

# CHƯƠNG 3:

## XỬ LÝ THÔNG TIN TRÊN MÁY TÍNH: LẬP TRÌNH CƠ SỞ DỮ LIỆU

Khoa Khoa học và kỹ thuật thông tin  
Bộ môn Thiết bị di động và Công nghệ Web

# NỘI DUNG

1. Stored procedure.
2. Ràng buộc toàn vẹn.
3. Trigger.

# Lập trình Procedure

# Giới thiệu

- Một **Stored Procedure** là bao gồm các câu lệnh **Transact-SQL** và được lưu lại trong cơ sở dữ liệu.
- Để thực thi chỉ cần gọi ra.
- **Transact-SQL (T-SQL)** là một ngôn ngữ lập trình được sử dụng làm trung gian giữa cơ sở dữ liệu và các ứng dụng. Nó tương đối dễ học vì thực chất nó được tạo bởi hầu hết là các lệnh **SQL**.

# Lợi ích của Store procedure

- Module hóa: Chỉ cần viết Stored Procedure 1 lần, sau đó có thể gọi nó nhiều lần ở trong ứng dụng.
- Thực thi nhanh hơn: Stored Procedure sẽ được biên dịch và lưu vào bộ nhớ khi được tạo ra - thực thi nhanh hơn so với việc gửi từng đoạn lệnh SQL tới SQL Server.
- Giảm tải băng thông: gom các câu lệnh SQL vào 1 Stored Procedure và chỉ phải gọi đến 1 lần duy nhất qua network thay vì phải gọi nhiều lần.

# Cú pháp

— Khai báo một store procedure:

```
CREATE PROCEDURE procedure_name  
AS  
sql_statement
```

— Thực thi store procedure:

```
EXEC procedure_name;
```

# Ví dụ

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

# Ví dụ

— Viết procedure liệt kê danh sách tất cả các khách hàng.

```
CREATE PROCEDURE Danh_sach_khach_hang
AS
    SELECT * FROM Customers
```

— Thực thi PROCEDURE:

```
EXEC Danh_sach_khach_hang;
```



# PROCEDURE có tham số

- Ta có thể truyền vào các tham số đầu vào cho một Procedure.  
Một Procedure có thể có 1 hoặc nhiều tham số.
- Có 3 trường hợp tham số cho Procedure là:
  - + Một tham số vào (input).
  - + Nhiều tham số vào (multiple input).
  - + Tham số ra (output).

# Ví dụ

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

# Ví dụ trường hợp 1 tham số

## — Khai báo Procedure:

```
CREATE PROCEDURE SelectAllCustomers @City nvarchar(30)
AS
    SELECT * FROM Customers WHERE City = @City
```

## — Gọi thực thi Procedure:

```
EXEC SelectAllCustomers 'London'
```

# Ví dụ trường hợp nhiều tham số

— Khai báo procedure:

```
CREATE PROCEDURE SelectAllCustomers @City nvarchar(30),  
@PostalCode nvarchar(10)  
AS
```

```
    SELECT * FROM Customers WHERE City = @City AND  
    PostalCode = @PostalCode
```

— Thực thi procedure:

```
EXEC SelectAllCustomers 'London', 'WA1 1DP';
```

# Ví dụ trường hợp tham số ra

— Khai báo procedure:

```
CREATE PROCEDURE SelectCustomerName @CustomerID char(2),
@CustomerName nvarchar(100) OUTPUT
AS
    SELECT @CustomerName=CustomerName FROM Customers WHERE
    CustomerID = @CustomerID
```

— Thực thi procedure:

```
DECLARE @CName nvarchar(100)
EXEC SelectCustomerName 2, @CName OUTPUT --tức là @CustomerID=2
print @CName
```

# Ràng buộc toàn vẹn

# Nội dung

1. Giới thiệu ràng buộc toàn vẹn (RBTV)
2. Các đặc trưng của một RBTV
3. Phân loại RBTV

# 1. Giới thiệu

— Ràng buộc toàn vẹn là các quy định, điều kiện từ ứng dụng thực tế, các điều kiện này là bất biến.

⇒ Vì thế phải luôn đảm bảo cơ sở dữ liệu thoả ràng buộc toàn vẹn sau mỗi thao tác làm thay đổi tình trạng của cơ sở dữ liệu.



# 1. Giới thiệu

- Ví dụ về ràng buộc toàn vẹn:
  - + Giới tính của học viên chỉ có thể là ‘Nam’ hoặc ‘Nu’.
  - + Điểm thi của học viên phải nằm trong khoảng từ 0 đến 10.

## 2. Các đặc trưng của một RBTV

2.1 Nội dung

2.2 Bối cảnh

2.3 Bảng tầm ảnh hưởng

## 2.1 Nội dung

- Mô tả chặt chẽ ý nghĩa của ràng buộc toàn vẹn.
- Nội dung được phát biểu bằng ngôn ngữ tự nhiên hoặc bằng ngôn ngữ hình thức (ngôn ngữ tân từ, đại số quan hệ, mã giả,...)
  - + Ngôn ngữ tự nhiên: dễ hiểu nhưng không chặt chẽ, logic.
  - + Ngôn ngữ hình thức: chặt chẽ, cô đọng.

## 2.2 Bối cảnh

- Là tập các quan hệ (bảng) khi thao tác trên những quan hệ đó có khả năng làm cho ràng buộc bị vi phạm.
- Đó là những quan hệ có thể vi phạm ràng buộc toàn vẹn khi thực hiện các thao tác thêm, xóa, sửa.

## 2.3 Bảng tầm ảnh hưởng (1)

- Nhằm xác định khi nào tiến hành kiểm tra ràng buộc toàn vẹn.  
Thao tác nào thực hiện có thể làm vi phạm ràng buộc toàn vẹn.
- Phạm vi ảnh hưởng của một ràng buộc toàn vẹn được biểu diễn bằng một bảng 2 chiều gọi là *bảng tầm ảnh hưởng*.

## 2.3 Bảng tầm ảnh hưởng (2)

### Một số quy định

- Những thuộc tính khoá (những thuộc tính nằm trong khoá chính của quan hệ) không được phép sửa giá trị.
- Thao tác thêm và xoá xét trên **một** bộ của quan hệ. Thao tác sửa xét sửa từng thuộc tính trên **một** bộ của quan hệ
- Trước khi xét thao tác thực hiện có thể làm vi phạm ràng buộc hay không thì CSDL phải thoả ràng buộc toàn vẹn trước.

## 2.3 Bảng tầm ảnh hưởng (3)

### — Bảng tầm ảnh hưởng của một ràng buộc

Ràng buộc Ri	Thêm	Xóa	Sửa
Quan hệ 1			
.....			
Quan hệ n			

- + : thực hiện thao tác có thể làm vi phạm RBTV
- : thực hiện thao tác không làm vi phạm RBTV
- +(A) : có thể làm vi phạm RBTV khi sửa trên thuộc tính A
- (\*) : không vi phạm RBTV do thao tác không thực hiện được

## 3. Phân loại

3.1 RBTV có bối cảnh trên 1 quan hệ

3.2 RBTV có bối cảnh trên nhiều quan hệ



## 3.1 RBTV có bối cảnh 1 quan hệ

3.1.1 RBTV miền giá trị.

3.1.2 RBTV liên thuộc tính

3.1.3 RBTV liên bộ

# Lược đồ CSDL quản lý giáo vụ

**HOCVIEN** (MAHV, HO, TEN, NGSINH, GIOITINH, NOISINH, CMND, MALOP)

**LOP** (MALOP, TENLOP, TRGLOP, SISO, MAGVCN)

**KHOA** (MAKHOA, TENKHOA, NGTLAP, TRGKHOA)

**MONHOC** (MAMH, TENMH, TCLT, TCTH, MAKHOA)

**DIEUKIEN** (MAMH, MAMH\_TRUOC)

**GIAOVIEN**(MAGV,HOTEN,HOCVI,HOCHAM,GIOITINH,NGSINH,NGVL,  
HESO, MUCLUONG, MAKHOA)

**GIANGDAY**(MALOP,MAMH,MAGV,HOCKY, NAM,TUNGAY,DENNGAY)

**KETQUATHI** (MAHV, MAMH, LANTHI, NGTHI, DIEM, KQUA)

## 3.1.1 Ràng buộc miền giá trị

— Là tập giá trị mà một thuộc tính có thể nhận.

— **R1**: Giới tính của học viên chỉ là Nam hoặc Nữ

+ Nội dung:

$\forall hv \in HOCVIEN (hv.Gioitinh \in \{'Nam', 'Nữ'\})$

+ Bối cảnh: HOCVIEN

+ Bảng tầm ảnh hưởng:

R1	Thêm	Xóa	Sửa
HOCVIEN	+	-	+(Gioitinh)

## 3.1.1 Ràng buộc miền giá trị

- ♦ **R2:** Điểm thi của học viên phải nằm trong khoảng từ 0 đến 10.

+ Nội dung:

$\forall kq \in \text{KETQUATHI} (kq.diem \geq 0 \wedge kq.diem \leq 10)$

Hoặc:  $\forall kq \in \text{KETQUATHI} (kq.diem \in [0..10])$

+ Bối cảnh: KETQUATHI

+ Bảng tầm ảnh hưởng:

R2	Thêm	Xóa	Sửa
KETQUATHI	+	-	+(diem)

## 3.1.2 Ràng buộc liên thuộc tính

- Là ràng buộc giữa các thuộc tính với nhau trên 1 bộ của quan hệ
- **R3**: Ngày bắt đầu (TUNGAY) giảng dạy một môn học cho một lớp luôn nhỏ hơn ngày kết thúc (DENNGAY)
  - + Nội dung:  
 $\forall gd \in GIANGDAY (gd.TUNGAY < gd.DENNGAY)$
  - + Bối cảnh : GIANGDAY
  - + Bảng tầm ảnh hưởng:

R3	Thêm	Xóa	Sửa
GIANGDAY	+	-	+(Tungay, Denngay)

## 3.1.2 Ràng buộc liên thuộc tính

— **R4:** Nếu điểm thi < 5 thì kết quả là rớt (ROT), ngược lại là đậu (DAU)

+ Nội dung:

$\forall k \in \text{KETQUATHI}$

$((k.\text{diem} < 5 \rightarrow k.\text{kqua} = \text{'ROT'}) \vee (k.\text{diem} \geq 5 \rightarrow k.\text{kqua} = \text{'DAU'}))$

+ Bối cảnh : KETQUATHI

+ Bảng tầm ảnh hưởng:

R4	Thêm	Xóa	Sửa
KETQUATHI	+	-	+(diem, kqua)

## 3.1.3 Ràng buộc liên bộ

- Là ràng buộc giữa các bộ trên cùng một quan hệ (có thể liên quan đến nhiều thuộc tính).
- **R5:** Tất cả các học viên phải có mã số phân biệt với nhau
  - + Nội dung:  
 $\forall h_1, h_2 \in \text{HOCVIEN}: \text{Nếu } h_1 \neq h_2 \text{ thì } h_1.\text{Mahv} \neq h_2.\text{Mahv}$
  - + Bối cảnh: quan hệ HOCVIEN
  - + Bảng tầm ảnh hưởng:

R5	Thêm	Xóa	Sửa
HOCVIEN	+	-	-(*)

## 3.1.3 Ràng buộc liên bộ

- Là ràng buộc giữa các bộ trên cùng một quan hệ (có thể liên quan đến nhiều thuộc tính).
- **R6:** Tất cả các KHOA phải có tenkhoa phân biệt với nhau
  - + Nội dung:  

$$\forall k_1, k_2 \in \text{KHOA} (k_1 \neq k_2 \rightarrow k_1.\text{Tenkhoa} \neq k_2.\text{Tenkhoa})$$
  - + Bối cảnh: KHOA
  - + Bảng tầm ảnh hưởng:

R6	Thêm	Xóa	Sửa
KHOA	+	-	+(tenkhoa)



## 3.1.3 Ràng buộc liên bộ

— **R7:** Các giáo viên có cùng học vị, cùng hệ số lương thì mức lương sẽ bằng nhau

■ Nội dung:

$$\forall gv_1, gv_2 \in GIAOVIEN( \\ (gv_1.Hocvi = gv_2.Hocvi) \wedge (gv_1.Heso = gv_2.Heso) \rightarrow \\ gv_1.Mucluong = gv_2.Mucluong$$

■ Bối cảnh: quan hệ GIAOVIEN

■ Bảng tầm ảnh hưởng:

R7	Thêm	Xóa	Sửa
GIAOVIEN	+	-	+(Hocvi, Heso, Mucluong)

## 3.2 RBTV có bối cảnh nhiều quan hệ

3.2.1 RBTV tham chiếu (khoá ngoại, phụ thuộc tồn tại)

3.2.2 RBTV liên thuộc tính

3.2.3 RBTV do thuộc tính tổng hợp

3.2.4 RBTV do chu trình trong lược đồ biểu diễn quan hệ

## 3.2.1 Ràng buộc tham chiếu

- Là ràng buộc quy định giá trị thuộc tính trong một bộ của quan hệ R phải phụ thuộc vào sự tồn tại của một bộ trong quan hệ S.
- RBTV tham chiếu còn gọi là ràng buộc phụ thuộc tồn tại hay ràng buộc khóa ngoại.

# 3.2.1 Ràng buộc tham chiếu

– **R8:** Học viên thi một môn học nào đó thì môn học đó phải có trong danh sách các môn học.

+ Nội dung:

- $\forall k \in \text{KETQUATHI} (\exists m \in \text{MONHOC} (k.\text{Mamh} = m.\text{Mamh}))$
- Hoặc:  $\text{KETQUATHI}[\text{Mamh}] \subseteq \text{MONHOC}[\text{Mamh}]$

+ Bối cảnh: KETQUATHI, MONHOC

+ Bảng tầm ảnh hưởng:

R8	Thêm	Xóa	Sửa
KETQUATHI	+	-	-(*)
MONHOC	-	+	-(*)

-(\*): Không sửa được mamh trong bảng monhoc, ketquathi

## 3.2.1 Ràng buộc tham chiếu

— **R9:** Môn học phải do một Khoa quản lý.

+ Nội dung:

- $\forall m \in \text{MONHOC} (\exists k \in \text{KHOA} (m.\text{makhoa} = k.\text{makhoa}))$
- Hoặc:  $\text{MONHOC}[\text{makhoa}] \subseteq \text{KHOA}[\text{makhoa}]$

+ Bối cảnh: quan hệ KHOA, MONHOC

+ Bảng tầm ảnh hưởng:

R9	Thêm	Xóa	Sửa
MONHOC	+	-	+(makhoa)
KHOA	-	+	-(*)

-(\*) : Không sửa được makhoa trong KHOA

## 3.2.2 Ràng buộc liên thuộc tính

- Là ràng buộc giữa các thuộc tính trên những quan hệ khác nhau.
- **R10**: Ngày giáo viên giảng dạy một môn học (TUNGAY) phải lớn hơn hoặc bằng ngày giáo viên đó vào làm.

+ Nội dung:  $\forall gd \in GIANGDAY$

$$\forall gv \in GIAOVIEN( gd.Magv = gv.Magv \rightarrow \\ gd.TUNGAY \geq gv.NGVL)$$

• Hoặc:  $\forall gd \in GIANGDAY$

$$\neg \exists gv \in GIAOVIEN( gd.Magv = gv.Magv \\ \wedge (gv.NGVL > gd.TUNGAY) )$$

## 3.2.2 Ràng buộc liên thuộc tính

- **R10:** Ngày giáo viên giảng dạy một môn học phải lớn hơn hoặc bằng ngày giáo viên đó vào làm.
  - + Bối cảnh: GIANGDAY, GIAOVIEN
  - + Bảng tầm ảnh hưởng:

R10	Thêm	Xóa	Sửa
GIANGDAY	+	-	+(Tungay, magv)
GIAOVIEN	-	-	+(Ngvl)

### 3.2.3 RBTV do thuộc tính tổng hợp

- Là ràng buộc giữa các thuộc tính, các bộ trên những quan hệ khác nhau.
- Thuộc tính tổng hợp là thuộc tính được tính toán từ giá trị của các thuộc tính khác, các bộ khác.
- Cho lược đồ CSDL:  
    HOADON(Sohd, Nghd, Makh, Trigia)  
    CTHD(Sohd, Masp, Soluong, Gia, Thanh tien)
- ♦ Ví dụ RBTV do thuộc tính tổng hợp:  
    Trị giá của một hoá đơn bằng tổng thành tiền của các chi tiết thuộc hoá đơn đó.



## 3.2.3 RBTV do thuộc tính tổng hợp

- Ví dụ RBTV do thuộc tính tổng hợp:

Trị giá của một hoá đơn bằng tổng thành tiền của các chi tiết thuộc hoá đơn đó

<b>HOADON</b>			
<b>SOHD</b>	<b>NGHD</b>	<b>MAKH</b>	<b>TRIGIA</b>
1001	23/07/2006	KH01	215,000
1002	12/08/2006	KH01	420,000
1003	23/08/2006	KH02	98,000
1004	02/09/2019	KH03	0

<b>CTHD</b>				
<b>SOHD</b>	<b>MASP</b>	<b>SOLUONG</b>	<b>GIA</b>	<b>THANH TIEN</b>
1001	TV02	10	5,000	50,000
1001	ST01	5	20,000	100,000
1001	BC01	5	3,000	15,000
1001	BC02	10	4,000	40,000
1001	ST08	10	1,000	10,000
1002	BC04	20	13,000	260,000
1002	BB01	20	5,000	100,000
1002	BB02	20	3,000	60,000
1003	BB03	7	14,000	98,000

## 3.2.3 RBTV do thuộc tính tổng hợp

— Cho lược đồ CSDL

SANPHAM(Masp, Tensp, Nuocsx, Gia)

KHACHHANG(Makh, Hoten, Doanhso)

HOADON(Sohd, Nghd, Makh, Trigia)

CTHD(Sohd, Masp, Soluong, Gia)

### 3.2.3 RBTV do thuộc tính tổng hợp

– **R12**: Doanh số của một khách hàng bằng tổng trị giá các hoá đơn của khách hàng đó.

+ Nội dung:

$$\forall kh \in KHACHHANG( \\ kh.Doanhso = \sum_{(hd \in HOADON: hd.Makh=kh.Makh)} (hd.Trigial))$$

Hoặc

$$\forall kh \in KHACHHANG( \\ kh.Doanhso = SUM\{hd.trigial \mid hd \in HOADON (hd.Makh=kh.Makh)\} \\ )$$

## 3.2.3 RBTV do thuộc tính tổng hợp

- Doanh số của một khách hàng bằng tổng trị giá các hoá đơn mà khách hàng đó đã mua
  - + Bối cảnh: KHACHHANG, HOADON
  - + Bảng tầm ảnh hưởng:

R12	Thêm	Xóa	Sửa
KHACHHANG	+	-	+(Doanhso)
HOADON	+	+	+(Makh,Trigia)

## 3.2.3 RBTV do thuộc tính tổng hợp

**HOCVIEN** (MAHV, HO, TEN, NGSINH, GIOITINH, CMND, MALOP)

**LOP** (MALOP, TENLOP, TRGLOP, SISO, MAGVCN)

— **R13**: Số của một lớp là số lượng học viên thuộc lớp đó.

+ Nội dung:

$$\forall I \in \text{LOP}( \\ I.\text{Siso} = \text{Count}_{(hv \in \text{HOCVIEN}: hv.\text{Malop} = I.\text{Malop})}(\text{mahv}))$$

Hoặc

$$\forall I \in \text{LOP}( \\ I.\text{Siso} = \text{Count}_{\{hv.\text{mahv} \mid hv \in \text{HOCVIEN} (hv.\text{malop} = I.\text{malop})\}} )$$

## 3.2.3 RBTV do thuộc tính tổng hợp

- **R13:** Số của một lớp là số lượng học viên thuộc lớp đó
  - + Bối cảnh: quan hệ LOP, HOCVIEN
  - + Bảng tầm ảnh hưởng:

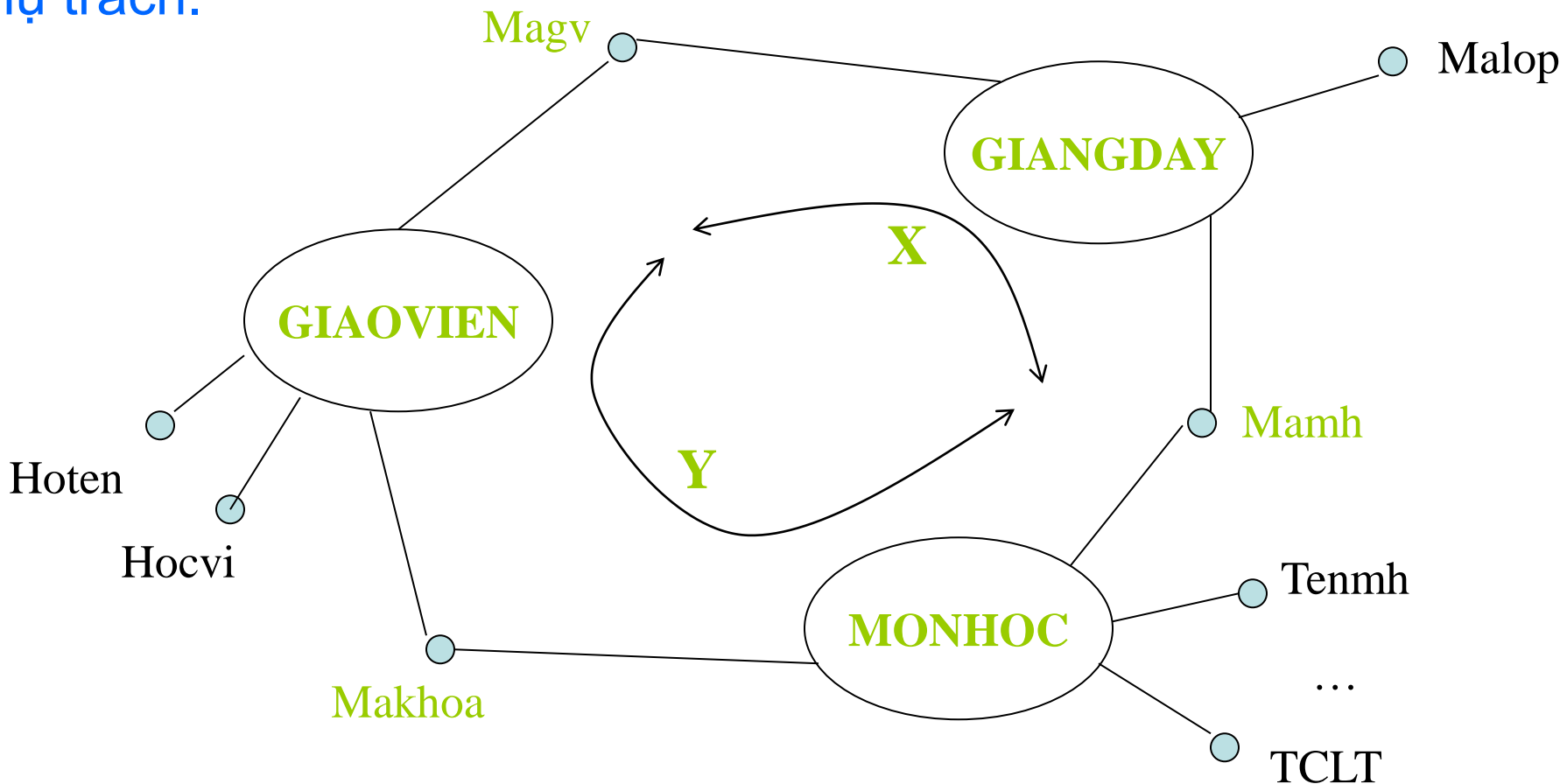
R13	Thêm	Xóa	Sửa
LOP	+	-	+(Siso)
HOCVIEN	+	+	+(Malop)

## 3.2.4 Do hiện diện của chu trình (1)

Biểu diễn lược đồ quan hệ dưới dạng đồ thị:

- + Quan hệ được biểu diễn bằng nút tròn rỗng to.
- + Thuộc tính được biểu diễn bằng nút tròn đặc nhỏ.
- + Tất cả các nút đều được chỉ rõ bằng tên của quan hệ hoặc thuộc tính. Thuộc tính thuộc một quan hệ được biểu diễn bởi một cung nối giữa nút tròn to và nút tròn nhỏ.
- + Nếu đồ thị biểu diễn xuất hiện một đường khép kín  
=> lược đồ CSDL có sự hiện diện của chu trình.

- **R14:** giáo viên chỉ được phân công giảng dạy những môn thuộc khoa giáo viên đó phụ trách.





# Bảng tầm ảnh hưởng tổng hợp (1)

— Bảng tầm ảnh hưởng tổng hợp của m ràng buộc trên n quan hệ bối cảnh

	$QH_1$			$QH_2$			...	$QH_n$		
	T	X	S	T	X	S	...	T	X	S
$R1$										
$R2$										
...										
$Rm$										

# Bảng tầm ảnh hưởng tổng hợp (2)

	HOCVIEN			GIAOVIEN			LOP			MONHOC			GIANGDAY			KETQUA THI		
	T	X	S	T	X	S	T	X	S	T	X	S	T	X	S	T	X	S
R1	+	-	+															
R2													+	-	+			
R3	+	-	_*															
R4				+	-	+												
R5										-	+	_*				+	-	_*
R6				-	-	+							+	-	+			
R7	-	-	+										-	-	+	+	-	+
R8	+	+	+				+	-	+									
R9				-	-	+				-	-	+	+	-	+			

# Lập trình Trigger

# Giới thiệu

- **Trigger** là **stored procedure** đặc biệt sẽ tự động thực hiện khi có hành động/sự kiện thay đổi dữ liệu (data modification) xảy ra như Update, Insert hoặc Delete lên 1 table mà trigger bảo vệ (được cài đặt).
- Trigger được dùng để đảm bảo tính toàn vẹn dữ liệu (Data Integrity) hoặc thực hiện các quy tắc nghiệp vụ (business rules) nào đó.

# Các tính chất của Trigger

- Liên kết với Table.
  - + Trigger được định nghĩa trên 1 table cụ thể gọi là Trigger liên kết Table.
- Thực hiện tự động.
  - + Nếu có các hành động INSERT, UPDATE, DELETE mà trigger định nghĩa nó sẽ tự động thực hiện.
  - + Nó không được gọi trực tiếp và không chấp nhận tham số.

# Các tính chất của Trigger (tt)

## — Là 1 giao tác:

- + Trigger và phát biểu tạo ra nó được thực hiện như là 1 giao tác , có thể rollback bất cứ đâu trong trigger.
- + Nếu có rollback và nó được thực hiện thì toàn Transaction sẽ Rollback.
- + Nếu trong 1 đoạn có rollback và nó được thực hiện thì toàn đoạn sẽ bị hủy các phát biểu sau Rollback sẽ không thực hiện.

## — Nên tránh lạm dụng Rollback trong Trigger vì nó phải Làm undo các thao tác trước đó → kiểm tra tính hợp lệ trước khi bắt đầu transaction.

# Lợi ích của Trigger

- Bổ sung dây chuyền (cascade).
  - + Các bổ sung dây chuyền bằng Trigger làm giảm ược các coding cần thiết cho sự thay đổi trên các bảng liên quan.
- Làm tăng cường toàn vẹn dữ liệu.
  - + Do có thể tham chiếu nhiều cột trên các bảng nên có tác dụng hơn check. Tuy vậy các ràng buộc ưu tiên kiểm tra trước, nếu vi phạm trigger không thực thi.
- Thông báo lỗi do người dùng định nghĩa.
  - + Các thông báo Rule, Default, check
- Bảo quản dữ liệu không tiêu chuẩn hóa.

# Bảo quản dữ liệu không tiêu chuẩn hóa.

Ví dụ:

HOADON (MSHD.NGAYHD,**TONGTIEN**)

CTHOADON (MSHD, MSMH, **SOLUONG, DONGIA** )

*Khi sửa số lượng, đơn giá dẫn đến ảnh hưởng tổng tiền.*



# Lưu ý

- 1 Bảng có thể viết nhiều trigger.
- Trigger không thể tạo trên View và bảng tạm.
- Trigger không thể trả về các tập kết quả.

# Trigger

- Khi trigger được thực thi, SQL tự động tạo ra 2 bảng tạm với cùng cấu trúc với bảng mà trigger được định nghĩa trên đó.
- Bảng INSERTED chứa dữ liệu mới khi thực thi câu lệnh Insert hoặc câu lệnh Update.
- Bảng DELETED chứa dữ liệu bị xóa khi thực thi câu lệnh Delete hoặc chứa dữ liệu cũ khi thực thi câu lệnh Update.
- Hai bảng này chỉ tồn tại trong thời gian trigger xử lý và cục bộ cho mỗi trigger. Không thể thay đổi DL trên các bảng deleted và inserted trực tiếp, có thể dùng phát biểu select.

Hoạt động	Bảng INSERTED	Bảng DELETED
INSERT	dữ liệu mới được insert	không có dữ liệu
DELETE	không có dữ liệu	chứa dữ liệu bị xóa
UPDATE	chứa dữ liệu sau khi được cập nhật	chứa dữ liệu trước khi cập nhật

# Trigger

— Đối với thao tác insert:

Insert into HOADON values (1004, '01/09/2006', 'KH02', 180000)

HOADON			
SOHD	NGHD	MAKH	TRIGIA
1001	23/07/2006	KH01	320,000
1002	12/08/2006	KH01	840,000
1003	23/08/2006	KH02	100,000
1004	01/09/2006	KH02	180,000

INSERTED			
SOHD	NGHD	MAKH	TRIGIA
1004	01/09/2006	KH02	180,000

DELETED			
SOHD	NGHD	MAKH	TRIGIA

1004	01/09/2006	KH02	180,000
------	------------	------	---------

# Trigger

— Đối với thao tác delete

Delete from HOADON where soh=1004

HOADON			
SOHD	NGHD	MAKH	TRIGIA
1001	23/07/2006	KH01	320,000
1002	12/08/2006	KH01	840,000
1003	23/08/2006	KH02	100,000

INSERTED			
SOHD	NGHD	MAKH	TRIGIA

DELETED			
SOHD	NGHD	MAKH	TRIGIA
1004	01/09/2006	KH02	180,000

1004	01/09/2006	KH02	180,000
------	------------	------	---------



# Trigger

— Đối với thao tác update:

Update HOADON set makh='kh07', trigia=300000 Where sohd=1004

HOADON			
SOHD	NGHD	MAKH	TRIGIA
1001	23/07/2006	KH01	320,000
1002	12/08/2006	KH01	840,000
1003	23/08/2006	KH02	100,000
1004	01/09/2006	KH07	300,000

INSERTED			
SOHD	NGHD	MAKH	TRIGIA
1004	01/09/2006	KH07	300,000

DELETED			
SOHD	NGHD	MAKH	TRIGIA
1004	01/09/2006	KH02	180,000

# Các thao tác với trigger

- Tạo trigger.
- Sửa trigger.
- Xoá trigger.

# Tạo trigger

Create Trigger TenTrigger

On table

After for *insert, update, delete*

As

*phát biểu SQL*

# Ví dụ 1: Phát biểu: Một sinh viên (MSSV) không được thi quá 2 lần.

```
Create Trigger KiemTraThi2Lan
On KETQUA for Insert
As
Begin
    If (Select Count(*) from KETQUA a, INSERTED b
    Where a.MSSV=b.MSSV and a.MSMH=b.MSMH)>1
    Begin
        Print ' Khong the qua 2 lan'
        Rollback Transaction
    End
End
```



# Sửa trigger

Alter Trigger TenTrigger

On Tentable

After for *insert, update, delete*

AS

Phát biểu SQL

# Sửa trigger KiemTraThi2Lan.

```

Alter  Trigger KiemTraThi2Lan
On KETQUA for Insert
As begin
If (Select  Count(*) from KETQUA a,          INSERTED  b
Where a.MSSV=b.MSSV and  a.MSMH=b.MSMH) >2
Begin
    Print ' Khong the qua 2 lan'
    Rollback Transaction
End

```

# Xoá trigger

- Cú pháp: Drop Trigger TenTrigger
- Có thể làm mất hiệu lực tạm thời của Trigger lên 1 table
- Cú pháp :  

```
Alter Table    tenTable
    {Enable/ Disable } Trigger {all/TenTrigger}
```
- Vd :  

```
Alter      Table  KETQUA
    Disable Trigger  KiemTraThi2Lan
```
- Muốn có hiệu lực lại : dùng **Enable**

# Trigger

- **KHACHHANG** (MAKH, HOTEN, NGSINH, NGDK)
- **HOADON** (SOHD, NGHD, MAKH, TRIGIA)
- Ngày mua hàng (NGHD) của một khách hàng thành viên sẽ lớn hơn hoặc bằng ngày khách hàng đó đăng ký thành viên (NGDK).

	THÊM	XÓA	SỬA
KHACHHANG	-	-	+ (NGDK)
HOADON	+	-	+ (NGHD, MAKH)

# Trigger

```
CREATE TRIGGER nghd_ngdk_hoadon_insert
ON hoadon
AFTER INSERT
AS
    DECLARE @ng_muahang    smalldatetime
    DECLARE @ng_dangky     smalldatetime
    SELECT    @ng_muahang=nghd, @ng_dangky=ngdk
    FROM      khachhang, inserted
    WHERE     khachhang.makh=inserted.makh
    IF @ng_muahang< @ng_dangky
    BEGIN
        rollback transaction
        print 'ngay mua hang phai lon hon ngay dang ky'
    END;
```

# Trigger

— HOẶC:

```
CREATE TRIGGER nghd_hoadon_insert
ON hoadon
AFTER INSERT
AS
    IF (EXISTS (SELECT *
                FROM inserted i JOIN kháchhang kh ON i.makh=kh.makh
                WHERE i.nghd<kh.ngdk))
BEGIN
    rollback transaction
    print 'ngày mua hàng phải lớn hơn ngày đăng ký'
END
```

# Hoạt động khi update

- Có thể định nghĩa 1 trigger để giám sát việc cập nhật DL trên 1 cột đặc biệt bằng cách dùng phát biểu **if update**, nó cho phép thực hiện khi có sự cập nhật trên các cột đã chỉ định.
- Cú pháp: If update (column) [{AND | OR} update (column)]
- VD:

```
Create Trigger CapnhatMSSV
```

```
    If update (MSSV)
```

```
    Begin
```

```
        Print 'Không được cập nhật mã số NV'
```

```
        Rollback transaction
```

```
    End
```

# Trigger

```
CREATE TRIGGER nghd_ngdk_hoadon_update
ON hoadon
AFTER UPDATE
AS
IF (UPDATE (makh) OR UPDATE (nghd))
BEGIN
    DECLARE @ng_muahang      smalldatetime
    DECLARE @ng_dangky       smalldatetime
    SELECT  @ng_muahang=nghd, @ng_dangky=ngdk
    FROM    khachhang, inserted
    WHERE   khachhang.makh=inserted.makh
    IF @ng_muahang< @ng_dangky
    BEGIN
        rollback transaction
        print 'ngay mua hang phai lon hon ngay dang ky'
    END
END;
END;
```



# Trigger

HOẶC:

```
CREATE TRIGGER nghd_hoadon_update
ON hoadon
AFTER UPDATE
AS
IF (UPDATE (makh) OR UPDATE (nghd))
BEGIN
    IF (EXISTS (SELECT *
                FROM inserted i JOIN kháchhang kh ON i.makh=kh.makh
                WHERE i.nghd<kh.ngdk))
    BEGIN
        rollback transaction
        print 'ngày mua hàng phải lớn hơn ngày đăng ký'
    END
END
END
```

# Trigger

```
CREATE TRIGGER nghd_ngdk_khachhang_update
ON kháchhang
AFTER UPDATE
AS
    DECLARE    @ng_dangky  smalldatetime, @makhhang char(4)
    SELECT      @ng_dangky=ngdk, @makhhang=makh
    FROM        inserted
    IF (UPDATE (ngdk))
    BEGIN
        IF (EXISTS (SELECT *
                     FROM    hoadon
                     WHERE   makh=@makhhang AND @ng_dangky>nghd))
        BEGIN
            rollback transaction
            print 'thao tac sua ngay dang ky phai nho hon ngay hoa don'
        END
    END
END;
```

# Trigger

HOẶC:

```
CREATE TRIGGER nghd_ngdk_khachhang_update
ON kháchhang
AFTER UPDATE
AS
IF (UPDATE (ngdk))
BEGIN
    IF (EXISTS (SELECT *
                FROM inserted i JOIN hoadon hd ON i.makh=hd.makh
                WHERE hd.nghd<i.ngdk))
    BEGIN
        rollback transaction
        print 'Loi. Ngay dang ky phai nho hon ngay mua hang'
    END
    else print 'Chinh sua thanh cong'
END
END
```

# Tổng kết

- Store procedure là một dạng thủ tục, dùng để nhóm các câu lệnh SQL lại với nhau. Có 2 dạng là: không tham số và có tham số (tham số gồm 2 dạng là tham số vào và tham số ra).
- Trigger được thực thi tự động khi có hành vi thay đổi trên CSDL. Trigger không có tham số.

# TÀI LIỆU THAM KHẢO

1. Nguyễn Gia Tuấn Anh, Trương Châu Long, *Bài tập và bài giải SQL Server*, NXB Thanh niên (2005).
2. Đỗ Phúc, Nguyễn Đăng Ty, *Cơ sở dữ liệu*, NXB Đại học quốc gia TP HCM (2010).
3. Nguyễn Gia Tuấn Anh, Mai Văn Cường, Bùi Danh Hương, *Cơ sở dữ liệu nâng cao*, NXB Đại học quốc gia TP HCM (2019).
4. Itzik Ben-Gan, *Microsoft SQL Server 2012- TSQL Fundamentals*.



# BÀI TẬP

1. Nhập vào MAKH,HOTEN,... kiểm tra xem đã tồn tại chưa, nếu có hãy thông báo lỗi, nếu không hãy thêm thông tin đó.
2. Nhập SOHD, NGHD,... kiểm tra xem hóa đơn này tồn tại chưa, kiểm tra mã KH, mã NV, nếu tồn tại thì thông báo lỗi, không thì hãy thêm thông tin vào bảng HOADON.
3. Nhập vào MASP, hãy cho biết sản phẩm này được bán trong bao nhiêu hóa đơn rồi.
4. Hãy thêm thông tin khách hàng mới là họ tên của bạn, những thông tin khác tùy ý, thêm dữ liệu cho các bảng HOADON, CTHD tương ứng. Sau đó viết procedure: Nhập vào MAKH (cái bạn vừa thêm) kiểm tra xem có tồn tại thì xóa thông tin khách hàng đó và những dữ liệu liên quan, nếu không thì thông báo không có thông tin để xóa.
5. Thêm 1 cột số lần mua hàng của khách hàng vào bảng KHACHHANG, bổ sung dữ liệu cột đó bằng procedure.
6. Các câu trigger làm trong bài tập thực hành CSDL.