

## Lab 08

### Indexes

#### Phần I – Hướng dẫn thực hành từng bước

##### 1. Mục tiêu

- Định nghĩa và giải thích Indexes (chỉ mục).
- Hiểu về hiệu năng của Indexes và áp dụng vào tình huống cụ thể.
- Hiểu về Clustered Indexes.
- Hiểu về NonClustered Indexes.

##### 2. Thực hiện

**SQL Server 2012** sử dụng các chỉ mục để tìm dữ liệu khi truy vấn được xử lý.

Công cụ SQL Server sử dụng chỉ mục theo cách tương tự như học viên sử dụng chỉ mục cuốn sách. Ví dụ, xem xét rằng bạn cần phải tìm tất cả các tham khảo đến các câu lệnh INSERT trong một cuốn sách SQL. Cách tiếp cận ngay lập tức đã sử dụng sẽ quét từng trang của cuốn sách bắt đầu từ trang bắt đầu. Bạn đánh dấu mỗi lần từ INSERT được tìm thấy, cho đến tới cuối cuốn sách. Cách tiếp cận này rất tốn thời gian và công sức. Cách thứ hai là sử dụng chỉ mục ở phía sau của cuốn sách để tìm số trang cho mỗi lần xuất hiện của các câu lệnh INSERT. Cách thứ hai tạo ra các kết quả tương tự như cách đầu tiên, nhưng rất tiết kiệm thời gian.

Khi SQL Server đã không xác định bất kỳ chỉ mục để tìm kiếm, khi đó quá trình này tương tự như cách đầu tiên trong ví dụ, công cụ SQL cần phải ghé thăm mỗi hàng trong bảng. Trong thuật ngữ cơ sở dữ liệu, hành vi này được gọi là quét bảng, hoặc chỉ là quét.

Quét bảng không phải luôn phiền hà, nhưng đôi khi không thể tránh khỏi. Tuy nhiên, khi bảng lớn lên đến hàng ngàn và hàng triệu hàng và hơn thế nữa, quét trở nên chậm hơn và tốn kém hơn. Trong các trường hợp này, đặc biệt nên sử dụng chỉ mục.

Việc tạo hoặc loại bỏ các chỉ mục khỏi lược đồ cơ sở dữ liệu sẽ không ảnh hưởng đến mã của một ứng dụng. Chỉ mục hoạt động ở phía sau với sự hỗ trợ của công cụ

cơ sở dữ liệu. Hơn nữa, việc tạo ra một chỉ mục thích hợp có thể làm tăng đáng kể hiệu suất của ứng dụng.

**Bài thực hành 1:** Tạo một Indexes cho cột MA\_TINH\_THANH thuộc bảng TINH\_THANH.

Chỉ mục sẽ được tự động tạo ra khi các ràng buộc **PRIMARY KEY** và **UNIQUE** được định nghĩa trên bảng. Chỉ mục giảm các hoạt động nhập/xuất của đĩa và tiêu thụ các tài nguyên hệ thống ít hơn.

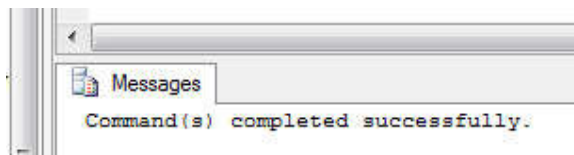
Chỉ mục trở đến vị trí của một hàng trên một trang dữ liệu thay vì tìm kiếm suốt cả bảng. Xem xét các sự kiện và hướng dẫn sau đây về chỉ mục:

- Chỉ mục làm tăng tốc độ các truy vấn nối các bảng hoặc thực hiện các hoạt động phân loại.
- Chỉ mục thực hiện sự duy nhất của các hàng nếu được định nghĩa khi bạn tạo chỉ mục.
- Chỉ mục được tạo ra và duy trì theo thứ tự tăng dần hoặc giảm dần.

**Bước 1:** Viết lệnh SQL như sau:

```
-- Tạo Indexes cho bảng Tỉnh Thành trên cột Ma_tinh_thanh  
CREATE INDEX ID_ma_tinh_thanh  
ON TinhThanh (Ma_tinh_thanh)
```

**Bước 2:** Thực thi câu lệnh SQL trên:



**Bài thực hành 2:** Tạo một Clustered Index cho cột Ma\_KH trong bảng Khách\_Hang.

Chỉ mục ghép cụm có thể được tạo ra trên một bảng sử dụng một cột không có các giá trị trùng lặp. Chỉ mục này sẽ tổ chức lại các bản ghi theo thứ tự tuần tự của những giá trị trong cột chỉ mục.

Chỉ mục ghép cụm được sử dụng để định vị một hàng đơn lẻ hoặc một loạt các hàng. Bắt đầu từ trang đầu tiên của chỉ mục, giá trị tìm kiếm được kiểm tra đối với mỗi giá trị khóa trên trang này. Khi giá trị khóa so khớp được tìm thấy, công cụ cơ sở dữ liệu chuyển đến trang được chỉ ra bằng giá trị đó như được trình bày trong hình 11.13.

Hàng mong muốn hoặc một loạt các hàng sau đó được truy cập.

Chỉ mục ghép cụm rất hữu ích cho các cột được tìm kiếm thường xuyên cho các giá trị khóa hoặc được truy cập theo thứ tự sắp xếp.

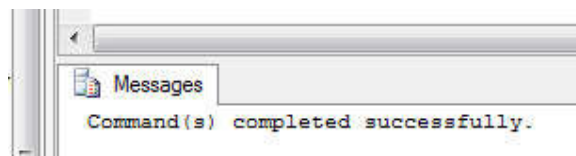
Chỉ mục ghép cụm được tự động tạo ra trên một bảng khi khóa chính được định nghĩa trên bảng đó. Trong bảng không có cột khóa chính, chỉ mục ghép cụm lý tưởng nên được định nghĩa trên:

- ✓ Cột khóa được tìm kiếm trên diện rộng.
- ✓ Cột được sử dụng trong các truy vấn trả lại các tập kết quả lớn.
- ✓ Cột có dữ liệu duy nhất.
- ✓ Cột được sử dụng trong phép nối bảng.

#### Bước 1: Gõ lệnh SQL:

```
-- Tạo Clustered Indexes cho bảng Khách Hàng trên cột Ma_KH  
CREATE CLUSTERED INDEX ID_Ma_Khach_Hang  
ON KhachHang (Ma_KH)
```

#### Bước 2: Thực thi lệnh SQL:



#### **Bài thực hành 3:** Tạo một NonClustered Indexes cho cột Email\_KH trên bảng Khach\_hang.

Các chỉ mục không ghép cụm rất hữu ích khi bạn cần nhiều cách để tìm kiếm dữ liệu. Một số thực tế và hướng dẫn phải được xem xét trước khi tạo ra một chỉ mục không ghép cụm như sau :

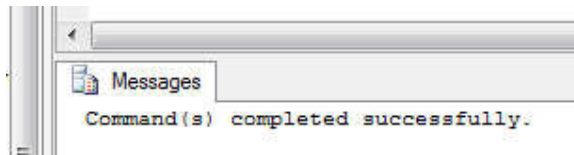
- ✓ Khi một chỉ mục ghép cụm được tái tạo hoặc tùy chọn DROP \_ EXISTING được sử dụng, SQL Server xây dựng lại các chỉ mục không ghép cụm hiện có.

- ✓ Một bảng có thể có đến 999 chỉ mục không ghép cụm.
- ✓ Tạo chỉ mục ghép cụm trước khi tạo ra một chỉ mục không ghép cụm. Cú pháp sau đây tạo ra một chỉ mục không ghép cụm.

#### Bước 1: Gõ lệnh SQL:

```
-- Tạo NonClustered Indexes cho bảng Khách Hàng trên cột Email_KH
CREATE NONCLUSTERED INDEX NID_Email_Khach_Hang
ON KháchHang (Email_KH)
```

#### Bước 2: Thực thi lệnh SQL sau:



Sự khác nhau	
Clustered Indexes	Non-Clustered Index
Được sử dụng trong các truy vấn trả lại các tập kết quả lớn	Được sử dụng trong các truy vấn không trả lại các tập kết quả lớn
Chỉ một chỉ mục ghép cụm có thể được tạo ra trên một bảng	Nhiều chỉ mục không ghép cụm có thể được tạo ra trên một bảng
Dữ liệu được lưu trữ theo một cách có sắp xếp trên khóa ghép cụm	Dữ liệu không được lưu trữ theo một cách có sắp xếp trên khóa không ghép cụm
Các nút lá của một chỉ mục ghép cụm chứa các trang dữ liệu	Những nút lá của một chỉ mục không ghép cụm chứa các trang chỉ mục

#### Phần II – Phụ lục CSDL sql cho bài thực hành

```
-- Database mẫu cho bài thực hành Lab 08 - Indexes

CREATE DATABASE QuanLyBanHang
GO -- Tạo CSDL

USE QuanLyBanHang
GO -- Sử dụng CSDL

-- Tạo bảng Loại Sản Phẩm
```

```
CREATE TABLE loaiSanPham(  
    Ma_loai INT IDENTITY(1,1) PRIMARY KEY,  
    Ten_loai NVARCHAR(50) NOT NULL,  
    Trang_thai BIT,  
    Meta_keyword NVARCHAR(128),  
    Meta_description NTEXT  
)  
GO  
  
-- Tạo bảng Sản Phẩm  
CREATE TABLE SanPham(  
    Ma_SP INT IDENTITY(1,1) PRIMARY KEY,  
    Ma_loai INT NOT NULL,  
    Ten_sp NVARCHAR(128) NOT NULL,  
    Gia_ban FLOAT,  
    Gia_KM FLOAT,  
    Meta_keyword NVARCHAR(128),  
    Meta_description NTEXT,  
    Mo_ta NTEXT  
)  
GO  
  
-- Tạo bảng Hóa Đơn  
CREATE TABLE HoaDon(  
    Ma_HD INT IDENTITY(1,1) PRIMARY KEY,  
    Ma_KH INT NOT NULL,  
    Ngay_mua_hang DATE NOT NULL,  
    Tong_tien FLOAT NOT NULL,  
    Trang_thai_don_hang SMALLINT,  
    Ten_nguoi_nhan NVARCHAR(100) NOT NULL,  
    Dia_chi_nguoi_nhan NVARCHAR(250) NOT NULL,  
    Email_nguoi_nhan VARCHAR(50) NOT NULL,  
    Phuong_thuc_thanh_toan SMALLINT NOT NULL,  
    Phuong_thuc_van_chuyen SMALLINT NOT NULL  
)  
GO  
  
-- Tạo bảng Chi Tiết Hóa Đơn  
CREATE TABLE ChiTietHoaDon(  
    So_TT INT IDENTITY(1,1) PRIMARY KEY,  
    Ma_HD INT NOT NULL,  
    Ma_SP INT NOT NULL,  
    Gia_ban FLOAT NOT NULL,  
    So_luong INT NOT NULL,  
    Trang_thai BIT,  
    Ngay_mua DATE  
)  
GO  
  
-- Tạo bảng Khách Hàng  
CREATE TABLE KhachHang(  
    Ma_KH INT, -- IDENTITY(1,1) PRIMARY KEY,  
    Ten_KH NVARCHAR(128) NOT NULL,  
    Email_KH VARCHAR(50) NOT NULL,  
    Mat_khau VARCHAR(50) NOT NULL,  
    Dien_thoai VARCHAR(11) NOT NULL,  
    Ngay_sinh DATE,  
    Gioi_tinh BIT,
```

```

        Ma_phuong_xa INT,
        Dia_chi_nha NVARCHAR(250),
        Trang_thai BIT,

    )
GO

-- Tạo bảng Tỉnh Thành
CREATE TABLE TinhThanh(
    Ma_tinh_thanh INT, -- IDENTITY(1,1) PRIMARY KEY,
    Trang_thai BIT,
    Ten_tinh_thanh NVARCHAR(50) NOT NULL
)
GO

-- Tạo bảng Phường Xã
CREATE TABLE PhuongXa(
    Ma_phuong_xa INT IDENTITY(1,1) PRIMARY KEY,
    Ma_quan_huyen INT NOT NULL,
    Ten_phuong_xa NVARCHAR(50) NOT NULL,
    Trang_thai BIT
)
GO

-- Tạo bảng Quận Huyện
CREATE TABLE QuanHuyen(
    Ma_quan_huyen INT, -- IDENTITY(1,1) PRIMARY KEY,
    Ma_tinh_thanh INT NOT NULL,
    Ten_quan_huyen NVARCHAR(50) NOT NULL,
    Trang_thai BIT
)
GO

-- ===== KẾT THÚC KHỞI TẠO CSDL =====

-- ##### Thêm dữ liệu vào bảng #####
-- Loại Sản Phẩm
INSERT INTO
loaiSanPham(Ten_loai,Trang_thai,Meta_keyword,Meta_description)VALUES
('VGA', 0, 'VGA ASUS', N'VGA hót nhất'),
('Mainboard', 1, 'Mainboard ASUS', N'MainBoard hót nhất'),
(N'Quạt tản nhiệt', 1, N'Quạt Cooler Master', N'Quạt tản nhiệt tốt nhất')

GO

-- Sản Phẩm
INSERT INTO
SanPham(Ma_loai,Ten_sp,Gia_ban,Gia_KM,Meta_keyword,Meta_description,Mo_ta
)VALUES
(1, 'Vga Asus 601', 232.00, 215.00, 'Asus 601', N'Vga Asus tốt nhất', N'Bảo hành 3 năm'),
(2, 'Mainboard Asus 901', 899.00, 800.00, 'Asus 901', N'Mainboard Asus tốt nhất', N'Bảo hành 1 năm'),
(2, 'Mainboard Asus 100', 990.00, 818.00, 'Asus 100', N'Mainboard Asus tốt nhất', N'Bảo hành 1 năm')
GO

-- Khách Hàng
INSERT INTO KhachHang(Ten_KH, Email_KH, Mat_khau, Dien_thoai, Ngay_sinh,
```

```
Gioi_tinh, Ma_phuong_xa, Dia_chi_nha, Trang_thai)VALUES
('Hoàng Anh Tú', 'anhtu@gmail', 'abc123', '0909879877', '1990-09-
09', 1, 3, '3/HQV quận Cầu Giấy HN', 0),
('Mai Hoa', 'maihoa@gmail', '123456', '0901239877', '1987-09-01', 0, 4, '8/TTT
quận Hoàn Kiếm HNN', 1),
('Nguyễn Lâm', 'Nlam@gmail', '654321', '0912379877', '1966-12-
12', 1, 5, '12/Nghĩa Tân quận Cầu Giấy HN', 1)
GO

-- Hóa Đơn
INSERT INTO HoaDon (Ma_KH, Ngay_mua_hang, Tong_tien, Trang_thai_don_hang,
Ten_nguoi_nhan, Dia_chi_nguoi_nhan,
Email_nguoi_nhan, Phuong_thuc_thanh_toan, Phuong_thuc_van_chuyen)VALUES
(1, '2016-03-10', 800.00, 10, N'Hoàng Anh Tú', N'3/HQV quận Cầu Giấy
HN', 'anhtu@gmail', 3, 1),
(2, '2016-03-09', 1200.00, 20, N'Mai Hoa', N'8/TTT quận Hoàn Kiếm
HN', 'maihoa@gmail', 4, 0),
(3, '2016-03-08', 989.00, 10, N'Nguyễn Lâm', N'12/Nghĩa Tân quận Cầu Giấy
HN', 'Nlam@gmail', 5, 1)
GO

-- Chi tiết hóa đơn
INSERT INTO ChiTietHoaDon (Ma_HD, Ma_SP, Gia_ban, So_luong, Trang_thai,
Ngay_mua )VALUES
(1, 1, 800.00, 3, 1, '2016-03-10'),
(1, 2, 800.00, 1, 1, '2016-03-10'),
(1, 3, 1600.00, 2, 1, '2016-03-10'),
(2, 1, 215.00, 1, 0, '2016-03-08'),
(3, 1, 215.00, 1, 0, '2016-03-08')
GO

-- Tỉnh Thành
INSERT INTO TinhThanh (Trang_thai , Ten_tinh_thanh) VALUES
(0, N'Hà Nội'),
(0, N'Hà Nam'),
(0, N'Vĩnh Phúc'),
(1, N'Tiền Giang'),
(1, N'Bến Tre')
GO

-- Quận Huyện
INSERT INTO QuanHuyen (Ma_tinh_thanh, Ten_quan_huyen, Trang_thai )VALUES
(1, N'Cầu Giấy', 0),
(1, N'Tây Hồ', 0),
(1, N'Hoàn Kiếm', 1),
(1, N'Ba Đình', 0)
GO

-- Phường Xã
INSERT INTO PhuongXa (Ma_quan_huyen , Ten_phuong_xa , Trang_thai )VALUES
(1, N'Dịch Vọng', 0),
(1, N'Quan Hoa', 0),
(1, N'Nghĩa Tân', 0),
(3, N'Cửa Nam', 1)
GO
```

### Phần III - Bài tập tự làm

#### Bài số 1: Sử dụng câu lệnh T-SQL thực hiện các yêu cầu sau

**Phần 1: Cấu trúc bảng dữ liệu**

<b>Hang_Hoa (Hàng hoá)</b>				
<b>Tên cột</b>	<b>Ý nghĩa</b>	<b>Kiểu</b>	<b>Kích thước</b>	<b>Yêu cầu</b>
<b><u>MaHH</u></b>	Mã hàng hoá	Char	4	Not Null
TenHH	Tên hàng hoá	Varchar	50	Not Null

<b>NVL (Nguyên vật liệu)</b>				
<b>Tên cột</b>	<b>Ý nghĩa</b>	<b>Kiểu</b>	<b>Kích thước</b>	<b>Yêu cầu</b>
<b><u>MaNVL</u></b>	Mã nguyên vật liệu	Char	4	Not Null
TenNVL	Tên nguyên vật liệu	Varchar	50	Not Null
DvTinh	Đơn vị tính	Char	10	
DonGia	Đơn giá	Float		$\geq 0$

<b>Dinh_Muc (Định mức nguyên vật liệu cho hàng hoá)</b>				
<b>Tên cột</b>	<b>Ý nghĩa</b>	<b>Kiểu</b>	<b>Kích thước</b>	<b>Yêu cầu</b>
<b><u>MaHH</u></b>	Mã hàng hoá	Char	4	Not Null
<b><u>MaNVL</u></b>	Mã NVL	Char	4	Not Null
SlDinhMuc	Số lượng định mức	Real		$\geq 0$

<b>KHSX (Kế hoạch sản xuất)</b>				
<b>Tên cột</b>	<b>Ý nghĩa</b>	<b>Kiểu</b>	<b>Kích thước</b>	<b>Yêu cầu</b>
<b><u>NamThang</u></b>	Tháng năm sản xuất	Char	6	Not Null
<b><u>MaHH</u></b>	Mã hàng hoá	Char	4	Not Null
SlSanXuat	Số lượng sản xuất	Int		$> 0$



**\* Các ràng buộc quan hệ**

Tên bảng	Tên cột	Tên bảng tham chiếu
Dinh_Muc	MaHH	Hang_Hoa
Dinh_Muc	MaNVL	NVL
KHSX	MaHH	Hang_Hoa

Và các yêu cầu mô tả trên.

**\* Dữ liệu mẫu (Không bắt buộc)**

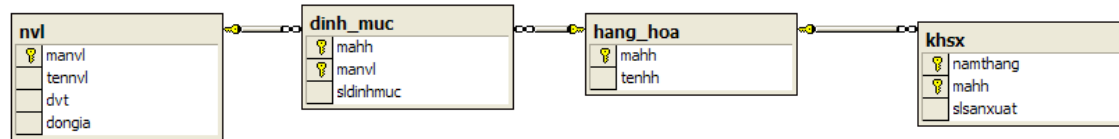
Bảng Hang_Hoa	
MaHH	TenHH
G001	Giày thể thao
G002	Giày thời trang
G003	Giày trẻ em

Bảng NVL			
MaNVL	TenNVL	Dvtvt	DonGia
D001	Da	Tám	100000
D002	Giả da	Mét	70000
D003	Nhựa	Kg	120000
D004	Keo	Kg	80000
D005	Chỉ	Mét	30000

Bảng Dinh_Muc		
MaHH	MaNVL	SiDinhMuc
G001	D001	0.3
G001	D002	0.2
G001	D003	0.3
G002	D001	0.2
G002	D002	0.1

G002	D003	0.2
G003	D002	0.1
G003	D003	0.1
G003	D004	0.3
G003	D005	0.3

Bảng KHSX		
NamThang	MaHH	SiSanXuat
200504	G001	8000
200504	G002	7500
200504	G003	7000
200505	G001	7500
200505	G002	8000



## Phần 2: Xây dựng các truy vấn sau

Liệt kê thông tin định mức các nguyên vật liệu cho hàng hoá có mã G001:

Mã nguyên vật liệu	Tên nguyên vật liệu	Đơn vị tính	Số lượng định mức
.....	.....	.....	.....

Thống kê theo mã nguyên vật liệu.

Ma NVL	Ten NVL	Tổng số hàng hoá
.....	.....	.....

Liệt kê số lượng dự toán của các nguyên vật liệu cần sử dụng cho hàng hoá có mã G001 theo kế hoạch sản xuất trong tháng 4 năm 2005.

Số lượng dự toán Theo mã NVL = Số lượng định mức Theo Mã NVL \* Số lượng sản xuất Theo hàng hoá

Ma NVL	Ten NVL	Số lượng dự toán
--------	---------	------------------

.....	.....	.....
-------	-------	-------

Tương tự câu c, thêm cột thành tiền dự toán.

Thành tiềnTheo mã NVL:=Số lượng dự toánTheo mã NVL \* Đơn giáTheo NVL

Ma NVL	Tên NVL	Số lượng dự toán	Thành tiền dự toán
.....	.....	.....	.....

### Phần 3: Xây dựng các view sau

Hướng dẫn: câu lệnh tạo view: CREATE VIEW tên\_view AS câu\_truy\_vấn

Liệt kê số lượng sản xuất của các hàng hoá có mã G001 theo kế hoạch sản xuất trong tháng 4 năm 2005.

Ma HH	Tên hàng hoá	Số lượng sản xuất
.....	.....	.....

Liệt kê các hàng hoá khác nhau đã được sản xuất trong năm 2005

Ma HH	Tên hàng hoá
.....	.....

Liệt kê các hàng hoá chỉ được sản xuất từ những NVL có đơn giá từ 80 000 trở lên

Ma HH	Tên hàng hoá	Tên NVL	Đơn giá
.....	.....	.....	

Liệt kê các hàng hoá được sản xuất từ những NVL có số lượng định mức là 0,3

Ma HH	Tên hàng hoá	Tên NVL	sldinhmuc
.....	.....	.....	

Liệt kê các hàng hoá có tên bắt đầu bằng chữ “Giày”, các chữ sau đó là ký tự bất kỳ

Ma HH	Tên hàng hoá
.....	.....

Liệt kê các tên hàng hoá có chứa dấu gạch dưới (\_)

Ma HH	Tên hàng hoá
.....	.....

Liệt kê các hàng hoá chưa từng có kế hoạch sản xuất

Ma HH	Tên hàng hoá
.....	.....

**Phần 4:** Sao lưu, thay đổi cấu trúc bảng

Thêm vào bảng `hang_hoa` cột `ghi_chu`, kiểu dữ liệu `varchar(50)`

Tạo 1 bảng sao của bảng `hang_hoa` có tên là `hang_hoa_backup` (lệnh `Select...into`)