một quan hệ Q

Là một khóa ứng viên được chọn từ tập các khóa ứng viên của Q, có thể xác 1 bộ duy nhất

1. Khóa ngoại (foreign key) của một quan hệ Q

Là một thuộc tính được tham chiếu từ một khóa chính hay một thuộc tính đảm bảo unique từ một quan hệ khác

1. Ràng buộc về miền giá trị của 1 thuộc tính

Là các giá trị của một thuộc tính phải đảm bảo nằm trong một miền xác định

1. Một thuộc tính có ràng buộc Unique?

Là các giá trị trong một thuộc tính phải đảm bảo là duy nhất và không được trùng nhau

1. Một thuộc tính có ràng buộc not null nghĩa là?

Là một thuộc tính không được có giá trị null

1. Một thuộc tính có ràng buộc Check (điều kiện) nghĩa là?

Là một thuộc tính có bộ giá trị đảm bảo được điều kiện Check được thiết lập trước đó

1. Câu lệnh khai báo, thêm và xóa các ràng buộc của quan hệ Q.

* Khi không cần khai báo tên cho ràng buộc

[thuộc tính] [kiểu ràng buộc]

* Khi cần gán tên cho ràng buộc

[tên – kiểu giá trị thuộc tính] CONSTRAINT [tên ràng buộc] [kiểu ràng buộc]

CONSTRAINT [tên ràng buộc] [kiểu ràng buộc]

Xóa – thêm

**ALTER TABLE [tên qh]** **DROP CONSTRAINT** [tên ràng buộc];

**ALTER TABLE** [tên qh] **ADD CONSTRAINT** [tên ràng buộc]

[kiểu ràng buộc]

1. Cho Q1(A int, B int), Q2(A int, C int). A trong Q2 là khóa ngoại tham chiếu qua A của Q1. Thao tác nào dưới đây không thực hiện được
2. Thêm 1 dòng có A = 1 vào Q2, nhưng trong Q1 chưa có dòng nào có A = 1.
3. Sửa 1 dòng trong Q2 có A = 1 thành A=0, nhưng trong Q1 chưa có dòng nào có A = 0.
4. Cả (1) và (2) đều đúng
5. Cả (1) và (2) đều sai
6. Trigger là gì?

Trigger là một chương trình con sẽ tự động được kích hoạt khi có biến cố trong CSDL

1. Ý nghĩa của biến cố, điều kiện và hành động trong trigger

Trigger sẽ được gọi khi có biến cố xảy ra do người lập trình chỉ định. Sau khi được kích hoạt, trigger sẽ kiểm tra điều kiện, nếu thỏa mã sẽ thực hiện các hành động được định sẵn trong trigger.

1. Ý nghĩa của For và After trong câu lệnh tạo trigger

For: ngay khi xảy ra biến cố thì mới thực hiện trigger

Alter: sau khi xảy ra biến cố thì mới thực hiện trigger

1. Ý nghĩa của câu lệnh dưới

Create trigger AA on Q after delete as declare @new int, @old int select @old=ol.B from deleted ol if (@old >2) print ('Notify Customer Relations');

Tạo một trigger AA trong bảng Q, sẽ được kích hoạt sau biến cố delete, sau khi xóa nếu giá trị cũ của thuộc tính B mà < 2 thì thông báo ('Notify Customer Relations');

1. Instead of trigger là gì?

Là một loại trigger đặc biệt, nó cho phép bạn bỏ qua câu lệnh INSERT, DELETE, UPDATE, trên một table hoặc view.

1. Ý nghĩa của câu lệnh dưới

Create trigger AA on Q after delete as declare @old int select @old=ol.B from deleted ol if (@old >2) Begin print ('Notify Customer Relations'); Rollback; end

Tạo một trigger AA trong bảng Q , được kích hoạt sau khi xảy ra biến cố delete. Khai báo biến @old để lưu giá trị vừa được xóa từ cột B của bảng tạm ol. Nếu giá trị này > 2 thì hoàn tác, thông báo và dừng chương trình

1. Cho

Create trigger AA on Q after delete as declare @old int select @old=ol.B from deleted ol if (@old >2) Begin print ('Notify Customer Relations'); Rollback; end

Từ khóa “after” trong câu lệnh trên có nghĩa là gì?

Sau khi biến cố delete xảy ra thì trigger mới hoạt động

1. Cho

Create trigger AA on Q for delete as declare @old int select @old=ol.B from deleted ol if (@old >2) Begin print ('Notify Customer Relations');

Từ khóa “for” trong câu lệnh trên có nghĩa là gì?

Ngay khi xảy ra biến cố delete thì trigger được kích hoạt

1. Ý nghĩa của câu lệnh dưới

Create trigger BB on Q1 instead of insert as Declare @newA int, @newB int Select @newA = n.A, @newB = n.B From Inserted n Insert into Q2 values(@newA, @newB)

Tạo một trigger BB trên bảng Q1, khi biến cố insert xảy ra trên Q1 thì sẽ bỏ qua để thực hiện các thao tác khác: gán giá trị được thêm vào cột A vào biến @newA, gán giá trị được thêm vào cột B vào biến @newB, sau đó đưa giá trị của 2 biến này vào bảng Q2

1. View là gì?

View giúp người ta tạo ra các bảng ảo ( Virtual table) chứa các thông tin, dữ liệu được lưu trữ sẵn cho người dùng sử dụng. Tuy nhiên, View không tồn tại như một cấu trúc lưu trữ dữ liệu trong csdl.

1. Mục đích của sử dụng view

* Để tập trung trên dữ liệu được xác định
* Để đơn giản hóa thao tác dữ liệu.
* Để tùy biến dữ liệu
* Để trích xuất dữ liệu tạo báo cáo
* Để bảo mật dữ liệu che giấu thông tin

1. Trong SQL server, cú pháp câu lệnh tạo, xóa, sửa view?

Tạo view

CREATE VIEW view\_name AS

SELECT column\_name(s)

FROM table\_name

WHERE condition;

Lệnh xóa view

DROP VIEW<name>

Lệnh sửa view

ALTER VIEW view\_name AS

SELECT column\_name(s)

FROM table\_name

WHERE condition

1. Phân loại View?

* Standard view
* Indexed view
* Partitioned view

1. Ý nghĩa của câu lệnh

Create view cc as Select A, max(B) as ln from Q1 group by A;

Tạo một view cc bằng cách chọn cột A, giá trị max tại cột B với bí danh ln, được gom nhóm theo A.

1. Cho PBan(MaPB int, TenPB char(20), MaNQL int). Viết câu lệnh thêm ràng buộc đảm bảo mỗi phòng ban có 1 người quản lý

**ALTER TABLE Pban.MaNQL** **ADD CONSTRAINT** MaNQLunique **UNIQUE**

1. Cho NV(MaNV int, Hoten char(30), Luong int) và câu lệnh alter table Emp add constraint chk\_luong\_min check (salary > 10000). Giải thích ý nghĩa của câu lệnh này.

Câu lệnh trên thêm ràng buộc là các giá trị trong cột salary phải trên 10000

1. Cho NV(MaNV int, Hoten char(30), Luong int, MaPB int) và view được định nghĩa như sau Create view V\_NV(MaPB, SoNV, TongLg, LgTB) as Select MaPB, Count(\*), Sum(Luong), Avg(Luong) From NV Group by MaPB;

Các câu SQL nào thực hiện được trên view V\_NV

1. Select \* From V\_NV
2. Delete from V\_NV Where MaPB = 1
3. Cả (1) và (2) đều đúng
4. Cả (1) và (2) đều sai