

Chương 1

TỔNG QUAN CƠ SỞ DỮ LIỆU ACCESS

1. GIỚI THIỆU MICROSOFT ACCESS

Microsoft Access là một Hệ Quản Trị Cơ Sở Dữ Liệu (QTCSDDL) tương tác người sử dụng chạy trong môi trường Windows. Microsoft Access cho chúng ta một công cụ hiệu lực và đầy sức mạnh trong công tác tổ chức, tìm kiếm và biểu diễn thông tin.

Microsoft Access cho ta các khả năng thao tác dữ liệu, khả năng liên kết và công cụ truy vấn mạnh mẽ giúp quá trình tìm kiếm thông tin nhanh. Người sử dụng có thể chỉ dùng một truy vấn để làm việc với các dạng cơ sở dữ liệu khác nhau. Ngoài ra, có thể thay đổi truy vấn bất kỳ lúc nào và xem nhiều cách hiển thị dữ liệu khác nhau chỉ cần động tác nhấp chuột.

Microsoft Access và khả năng kết xuất dữ liệu cho phép người sử dụng thiết kế những biểu mẫu và báo cáo phức tạp đáp ứng đầy đủ các yêu cầu quản lý, có thể vận động dữ liệu và kết hợp các biểu mẫu và báo cáo trong một tài liệu và trình bày kết quả theo dạng thức chuyên nghiệp.

Microsoft Access là một công cụ đầy năng lực để nâng cao hiệu suất công việc. Bằng cách dùng các Wizard của MS Access và các lệnh có sẵn (macro) ta có thể dễ dàng tự động hóa công việc mà không cần lập trình. Đối với những nhu cầu quản lý cao, Access đưa ra ngôn ngữ lập trình Access Basic (Visual Basic For application) một ngôn ngữ lập trình mạnh trên CSDL.

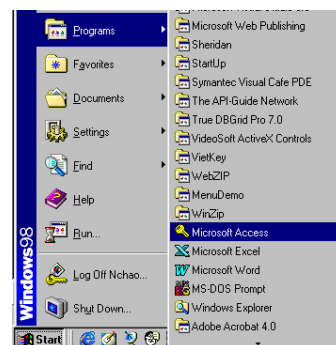
2. KHỞI ĐỘNG VÀ THOÁT KHỎI ACCESS

2.1. Khởi động ACCESS

Chọn nút Start trên thanh Task bar

Chọn Programs

Chọn Microsoft ACCESS



Khung hội thoại Microsoft ACCESS gồm:

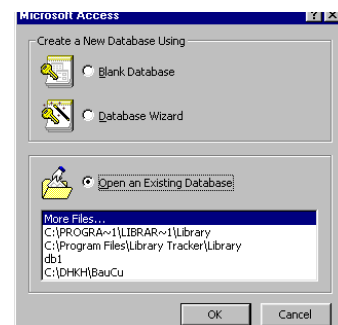
Create a New Database Using : Tạo CSDL ứng dụng mới.

Blank Database : Tạo CSDL trống.

Database Wizard : Tạo với sự trợ giúp

của Wizard.

Open an Existing Database : Mở một CSDL có sẵn.



2.2. Thoát khỏi ACCESS

Chọn File/Exit hoặc nhấn tổ hợp phím ALT+F4

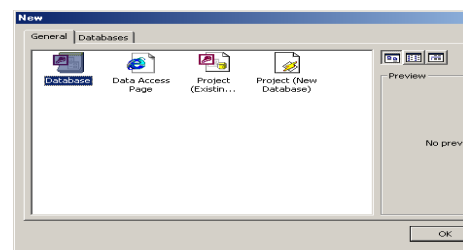
3. CÁC THAO TÁC TRÊN TẬP TIN CƠ SỞ DỮ LIỆU ACCESS

3.1. Tạo một tập tin CSDL

Thực hiện các thao tác sau:

Chọn File/New hoặc chọn biểu tượng

New trên thanh công cụ

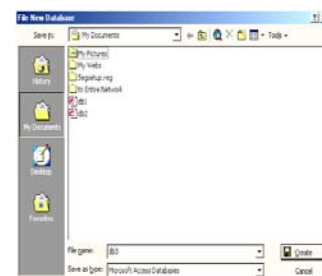


Chọn Database, chọn OK

Trong mục *Save in:* Chọn thư mục cần chứa tên tập tin.

File name: Chọn tên tập tin cần tạo

(Phần mở rộng mặc định là MDB)



3.2. Mở một CSDL đã tồn tại trên đĩa

Chọn File/Open database (Hoặc click biểu tượng Open)

Trong mục *Look in* : Chọn thư mục cần chứa tên tập tin cần mở.

File name: Chọn tên tập tin cần mở.

Chọn Open



3.3. Đóng một CSDL

Chọn File/Close hoặc ALT+F4

3.4. Các thành phần cơ bản của một tập tin CSDL ACCESS

Một tập tin CSDL ACCESS gồm có 6 thành phần cơ bản sau

Bảng (Tables) : Là nơi chứa dữ liệu

Truy vấn (Queries) : Truy vấn thông tin dựa trên một hoặc nhiều bảng.

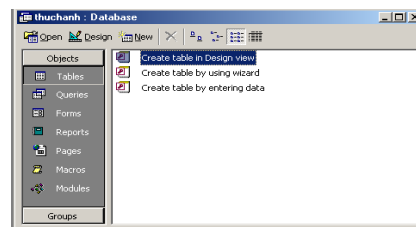
Biểu mẫu (Forms) : Các biểu mẫu dùng để nhập dữ liệu hoặc hiển thị dữ liệu.

Báo cáo (Reports) : Dùng để in ấn.

Pages (Trang) : Tạo trang dữ liệu.

Macros (Tập lệnh) : Thực hiện các tập lệnh.

Modules (Đơn thể) : Dùng để lập trình Access Basic



4. CÁCH SỬ DỤNG CỬA SỔ DATABASE

Như đã nói ở trên, một CSDL của Access chứa trong nó 7 đối tượng chứ không đơn thuần là bảng dữ liệu. Sau khi tạo mới một CSDL hoặc mở một CSDL có sẵn Access sẽ hiển thị một cửa sổ Database, trên đó hiển thị tên của CSDL đang mở và liệt kê 7 đối tượng mà nó quản lý, mỗi lớp đối tượng đều được phân lớp rõ ràng để tiện theo dõi.

4.1. Tạo một đối tượng mới

Trong cửa sổ Database, chọn tab chứa đối tượng cần tạo (Bảng, Truy vấn, Biểu mẫu, Báo cáo,...) hoặc thực hiện lệnh

View/Database Object - Table/Query/Form/Report/Pages/Macros/Modules

Chọn nút New.

4.2. Thực hiện một đối tượng trong CSDL

Trong cửa sổ Database, chọn tab cần thực hiện. Cửa sổ Database liệt kê tên các đối tượng có sẵn, chọn tên đối tượng cần mở.

Chọn nút Open (đối với Bảng, Truy vấn, Biểu mẫu, Trang) hoặc Preview (đối với Báo biểu) hoặc Run (đối với Macro và Module).

4.3. Sửa đổi một đối tượng có sẵn trong CSDL

Trong cửa sổ Database, chọn tab cần thực hiện. Cửa sổ Database liệt kê tên các đối tượng có sẵn, chọn tên đối tượng cần mở, Chọn nút Design.

Chương 2

BẢNG DỮ LIỆU

Bảng là đối tượng chủ yếu chứa các thông tin cần quản lý, có thể đó chỉ là một vài địa chỉ đơn giản hay cả vài chục nghìn bản ghi chứa đựng thông tin liên quan đến các hoạt động SXKD của một công ty xuất nhập khẩu nào đó. Trước khi ta muốn làm việc với bất kỳ một CSDL nào thì ta phải có thông tin để quản lý, các thông tin đó nằm trong các bảng, nó là cơ sở để cho người sử dụng tạo các đối tượng khác trong CSDL như truy vấn, biểu mẫu, báo biểu...

1. THIẾT KẾ CƠ SỞ DỮ LIỆU

Một CSDL được thiết kế tốt cho phép người sử dụng truy cập nhanh chóng đến những thông tin cần tham khảo, giúp tiết kiệm được thời gian truy xuất thông tin. Một CSDL thiết kế tốt giúp người sử dụng rút ra được những kết quả nhanh chóng và chính xác hơn.

Để thiết kế một CSDL tốt chúng ta phải hiểu cách mà một Hệ QTCSDL quản trị các CSDL như thế nào. MS Access hay bất kỳ một Hệ QTCSDL nào có thể cung cấp các thông tin cho chúng ta một cách chính xác và hiệu quả nếu chúng được cung cấp đầy đủ mọi dữ kiện về nhiều đối tượng khác nhau lưu trữ trong các bảng dữ liệu. Ví dụ ta cần một bảng để chứa thông tin về lý lịch của cán bộ, một bảng khác để chứa các đề tài nguyên cứu khoa học của các cán bộ...

Khi bắt tay thiết kế CSDL, chúng ta phải xác định và phân tích các thông tin muốn lưu trữ thành các đối tượng riêng rẽ, sau đó báo cho Hệ QTCSDL biết các đối tượng đó liên quan với nhau như thế nào. Dựa vào các quan hệ đó mà Hệ QTCSDL có thể liên kết các đối tượng và rút ra các số liệu tổng hợp cần thiết.

CÁC BƯỚC THIẾT KẾ CSDL

Bước 1: Xác định mục tiêu khai thác CSDL của chúng ta. Điều này quyết định các loại sự kiện chúng ta sẽ đưa vào MS Access.

Bước 2: Xác định các bảng dữ liệu cần thiết. Mỗi đối tượng thông tin sẽ hình thành một bảng trong CSDL của chúng ta.

Bước 3: Sau khi đã xác định xong các bảng cần thiết, tiếp đến ta phải chỉ rõ thông tin nào cần quản lý trong mỗi bảng, đó là xác định các trường. Mỗi loại thông tin trong bảng gọi là trường. Mọi mẫu in trong cùng một bảng đều có chung cấu trúc các trường. Ví dụ: Trong lý lịch khoa học cán bộ, những trường (thông tin) cần quản lý là: “HỌ VÀ TÊN”, “CHUYÊN MÔN”, “HỌC VỊ”, “HỌC HÀM”,...

Bước 4: Xác định các mối quan hệ giữa các bảng. Nhìn vào mỗi bảng dữ liệu và xem xét dữ liệu trong bảng này liên hệ thế nào với dữ liệu trong bảng khác. Thêm trường hoặc tạo bảng mới để làm rõ mối quan hệ này. Đây là vấn đề hết sức quan trọng, tạo được quan hệ tốt sẽ giúp chúng ta nhanh chóng truy tìm và kết xuất dữ liệu.

Bước 5: Tinh chế, hiệu chỉnh lại thiết kế. Phân tích lại thiết kế ban đầu để tìm lỗi, tạo bảng dữ liệu và nhập vào vài bản ghi, thử xem CSDL đó phản ánh thế nào với những yêu cầu truy xuất của chúng ta, có rút được kết quả đúng từ những bảng dữ liệu đó không. Thực hiện các chỉnh sửa thiết kế nếu thấy cần thiết.

2. KHÁI NIỆM VỀ BẢNG

Bảng là nơi chứa dữ liệu về một đối tượng thông tin nào đó như SINH VIÊN, HÓA ĐƠN,... Mỗi hàng trong bảng gọi là một **bản ghi** (record) chứa các nội dung riêng của đối tượng đó. Mỗi bản ghi của một bảng đều có chung cấu trúc, tức là các **trường** (field). Ví dụ: Cho bảng dưới đây để quản lý lý lịch khoa học cán bộ trong trường đại học, có các trường MACB (Mã cán bộ), TRINHDOVH (Trình độ văn hóa), CHUYENMON (Chuyên môn),...

Trong một CSDL có thể chứa nhiều bảng, thường mỗi bảng lưu trữ nhiều thông tin (dữ liệu) về một đối tượng thông tin nào đó, mỗi một thông tin đều có những kiểu đặc trưng riêng, mà với Access nó sẽ cụ thể thành những kiểu dữ liệu của các trường.

3. TẠO BẢNG MỚI TRONG CƠ SỞ DỮ LIỆU

Trong MS Access có hai cách để tạo bảng, một là cách dùng Table Wizard, nhưng các trường ở đây MS Access tự động đặt tên và không có bàn tay can thiệp của người sử dụng. Ở đây, sẽ đưa ra cách tạo mới bảng hoàn toàn do người sử dụng.

3.1. Tạo bảng không dùng Table Wizard

Trong cửa sổ Database, chọn tab Table (hoặc Lệnh *View/Daatabase object - Table*)

Chọn nút **New**, xuất hiện hộp thoại

Datasheet View: Trên màn hình sẽ xuất hiện một bảng trống với các trường (tiêu đề cột) lần lượt Field1, field2

Design View: Trên màn hình xuất hiện cửa sổ thiết kế bảng, người sử dụng tự thiết kế bảng.

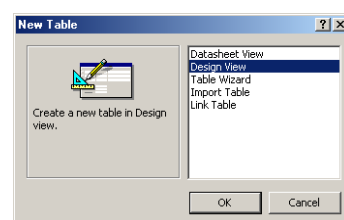


Table Wizard: Thiết kế bảng với sự trợ giúp của MS Access

Import table: Nhập các bảng và các đối tượng từ các tập tin khác vào CSDL hiện thời.

Link table: Tạo bảng bằng cách nối vào CSDL hiện thời các bảng của CSDL khác.

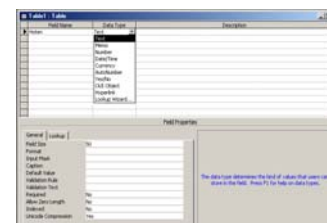
Chọn chức năng Design View, chọn OK.

3.2. Sử dụng Design View

Field Name: Tên trường cần đặt (thông tin cần quản lý)

Data Type: Kiểu dữ liệu của trường

Description: Mô tả trường, phần này chỉ mang ý nghĩa làm rõ thông tin quản lý, có thể bỏ qua trong khi thiết kế bảng.



Field properties: Các thuộc tính của trường

Xác định khoá chính của bảng (nếu có)

Xác định thuộc tính của bảng, Lưu bảng dữ liệu

Đặt tên trường

Tên trường ở đây không nhất thiết phải có độ dài hạn chế và phải sát nhau, mà ta có thể đặt tên trường tùy ý nhưng không vượt quá 64 ký tự kể cả ký tự trắng. Lưu ý rằng, tên trường có thể đặt dài nên nó dễ mô tả được thông tin quản lý, nhưng sẽ khó khăn hơn khi ta dùng các phát biểu SQL và lập trình Access Basic. Do đó khi đặt tên trường ta nên đặt ngắn gọn, dễ gọi nhớ và không chứa ký tự trắng.

Kiểu dữ liệu

MS Access cung cấp một số kiểu dữ liệu cơ bản sau:

| Kiểu dữ liệu | Dữ liệu vào | Kích thước |
|---------------|--|---|
| Text | Văn bản | Tối đa 255 byte |
| Memo | Văn bản nhiều dòng, trang | Tối đa 64000 bytes |
| Number | Số | 1,2,4 hoặc 8 byte |
| Date/Time | Ngày giờ | 8 byte |
| Currency | Tiền tệ (Số) | 8 byte |
| Auto number | ACCESS tự động tăng lên một khi một bản ghi được tạo | 4 byte |
| Yes/No | Lý luận (Boolean) | 1 bit |
| OLE Object | Đối tượng của phần mềm khác | Tối đa 1 giga byte |
| Lookup Wizard | | Trường nhận giá trị do người dùng chọn từ 1 bảng khác hoặc 1 danh sách giá trị định trước |
| Hyper link | Liên kết các URL | |

Quy định thuộc tính, định dạng cho trường

Đặt thuộc tính là một phần không kém quan trọng, nó quyết định đến dữ liệu thực sự lưu giữa trong bảng, kiểm tra độ chính xác dữ liệu khi nhập vào, định dạng

dữ liệu nhập vào ... Mỗi một kiểu dữ liệu sẽ có các thuộc tính và các đặc trưng và khác nhau. Sau đây là các thuộc tính, định dạng của các kiểu dữ liệu.

Để tăng thêm tốc độ xử lý khi nhập dữ liệu cũng như các công việc tìm kiếm sau này thì việc quy định dữ liệu rất quan trọng.

Các trường trong ACCESS có các thuộc tính sau:

3.2.1. Field Size

Quy định kích thước của trường và tùy thuộc vào từng kiểu dữ liệu

Kiểu Text: Chúng ta quy định độ dài tối đa của chuỗi.

Kiểu Number: Có thể chọn một trong các loại sau:

Byte: 0..255

Integer: -32768..32767

Long Integer: -3147483648.. 3147483647

Single: $-3,4 \times 10^{38}$.. $3,4 \times 10^{38}$ (Tối đa 7 số lẻ)

Double: -1.797×10^{308} .. 1.797×10^{308} (Tối đa 15 số lẻ)

Decimal Places

Quy định số chữ số thập phân (Chỉ sử dụng trong kiểu Single và Double)

Đối với kiểu Currency mặc định decimal places là 2

3.2.2. Format

Quy định dạng hiển thị dữ liệu, tùy thuộc vào từng kiểu dữ liệu.

Kiểu chuỗi: Gồm 3 phần

<Phần 1>;<Phần 2>;<Phần 3>

Trong đó:

<Phần 1>: Chuỗi định dạng tương ứng trong trường hợp có chứa văn bản.

<Phần 2>: Chuỗi định dạng tương ứng trong trường hợp không chứa văn bản.

<Phần 3>: Chuỗi định dạng tương ứng trong trường hợp null

Các ký tự dùng để định dạng chuỗi

| Ký tự | Tác dẽch vuũ du lẽchũng |
|--|---|
| @ | Chuỗi ký tự |
| > | Đổi tất cả ký tự nhập vào thành in hoa |
| < | Đổi tất cả ký tự nhập vào thành in thường |
| “Chuỗi ký tự “ | Chuỗi ký tự giữa 2 dấu nháy |
| \<ký tự> | Ký tự nằm sau dấu \ |
| [black] [White] [red] Hoặc [<số>] Trong đó 0<=số<=56 | Màu |

Ví dụ

| Cách định dạng | Dữ liệu | Hiển thị |
|---------------------------|--|--|
| @@@-@@@@ | 123456 abcdef | 123-456 abc-def |
| > | Tinhoc | TINHOC |
| < | TINHOC | Tinhoc |
| @;”Không có”;”Không biết” | Chuỗi bất kỳ Chuỗi rỗng Giá trị trống (Null) | Hiển thị chuỗi Không có Không biết |

Kiểu Number

Định dạng do ACCESS cung cấp

| Dạng | Dữ liệu | Hiển thị |
|----------------|---------|------------|
| General Number | 1234.5 | 1234.5 |
| Currency | 1234.5 | \$1.234.50 |
| Fixed | 1234.5 | 1234 |

| | | |
|------------|--------|----------|
| Standard | 1234.5 | 1,234.50 |
| Pecent | 0.825 | 82.50% |
| Scientific | 1234.5 | 1.23E+03 |

Định dạng do người sử dụng

<Phần 1>;<Phần 2>;<Phần 3>;<Phần 4>

<Phần 1>: Chuỗi định dạng tương ứng trong trường hợp số dương.

<Phần 2>: Chuỗi định dạng tương ứng trong trường hợp số âm.

<Phần 3>: Chuỗi định dạng tương ứng trong trường hợp số bằng zero.

<Phần 4>: Chuỗi định dạng tương ứng trong trường hợp null.

Các ký tự định dạng

| Ký tự | Tác dụng |
|-----------|----------------------------|
| .(Period) | Dấu chấm thập phân |
| ,(comma) | Dấu phân cách ngàn |
| 0 | Ký tự số (0-9) |
| # | Ký tự số hoặc khoảng trắng |
| \$ | Dấu \$ |
| % | Phần trăm |

Ví dụ

| Định dạng | Hiển thị |
|---------------|--|
| 0;(0);;"Null" | Số dương hiển thị bình thường Số âm được bao giữa 2 dấu ngoặc Số zero bị bỏ trống Null hiện chữ Null |
| +0.0;-0.0;0.0 | Hiển thị dấu + phía trước nếu số dương Hiển thị dấu - phía trước nếu số âm Hiển thị 0.0 nếu âm hoặc Null |

Kiểu Date/Time

Các kiểu định dạng do ACCESS cung cấp

| Dạng | Hiển thị |
|--------------|-----------------------|
| General date | 10/30/99 5:10:30PM |
| Long date | Friday, may 30 , 1999 |
| Medium date | 30-jul-1999 |
| Short date | 01/08/99 |
| Long time | 6:20:00 PM |
| Medium time | 6:20 PM |
| Short time | 18:20 |

Các ký tự định dạng

| Ký tự | Tác dụng |
|-------|----------------------------|
| : | Dấu phân cách giờ |
| / | Dấu phân cách ngày |
| d | Ngày trong tháng (1-31) |
| dd | Ngày trong tháng 01-31) |
| ddd | Ngày trong tuần (Sun -Sat0 |
| W | Ngày trong tuần (1-7) |
| WW | Tuần trong năm (1-54) |
| M | Tháng trong năm (1-12) |
| MM | Tháng trong năm (01-12) |
| q | Quý trong năm (1-4) |
| y | Ngày trong năm (1-366) |
| yy | Năm (01-99) |
| h | Giờ (0-23) |
| n | Phút (0-59) |
| s | Giây (0-59) |

Ví dụ

| Định dạng | Hiển thị |
|------------------|-----------------|
| Ddd,"mmm d",yyyy | Mon,jun 2, 1998 |
| Mm/dd/yyyy | 01/02/1998 |

Kiểu Yes/No

Các kiểu định dạng

| Định dạng | Tác dụng |
|------------|----------|
| Yes/No | Đúng/Sai |
| True/False | Đúng/Sai |
| On/Off | Đúng/Sai |

Định dạng do người sử dụng: Gồm 3 phần

<Phần 1>;<Phần 2>;<Phần 3>

Trong đó: <Phần 1>: Bỏ trống

<Phần 2>: Trường hợp giá trị trường đúng

<Phần 3>: Trường hợp giá trị trường sai

Ví dụ

| Định dạng | Hiển thị | |
|---------------|-----------------|------------------|
| | Trường hợp True | Trường hợp False |
| ;"Nam";"Nu" | Nam | Nu |
| ;"co";"Khong" | Co | Khong |

3.2.3. Input mask (Mặt nạ)

Thuộc tính này dùng để quy định mặt nạ nhập dữ liệu cho một trường.

Các ký tự định dạng trong input mask

| Ký tự | Tác dụng |
|----------|---|
| 0 | Bắt buộc nhập ký tự số |
| 9 | Không bắt buộc nhập, ký tự số |
| # | Không bắt buộc nhập, số 0-9, khoảng trắng, dấu + và - |
| L | Bắt buộc nhập, ký tự chữ |
| ? | Không bắt buộc nhập, ký tự chữ hoặc khoảng trắng |
| a | Bắt buộc nhập, ký tự chữ hoặc số |
| A | Không bắt buộc nhập, ký tự chữ hoặc số |
| & | Bắt buộc nhập, ký tự bất kỳ |
| C | Không bắt buộc nhập ký tự bất kỳ |
| < | Các ký tự bên phải được đổi thành chữ thường |
| > | Các ký tự bên phải được đổi thành chữ hoa |
| ! | Dữ liệu được ghi từ phải sang trái |
| \<Ký tự> | Ký tự theo sau \ sẽ được đưa thẳng vào |

Ví dụ

| Input mask | Dữ liệu nhập vào |
|---------------|------------------|
| (000)000-0000 | (054)828-8282 |
| (000)AAA-A | (123)124-E |

☞ **Chú ý:** Nếu muốn các ký tự gõ vào quy định thuộc tính input mask là password (Khi nhập dữ liệu vào tại các vị trí đó xuất hiện dấu *).

3.2.4. Caption

Quy định nhãn là một chuỗi ký tự sẽ xuất hiện tại dòng tiêu đề của bảng. Chuỗi ký tự này cũng xuất hiện tại nhãn các của các điều khiển trong các biểu mẫu hoặc báo cáo.

3.2.5. Default value

Quy định giá trị mặc định cho trường trừ Auto number và OEL Object

3.2.6. Validation rule và Validation Text

Quy định quy tắc hợp lệ dữ liệu (Validation rule) để giới hạn giá trị nhập vào cho một trường. Khi giới hạn này bị vi phạm sẽ có câu thông báo ở Validation text.

Các phép toán có thể dùng trong Validation rule

| Các phép toán | Phép toán | Tác dụng |
|--------------------|---------------------|--------------------|
| Phép so sánh | >, <, >=, <=, =, <> | |
| Phép toán logic | Or, and, not | Hoặc, và, phủ định |
| Phép toán về chuỗi | Like | Giống như |

☞ **Chú ý:** Nếu hằng trong biểu thức là kiểu ngày thì nên đặt giữa 2 dấu #.

Ví dụ

| Validation rule | Tác dụng |
|-------------------------------|---------------------------------------|
| <>0 | Khác số không |
| Like “*HUE*” | Trong chuỗi phải chứa HUE |
| <#25/07/76# | Trước ngày 25/07/76 |
| >=#10/10/77# and <=#12/11/77# | Trong khoảng từ 10/10/77 đến 12/12/77 |

3.2.7. Required

Có thể quy định thuộc tính này để bắt buộc hay không bắt buộc nhập dữ liệu cho trường.

| Required | Tác dụng |
|----------|-----------------------------|
| Yes | Bắt buộc nhập dữ liệu |
| No | Không bắt buộc nhập dữ liệu |

3.2.8. AllowZeroLength

Thuộc tính này cho phép quy định một trường có kiểu Text hay memo có thể hoặc không có thể có chuỗi có độ dài bằng 0.

☞ **Chú ý:** Cần phân biệt một trường chứa giá trị null (chưa có dữ liệu) và một trường chứa chuỗi có độ dài bằng 0 (Có dữ liệu nhưng chuỗi rỗng “”).

| AllowZeroLength | Tác dụng |
|-----------------|----------------------------|
| Yes | Chấp nhận chuỗi rỗng |
| No | Không chấp nhận chuỗi rỗng |

3.2.9. Index

Quy định thuộc tính này để tạo chỉ mục trên một trường. Nếu chúng ta lập chỉ mục thì việc tìm kiếm dữ liệu nhanh hơn và tiện hơn.

| Index | Tác dụng |
|--------------------|-----------------------------|
| Yes(Duplicate OK) | Tạo chỉ mục có trùng lặp |
| Yes(No Duplicate) | Tạo chỉ mục không trùng lặp |
| No | Không tạo chỉ mục |

3.2.10. New value

Thuộc tính này chỉ đối với dữ liệu kiểu auto number, quy định cách thức mà trường tự động điền số khi thêm bản ghi mới vào.

| New value | Tác dụng |
|-----------|-------------------|
| Increase | Tăng dần |
| Random | Lấy số ngẫu nhiên |

4. THIẾT LẬP KHOÁ CHÍNH (*primary key*)

4.1. Khái niệm khoá chính

Sức mạnh của một Hệ QTCSDL như Microsoft Access, là khả năng mau chóng truy tìm và rút dữ liệu từ nhiều bảng khác nhau trong CSDL. Để hệ thống có thể làm được điều này một cách hiệu quả, mỗi bảng trong CSDL cần có một trường hoặc một nhóm các trường có thể xác định duy nhất một bản ghi trong số rất nhiều bản ghi đang có trong bảng. Đây thường là một mã nhận diện như Mã nhân viên hay Số Báo Danh của học sinh. Theo thuật ngữ CSDL trường này được gọi là **khóa chính** (*primary key*) của bảng. MS Access dùng trường khóa chính để kết nối dữ liệu nhanh chóng từ nhiều bảng và xuất ra kết quả yêu cầu.

Nếu trong bảng chúng ta đã có một trường sao cho ứng với mỗi trị thuộc trường đó chúng ta xác định duy nhất một bản ghi của bảng, chúng ta có thể dùng trường đó làm trường khóa của bảng. Từ đó cho ta thấy rằng tất cả các trị trong trường khóa chính phải khác nhau. Chẳng hạn đừng dùng tên người làm trường khóa vì tên trường là không duy nhất.

Nếu không tìm được mã nhận diện cho bảng nào đó, chúng ta có thể dùng một trường kiểu **Autonumber** (ví dụ Số Thứ Tự) để làm trường khóa chính.

Khi chọn trường làm khóa chính chúng ta lưu ý mấy điểm sau:

MS Access không chấp nhận các giá trị trùng nhau hay trống (null) trong trường khóa chính.

Chúng ta sẽ dùng các giá trị trong trường khóa chính để truy xuất các bản ghi trong CSDL, do đó các giá trị trong trường này không nên quá dài vì khó nhớ và khó gõ vào.


Kích thước của khóa chính ảnh hưởng đến tốc độ truy xuất CSDL. Để đạt hiệu quả tối ưu, dùng kích thước nhỏ nhất để xác định mọi giá trị cần đưa vào trường.

4.2. Cách đặt khoá chính

Ta có thể tự chọn trường làm khóa chính cho bảng bằng các bước sau đây:

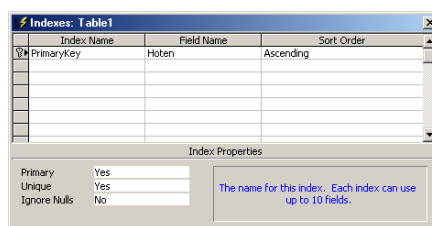
Mở bảng ở chế độ Design View

Nhấp chọn trường cần đặt

Thực hiện lệnh *Edit - Primary Key* hoặc nhấp chọn nút  trên thanh công cụ của mục này .

☞ **chú ý:** Không phải mọi trường đều có thể làm khóa chính, mà chỉ có các trường có các kiểu dữ liệu không phải là **Memo** và **OLE Object.**, **Hyper Link**.

Để hủy bỏ khóa chính hoặc các đã thiết lập thì thực hiện lệnh *View - Indexes*, trong hộp thoại này chọn và xóa đi những trường khóa đã thiết lập:



5. LƯU BẢNG

Sau khi thiết kế xong, ta tiến hành lưu bảng vào CSDL, có thể thực hiện một trong hai thao tác sau:

Thực hiện lệnh *File - Save*.

Nhấp chọn nút  trên thanh công cụ của mục này (Table Design)

6. HIỆU CHỈNH BẢNG

6.1. Di chuyển trường: Các thao tác để di chuyển thứ tự các trường

Đưa con trỏ ra đầu trường đến khi con trỏ chuột chuyển thành ➡ thì nhấp chọn.

Đưa con trỏ ra đầu trường vừa chọn, nhấn và kéo đến vị trí mới.

6.2. Chèn trường : Các thao tác lần lượt như sau

Chọn trường hiện thời là trường sẽ nằm sau trường được chèn vào

Thực hiện lệnh *Insert/ Row*

6.3. Xóa trường: Các thao tác lần lượt như sau

Chọn trường cần xóa

Thực hiện lệnh *Edit - Delete Rows*

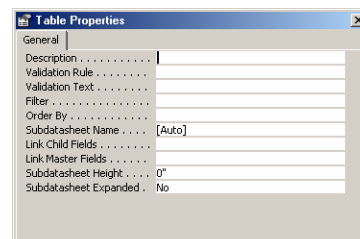
6.4. Quy định thuộc tính của bảng

Mở bảng ở chế độ Design View, chọn View/Properties

Description: Dòng mô tả bảng

Validation Rule: Quy tắc hợp lệ dữ liệu cho toàn bảng.

Validation Text: Thông báo lỗi khi dữ liệu không hợp lệ.





7. XEM THÔNG TIN VÀ BỔ SUNG BẢN GHI

7.1. Xem thông tin ở chế độ datasheet

Muốn xem thông tin trong một bảng chúng ta phải chuyển bảng sang một chế độ hiển thị khác gọi là *Datasheet*. Trong chế độ hiển thị này, mỗi bản ghi được thể hiện trên một hàng ngang, hàng đầu tiên là các tên trường.

Sau đây là các cách để chuyển sang chế độ hiển thị Datasheet:

Trong cửa sổ Database của CSDL đang mở, nhấp chọn tab Table. Trong mục này chọn bảng cần hiển thị rồi chọn nút **Open**, bảng sẽ được mở để bổ sung và chỉnh sửa dữ liệu.

Ta có thể chuyển sang chế độ Datasheet ngay khi đang ở trong chế Design, bằng cách nhấp chọn nút  thì bảng sẽ chuyển sang chế độ Datasheet, để quay trở về chế độ Design, ta nhấp chọn lại nút . Hoặc chọn lệnh *View - /Design View*.

7.2. Bổ sung bản ghi cho bảng

Sau khi hoàn thành công việc thiết kế cấu trúc bảng, ta tiến hành nhập dữ liệu, tức là bổ sung các bản ghi, cho bảng. Hiển thị bảng ở chế độ hiển thị Datasheet, mỗi hàng đại diện cho một bản ghi. Ở đây có các ký hiệu chúng ta cần biết công dụng của chúng



Bản ghi hiện thời




Bản ghi đang nhập dữ liệu



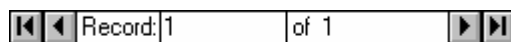
Bản ghi mới

7.2.1. Bổ sung bản ghi mới cho CSDL


Đang đứng tại một bản ghi nào đó (không phải là bản ghi mới), chọn nút  trên thanh công cụ. Hoặc thực hiện lệnh *Record - Go To - New*.

7.2.2. Di chuyển giữa các bản ghi

Ta có thể di chuyển qua lại giữa các bản ghi bằng cách dùng công cụ *Chọn lựa bản ghi* (Record Selector).



7.2.3. Nhập dữ liệu cho bản ghi

Khi đang nhập dữ liệu cho một bản ghi nào đó, thì đầu hàng của bản ghi đó xuất hiện biểu tượng .

| Tổ hợp phím | Tác dụng |
|-------------|-----------------------|
| Tab | Sang ô kế tiếp |
| Shift Tab | Sang ô phía trước |
| Home | Đến đầu dòng |
| End | Đến cuối dòng |
| Ctrl Home | Đến bản ghi đầu tiên |
| Ctrl End | Đến bản ghi cuối cùng |
| Shift F2 | Zoom |

Theo chuẩn, khi nhập dữ liệu thì Access sẽ lấy font mặc định là MS San Serif, điều này có thể giúp cho ta hiển thị được tiếng Việt chỉ khi MS San Serif đó là của VietWare hoặc của ABC.

Để không phụ thuộc vào điều này, ta nên chọn font trước khi tiến hành nhập dữ liệu. Trong chế độ hiển thị Datasheet, thực hiện lệnh *Format - Font...*

Khi nhập dữ liệu là trường cho trường **OLE Object**, ta thực hiện như sau: Lệnh *Edit - Object...*

7.2.4. Chọn các bản ghi

Đánh dấu chọn bản ghi:

Chọn Edit/Select Record: Để chọn bản ghi hiện hành

Chọn Edit/ Select all Record để chọn toàn bộ

7.2.5. Xóa bản ghi

Chọn các bản ghi cần xóa, sau đó thực hiện lệnh *Edit - Delete* (hoặc nhấn phím DELETE) .

8. THIẾT LẬP QUAN HỆ GIỮA CÁC BẢNG

8.1. Các loại quan hệ trong cơ sở dữ liệu ACCESS

8.1.1. Quan hệ một - một (1-1)

Trong quan hệ một - một, mỗi bản ghi trong bảng A có tương ứng với một bản ghi trong bảng B và ngược lại mỗi bản ghi trong bảng B có tương ứng duy nhất một bản ghi trong bảng A.

Ví dụ: Cho 2 bảng dữ liệu

Bảng **Danh sach**(**Masv**, ten, Ngaysinh, gioitinh) và bảng **Diem thi**(**Masv**, diem)

| Ten | Ngaysinh | Gioitinh | Masv | | Masv | diem |
|------|----------|----------|------|---|------|------|
| An | 20/10/77 | Yes | A001 | ↔ | A001 | 9 |
| Bình | 21/07/80 | No | A002 | ↔ | A002 | 7 |
| Thủy | 02/12/77 | Yes | A003 | ↔ | A003 | 9 |
| Lan | 03/04/80 | No | A004 | ↔ | A004 | 4 |
| Hồng | 12/11/77 | No | A005 | ↔ | A005 | 5 |

Bảng **Danh sach** và **diem thi** có mối quan hệ **1-1** dựa trên trường **Masv**.

8.1.2. Quan hệ một nhiều (1-∞)

Là mối quan hệ phổ biến nhất trong CSDL, trong quan hệ một nhiều : Một bản ghi trong bảng A sẽ có thể có nhiều bản ghi tương ứng trong bảng B, nhưng ngược lại một bản ghi trong bảng B có duy nhất một bản ghi tương ứng trong bảng A.

Ví dụ: Trong một khoa của một trường học nào đó có nhiều sinh viên, nhưng một sinh viên thuộc một khoa nhất định. Ta có 2 bảng dữ liệu như sau:

Bảng Danhsachkhoa(**Makhoa**, tenkhoa, sodthoi)

Bảng danhsachsv(Makhoa, Ten, Quequan, lop)

| Tenkhoa | Sodthoi | Makhoa | | Makhoa | Ten | Quequan | Lop |
|---------|---------|--------|---|--------|-------|---------|-----|
| CNTT | 826767 | 01 | → | 01 | Thanh | Huế | K23 |
| TOÁN | 878787 | 02 | → | 01 | Tùng | Qbình | K24 |
| LÝ | 868785 | 03 | → | 02 | Thủy | Huế | K25 |
| | | | → | 02 | Hùng | ĐN | K26 |
| | | | → | 03 | Lan | Huế | K25 |
| | | | → | 03 | Hương | ĐN | K26 |

Bảng Danhsachkhoa và bảng danhsachsv có mối quan hệ 1-∞ dựa trên trường Makhoa.

8.3. Quan hệ nhiều nhiều(∞-∞)

Trong quan hệ nhiều nhiều, mỗi bản ghi trong bảng A có thể có không hoặc nhiều bản ghi trong bảng B và ngược lại mỗi bản ghi trong bảng B có thể có không hoặc nhiều bản ghi trong bảng A.

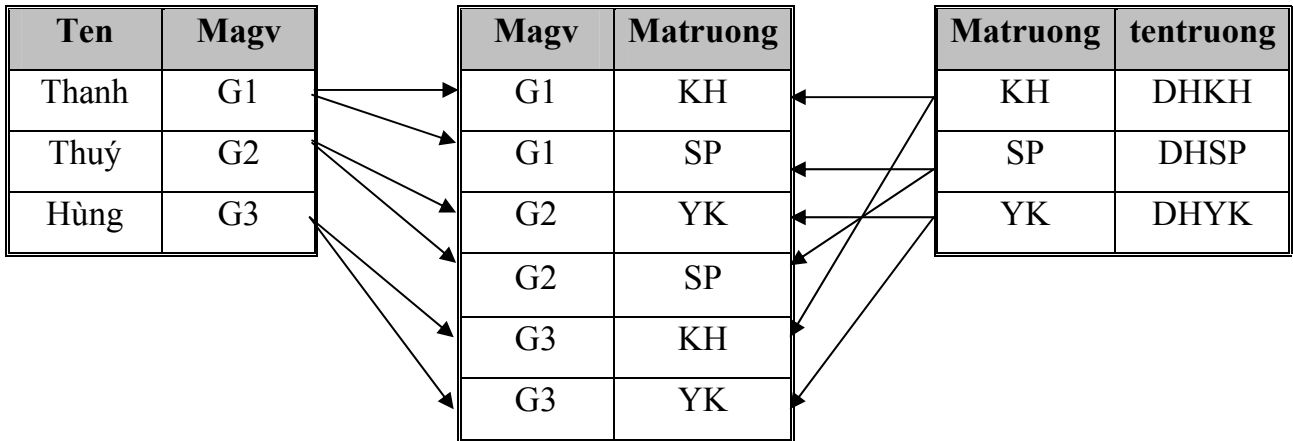
Khi gặp mối quan hệ nhiều- nhiều để không gây nên sự trùng lặp và dư thừa dữ liệu thì người ta tách quan hệ nhiều-nhiều thành 2 quan hệ một-nhiều bằng cách tạo ra một bảng phụ chứa khóa chính của 2 bảng đó.

Ví dụ: Một giáo viên có thể dạy cho nhiều trường và một trường có nhiều giáo viên tham gia giảng dạy. Đây là một mối quan hệ nhiều-nhiều

Bảng Danhsachgv(**Magv**,ten)

Bảng Danhsachtruong(**Matruong**, Tentruong)

Tạo ra bảng Phancongday(**Magv**, **matruong**)



Bảng **Danhsachgv** và bảng **Phancongday** có mối quan hệ **1-∞** dựa trên trường **Magv**.

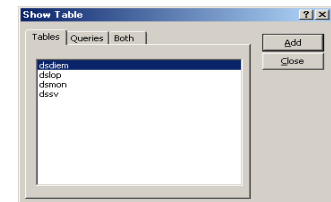
Bảng **Danhsachtruong** và bảng **Phancongday** có mối quan hệ **1-∞** dựa trên trường **Matruong**.

8.4. Thiết lập mối quan hệ giữa các bảng dữ liệu (Relationships)

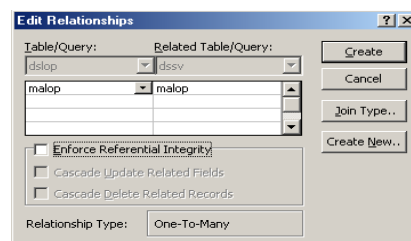
Tại cửa sổ Database, thực hiện lệnh Tools/Relationship



Trong cửa sổ Show bảng cần thiết lập quan hệ, sau đó chọn **Add** và **Close**.



Kéo trường liên kết của bảng quan hệ vào trường của bảng được quan hệ (Table related).



Bật chức năng Enforce Referential Integrity (Nếu muốn quan hệ này bị ràng buộc tham chiếu toàn vẹn), chọn mỗi quan hệ (one-many) hoặc (one-one).

Chọn nút **Create**.

Chú ý

Quan hệ có tính tham chiếu toàn vẹn sẽ đảm bảo các vấn đề sau:

Khi nhập dữ liệu cho trường tham gia quan hệ ở bên nhiều thì phải tồn tại bên một.

Không thể xoá một bản ghi của bảng bên một nếu trong quan hệ đã tồn tại những bản ghi bên nhiều có quan hệ với bản ghi bên một đó.

Trường hợp vi phạm các quy tắc trên thì sẽ nhận được thông báo lỗi.

8.4.1. Thiết lập thuộc tính tham chiếu toàn vẹn trong quan hệ

Trong khi chọn mỗi quan hệ giữa các bảng, có 2 thuộc tính tham chiếu toàn vẹn đó là *Cascade update related fields*, *Cascade Delete related records*, có thiết lập 2 thuộc tính này.

Nếu chọn thuộc tính *Cascade update related fields*, khi dữ liệu trên khoá chính của bảng bên một thay đổi thì Access sẽ tự động cập nhật sự thay đổi đó vào các trường tương ứng (có quan hệ) trên các bảng bên Nhiều, hay nói cách khác, dữ liệu ở bảng bên nhiều cũng thay đổi theo.

Nếu chọn thuộc tính *Cascade Delete related records*, khi dữ liệu trên bảng bên một bị xoá thì dữ liệu trên bảng bên nhiều cũng sẽ bị xoá..

8.4.2. Kiểu kết nối (Join type)

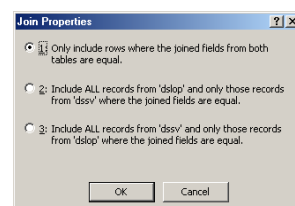
Trong quá trình thiết lập quan hệ giữa các bảng, nếu không chọn nút Create, chọn nút join type để chọn kiểu liên kết

Mục 1: Liên kết nội (Inner join)

Mục 2 và mục 3 là liên kết ngoại (outer join)

8.4.3. Điều chỉnh các mối quan hệ

Mở cửa sổ quan hệ (Tools/Relationship)



Click chuột phải, chọn edit relationship

8.4.4. Xóa các mối quan hệ

Mở cửa sổ quan hệ (Tools/Relationship)

Chọn mối quan hệ giữa các bảng, nhấn delete.

9. SẮP XẾP VÀ LỌC DỮ LIỆU

9.1. Một số phép toán và hàm

9.1.1. Một số phép toán

Ký tự thay thế: ? : Thay thế cho một ký tự bất kỳ

Ký tự * : Thay thế cho một dãy các ký tự.

Phép toán **Like**: Giống như

In: Kiểm tra một giá trị có thuộc một tập các giá trị hay không?

Is Null: Giá trị của một trường là Null.

Is not Null: Giá trị của một trường là không Null.

Between.....and: Kiểm tra xem một giá trị có thuộc một "đoạn" nào đó hay không?

9.1.2. Một số hàm

Hàm **Left\$(<Chuỗi>,<n>)**: Trích bên trái chuỗi n ký tự.

Hàm **Right\$(<Chuỗi>,<n>)**: Trích bên phải chuỗi n ký tự.

Hàm **Ucase(<Chuỗi>)**: Trả lại một chuỗi in hoa.

Hàm **Lcase(<Chuỗi>)**: Trả lại một chuỗi in thường.

Hàm **IIF(<Điều kiện>,<Giá trị 1>,<Giá trị 2>)**: Nếu <Điều kiện> nhận giá trị true thì hàm trả lại <Giá trị 1>, ngược lại hàm trả lại <Giá trị 2>.

9.2. Sắp xếp dữ liệu

9.2.1. sắp xếp trên một trường

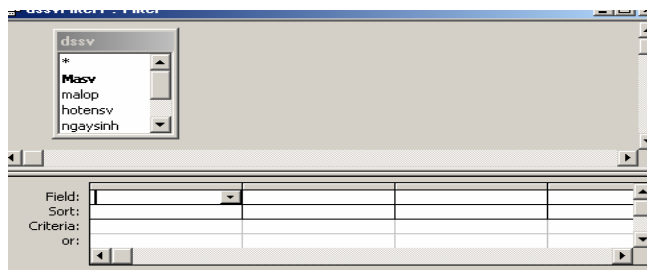
Đặt con trỏ tại trường cần sắp xếp

Thực hiện lệnh Records/ Sort/ Sort Ascending (Nếu sắp xếp tăng dần)

/ Sort Descending (Nếu sắp xếp giảm dần)

9.2.2. sắp xếp trên nhiều trường

Mở bảng trong chế độ Datasheet rồi thực hiện lệnh
Records/ Filter/ Advanced Filter sort



Trong hàng Field: Chọn các trường cần sắp xếp (Thứ tự ưu tiên từ trái sang phải)

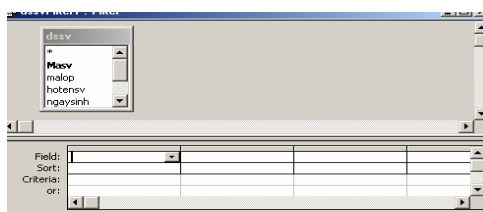
Trong hàng Sort: Chọn tiêu chuẩn sắp xếp.

Trong hàng Criteria: Chọn điều kiện sắp xếp (nếu có).

Xem kết quả.: Chọn Filter/Apply filter.

9.3. Lọc dữ liệu

Mở bảng trong chế độ Datasheet rồi thực hiện lệnh
Records/ Filter/ Advanced Filter sort



Trong hàng Field: Chọn các trường làm tiêu chuẩn lọc dữ liệu

Trong hàng Criterie: Chọn tiêu chuẩn lọc dữ liệu.

Thực hiện lệnh Filter/ Apply filter sort để xem kết quả..

Chương 3

TRUY VẤN (QUERY)

Sức mạnh thực sự của CSDL là khả năng tìm đúng và đầy đủ thông tin mà chúng ta cần biết, trình bày dữ liệu sắp xếp theo ý muốn. Để đáp ứng yêu cầu trên, Access cung cấp một công cụ truy vấn cho phép đặt câu hỏi với dữ liệu đang chứa bên trong các bảng trong CSDL.

1. KHÁI NIỆM TRUY VẤN

Truy vấn là một công cụ cho phép đặt câu hỏi với dữ liệu trong bảng dữ liệu trong CSDL.

Loại truy vấn thông dụng nhất là truy vấn chọn (Select Query). Với kiểu truy vấn này chúng ta có thể xem xét dữ liệu trong các bảng, thực hiện phân tích và chỉnh sửa trên dữ liệu đó, có thể xem thông tin từ 1 bảng hoặc có thể thêm nhiều trường từ nhiều bảng khác nhau.

Ví dụ:

Cho 2 bảng dữ liệu KHOHANG (MAHANG, TENHANG, GIA)
BANHANG(MAHANG, TENKHACH, SOLUONG, NGAYMUA). Hãy hiển thị những khách hàng mua hàng trong tháng 7 bao gồm các thông tin: MAHANG, TENHANG, GIA, TENKHACH.

Sau khi thực hiện truy vấn, dữ liệu thỏa mãn yêu cầu được rút ra và tập hợp vào một bảng kết quả gọi là **Dynaset** (Dynamic set). Dynaset cũng hoạt động như 1 bảng (Table) nhưng nó không phải là bảng vfa kết quả khi hiển thị có thể cho phép sửa đổi.

Một loại bảng thể hiện kết quả truy vấn khác là **Snapshot**, nó tương tự như dynaset tuy nhiên không thể sửa đổi thông tin (Như truy vấn Crosstab....).

1.1. Các loại truy vấn trong Access

Select Query : Truy vấn chọn

Crosstab Query : Truy vấn tham khóa chéo (Thể hiện dòng và cột)

Action Query : Truy vấn hành động gồm

Truy vấn tạo bảng (make table Query)

Truy vấn nối (append Query)

Truy vấn cập nhật (Update Query)

Truy vấn xóa dữ liệu (Delete Query)

SQL Query : Truy vấn được viết bởi ngôn ngữ SQL.

Pass through Query : Gởi các lệnh đến một CSDL SQL như Microsoft SQL server.

1.2. Sự cần thiết của truy vấn

Khi đứng trước một vấn đề nào đó trong CSDL, nếu sử dụng công cụ truy vấn thì có thể thực hiện được các yêu cầu sau:

Sự lựa chọn các trường cần thiết.

Lựa chọn những bản ghi.

Sắp xếp thứ tự các bản ghi.

Lấy dữ liệu chứa trên nhiều bảng khác nhau trong CSDL.

Thực hiện các phép tính.

Sử dụng truy vấn làm nguồn dữ liệu cho một biểu mẫu (Form), báo cáo (report) hoặc một truy vấn khác (Query).

Thay đổi dữ liệu trong bảng.

2. CÁC CHẾ ĐỘ HIỂN THỊ TRUY VẤN

2.1. Cửa sổ thiết kế truy vấn (Design view).

Trong chế độ này, người sử dụng có thể tạo, sửa chữa một truy vấn nào đó. Màn hình truy vấn chứa hai phần, phần thứ nhất chứa các bảng (hoặc truy vấn) tham gia truy vấn, phần thứ hai gọi là vùng lưới QBE (Query By Example).

2.2. Cửa sổ hiển thị truy vấn (DataSheet view).

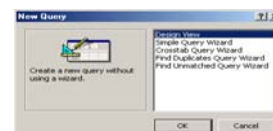
Sử dụng chế độ này để xem kết quả.

2.3. Cửa sổ lệnh SQL (SQL view).

Sử dụng chế độ này để xem mã lệnh của truy vấn đang tạo

3. TẠO TRUY VẤN

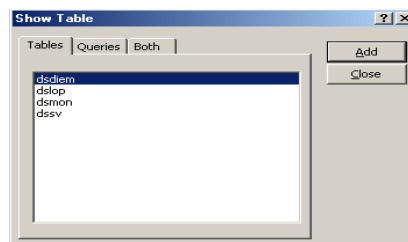
3.1. Tạo mới 1 truy vấn



Từ cửa sổ Database, click vào đối tượng Queries.

Chọn nút New.

Chọn Design View, chọn OK



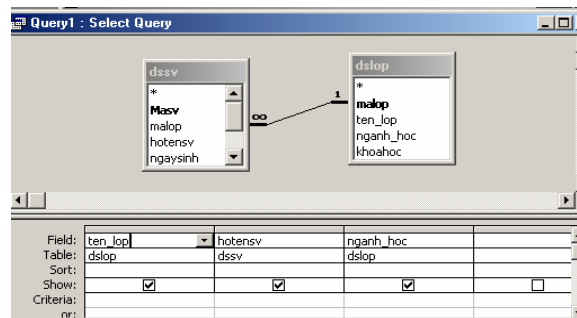
Trong bảng Show Table chọn tables để hiển thị các bảng, chọn các bảng tham gia vào truy vấn và nhấn nút **Add**, sau đó nhấn **Close**. (Nếu chọn Queries thì hiển thị truy vấn, chọn both thì hiển thị cả truy vấn và bảng dữ liệu).

Đưa các trường từ các bảng vào tham gia truy vấn bằng cách kéo các trường và thả vào hàng **Field** trong vùng lưới QBE.

Trong hàng Sort: Sắp xếp dữ liệu (nếu có)

Trong hàng **Criteria** đặt tiêu chuẩn (nếu có)

Lưu truy vấn.



☞ Chú ý

Mỗi truy vấn có:

Tối đa là 32 bảng tham gia.

Tối đa là 255 trường.

Kích thước tối đa của bảng dữ liệu (do truy vấn tạo ra) là 1 gigabyte.

Số trường dùng làm khóa sắp xếp tối đa là 10.

Số truy vấn lồng nhau tối đa là 50 cấp.

Số ký tự tối đa trong ô của vùng lưới là 1024.

Số ký tự tối đa trong dòng lệnh SQL là 64000.

Số ký tự tối đa trong tham số là 255.

3.2. Thay đổi thứ tự, xóa các trường

Các trường trong truy vấn sẽ hiển thị theo thứ tự như xuất hiện trong vùng lưới QBE.

3.2.1. Thay đổi thứ tự của trường

Đưa con trỏ vào thanh chọn sao cho con trỏ biến thành hình mũi tên trỏ xuống

Click để chọn trường

Drag để thay đổi vị trí.

3.2.2.xoá trường

Đưa con trỏ vào thanh chọn sao cho con trỏ biến thành hình mũi tên trở xuống

Click để chọn trường

Nhấn phím delete (Nếu muốn xoá tất cả các trường trong vùng lưới QBE: chọn Edit/clear grid)

3.3. Thể hiện hoặc che dấu tên bảng trong vùng lưới QBE

Muốn biết tên trường hiện tại trong vùng lưới QBE là của tên bảng nào, tại chế độ Design View người sử dụng thực hiện View/tables name.

3.4. Xem kết quả của truy vấn.

Tại cửa sổ Database chọn tên truy vấn rồi chọn Open, hoặc trong khi thiết kế truy vấn thực hiện lệnh View/datasheet View.

3.5. Đổi tiêu đề cột trong truy vấn.

Đổi tên tiêu đề cột trong truy vấn mục đích là làm cho bảng kết xuất dễ đọc hơn (Trừ khi đã quy định thuộc tính Caption).

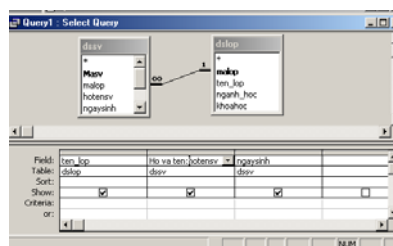
Muốn đổi tên tiêu đề cột thực hiện theo các bước sau:

Mở truy vấn ở chế độ Design View

Chọn vào bên trái ký tự đầu tiên của tên trường trong vùng lưới QBE

Gõ vào tên mới, theo sau là dấu 2 chấm (:).

Ví dụ:



| ten_lop | Họ và tên | ngaysinh |
|-----------|-------------------|----------|
| Tin K23 A | Nguyễn Tèo | 10-10-78 |
| Tin K23 B | Trần Bình Chi | 05-05-79 |
| Tin K24 A | Nguyễn Thanh Thuý | 12-10-79 |
| Tin K24 A | Lê Mậu Ngọc | 08-08-79 |
| Tin K24 B | Võ Tông | 02-02-79 |
| Tin K25 A | Nguyễn Khánh | 10-05-75 |
| Tin K25 A | Nguyễn Tiểu | 10-10-78 |
| Tin K25 A | Trần Bạch đăng | 20-10-79 |
| Tin K25 B | Lê Thuý Thu | 03-03-79 |
| Tin K25 B | Trần Thanh | 25-07-77 |

3.6. Định thứ tự sắp xếp

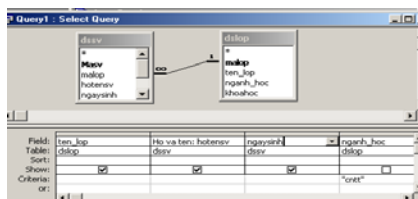
Có thể sử dụng nhanh trong chế độ datasheet View .

Có thể tạo sắp xếp trong khi thiết kế truy vấn bằng cách chọn Ascending (tăng dần) hoặc Descending (giảm dần) trong hàng Sort của vùng lưới QBE .

☞ **Chú ý:** Nếu có nhiều trường định vị sắp xếp thì theo thứ tự ưu tiên từ trái sang phải.

3.7. Che dấu hay thể hiện các trường trong Dynaset

Tại hàng Show ứng với trường cần che dấu chúng ta không chọn mặt dù nó vẫn tồn tại, vẫn tham gia truy vấn.



| ten_lop | Họ và tên | ngaysinh |
|-----------|----------------|----------|
| Tin K24B | Vũ Tông | 02-02-79 |
| Tin K25 A | Nguyễn Khánh | 10-05-75 |
| Tin K25 A | Nguyễn Tiểu | 10-10-78 |
| Tin K25 A | Trần Bạch đặng | 20-10-79 |
| Tin K25 B | Lê Thủy Thu | 03-03-79 |
| Tin K25 B | Trần Thanh | 25-07-77 |

3.8. Mối quan hệ giữa thuộc tính của trường trong truy vấn và trong bảng dữ liệu

Theo mặc nhiên, các trường trong truy vấn kế thừa tất cả các thuộc tính của trường trong bảng làm nguồn dữ liệu. Nếu không quy định lại trong truy vấn, các trường trong Dynaset hoặc snapshot luôn kế thừa các thuộc tính của bảng làm nguồn dữ liệu. Nếu thay đổi thiết kế trong bảng làm nguồn dữ liệu và thay đổi thuộc tính của các trường thì thuộc tính này cũng được thay đổi trong truy vấn. Tuy nhiên, nếu quy định lại các thuộc tính cho các trường trong truy vấn thì các thuộc tính của các trường trong bảng làm nguồn dữ liệu không thay đổi.

4. THIẾT KẾ TRUY VẤN CHỌN

4.1. Định nghĩa truy vấn chọn

Truy vấn chọn là loại truy vấn được chọn lựa, rút trích dữ liệu từ các bảng dữ liệu thỏa mãn một hoặc nhiều điều kiện nào đó. Khi thực hiện truy vấn chọn, Access tác động lên dữ liệu và thể hiện các bản ghi thỏa mãn các điều kiện đặt ra trong một bảng kết quả gọi là Recordset.

4.2. Lập phép chọn trong truy vấn

4.2.1. Chọn một nhóm các bản ghi thỏa mãn một điều kiện nào đó

Muốn thực hiện các phép chọn trong khi thể hiện truy vấn người ta thường sử dụng các phép toán sau:

| Phép toán | Ví dụ | Ý nghĩa |
|------------------|-------------------------------|----------------------------|
| < | <#20/10/99# | Trước ngày 20/10/99 |
| > | >#10/10/98# | Sau ngày 10/10/98 |
| >= | >= #05/05/90# | Sau và trong ngày 05/05/90 |
| <> | <>#01/01/99# | Khác ngày 01/01/99 |
| = | = #10/10/97# | Trong ngày 10/10/97 |
| Between and | Between #1/2/97# and #1/7/97# | Từ ngày 1/2/97 đến 1/7/97 |
| ... | | |

Ví dụ:

Cho 2 bảng dữ liệu Dslop(**Malop**, Tenlop, Ngành_hoc, khoa_hoc)

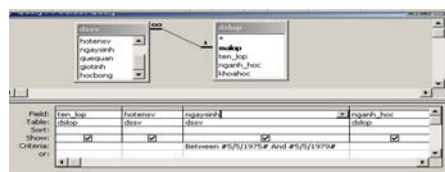
Dssv(Masv, malop, hotensv, ngaysinh, quequan, giotinh, hoc_bong)

Tạo một truy vấn để hiển thị danh sách những sinh viên có ngaysinh trong khoảng thời gian từ 05/05/75 đến 05/05/79 bao gồm các trường: Tenlop, Hotensv, Ngaysinh, ngành_hoc.

Tạo truy vấn chọn và đưa 2 bảng dslop và dssv vào tham gia truy vấn

Đưa các trường Tenlop, hotensv, ngaysinh, ngành_hoc vào vùng lưới QBE

Trong hàng Criteria của trường Ngaysinh: Between #05/05/75# and #05/05/79#



| Query1 : Select Query | | | |
|-----------------------|---------------|----------|-----------|
| ten_lop | hotensv | ngaysinh | nganh_hoc |
| Tin K23 A | Nguyễn Tèo | 10-10-78 | Tin học |
| Tin K23 B | Trần Bình Chi | 05-05-79 | Tin học |
| Tin K24 B | Võ Tông | 02-02-79 | CNTT |
| Tin K25 A | Nguyễn Khánh | 10-05-75 | CNTT |
| Tin K25 B | Lê Thuý Thu | 03-03-79 | CNTT |
| Tin K25 B | Trần Thanh | 25-07-77 | CNTT |
| Tin K25 A | Nguyễn Tiểu | 10-10-78 | CNTT |

4.2.2. Ký tự thay thế

Ký tự * : Thay thế một nhóm ký tự bất kỳ.

Ký tự ? : Thay thế 1 ký tự.

Ký tự [] : Thay thế các ký tự trong ngoặc vuông.

Ký tự ! : Phủ định.

Ký tự - : Từ ký tự đến ký tự.

Ví dụ

Cho 2 bảng dữ liệu Dslop(**Malop**, Tenlop, Ngành_hoc, khoa_hoc)

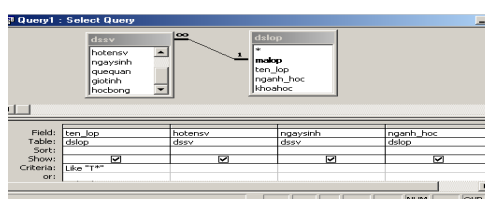
Dssv(Masv, malop, hotensv, ngaysinh, quequan, giotinh, hoc_bong)

Tạo một truy vấn để hiển thị danh sách những sinh viên có Tenlop bắt đầu là "T" bao gồm các trường: Tenlop, Hotensv, Ngaysinh, ngành_hoc.

Tạo truy vấn chọn và đưa 2 bảng dslop và dssv vào tham gia truy vấn

Đưa các trường Tenlop, hotensv, ngaysinh, ngành_hoc vào vùng lưới QBE

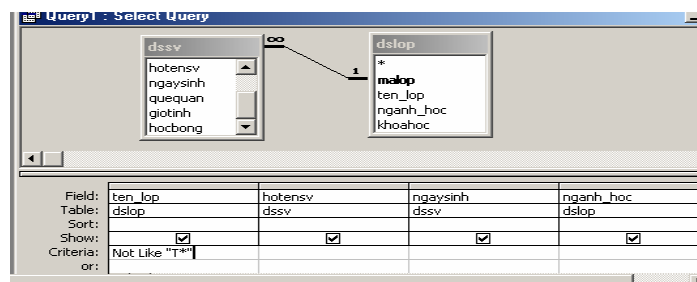
Trong hàng Criteria của trường Tenlop: Like "T*"



4.2.3. Chọn các bản ghi không phù hợp với một giá trị nào đó

Dùng toán tử **Not**

Ví dụ: Tạo một truy vấn để hiển thị danh sách những sinh viên có Tenlop không bắt đầu là "T" bao gồm các trường: Tenlop, Hotensv, Ngaysinh, ngành_hoc.



4.2.4. Định nhiều tiêu chuẩn trong lựa chọn

Dùng phép “Và” và phép “Hoặc” trong một trường

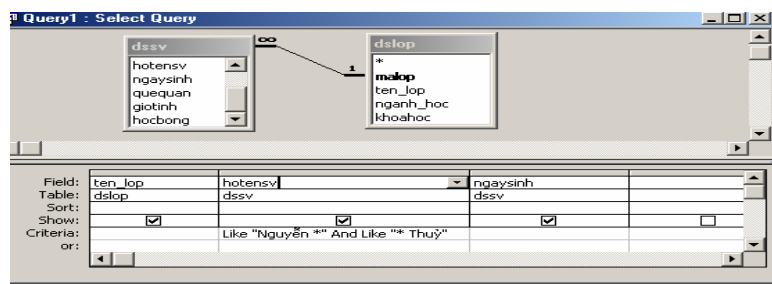
Muốn quy định nhiều tiêu chuẩn trong cùng một trường, chúng ta phải sử dụng toán tử AND (và) cùng toán tử OR (hoặc).

Ví dụ: Tạo một truy vấn để hiển thị danh sách những sinh viên có Họ là "Nguyễn" và Tên "Thuỷ" bao gồm các trường: Tenlop, Hotensv, Ngaysinh.

Tạo truy vấn chọn và đưa 2 bảng dslop và dssv vào tham gia truy vấn

Đưa các trường Tenlop, hotensv, ngaysinh vào vùng lưới QBE

Trong hàng Criteria của trường Hotensv : Like "Nguyễn *" and "* Thuỷ"



Dùng phép “Và” và phép “Hoặc” trên nhiều trường

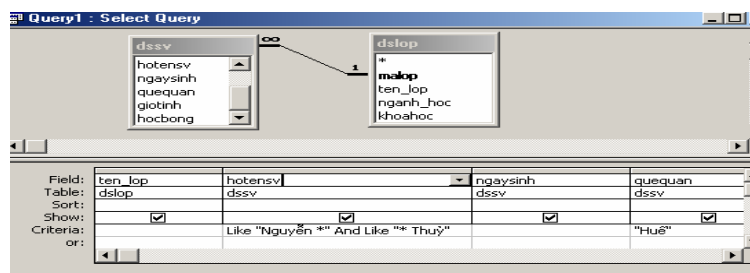
Ví dụ: Tạo một truy vấn để hiển thị danh sách những sinh viên có Họ là "**Nguyễn**" và tên "**Thuỷ**" và có quê quán ở "**Huế**" bao gồm các trường: Tenlop, Hotensv, ngaysinh, Quequan.

Tạo truy vấn chọn và đưa 2 bảng dslop và dssv vào tham gia truy vấn

Đưa các trường Tenlop, hotensv, ngaysinh, quequan vào vùng lưới QBE

Trong hàng Criteria của trường Hotensv : Like "Nguyễn *" and "* Thuỷ"

Quequan : Huế



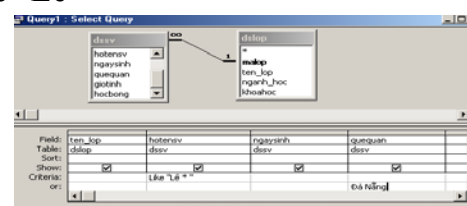
Tạo một truy vấn để hiển thị danh sách những sinh viên có Họ là "**Lê**" hoặc có quê quán ở "**Đà Nẵng**" bao gồm các trường: Tenlop, Hotensv, Ngaysinh, Quequan.

Tạo truy vấn chọn và đưa 2 bảng dslop và dssv vào tham gia truy vấn

Đưa các trường Tenlop, hotensv, ngaysinh, Quequan vào vùng lưới QBE

Trong hàng Criteria của trường Hotensv : Like "Lê *"

Trong hàng or của trường Quequan : Đà Nẵng



4.2.5. Chọn các bản ghi có chứa có giá trị

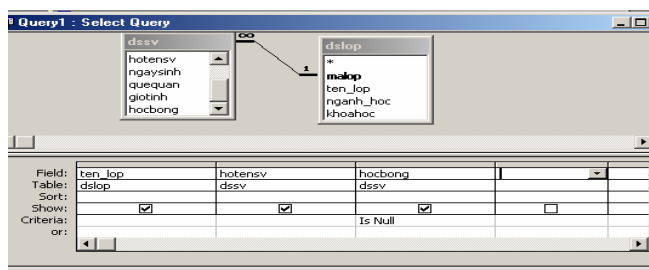
Chúng ta có thể chọn các bản ghi có chứa hoặc không chứa giá trị, chẳng hạn như tìm những sinh viên mà không có số điện thoại nhà ở.

Access cung cấp 2 phép toán

| Phép toán | Ý nghĩa |
|-------------|---------------------------|
| IS NULL | Trường không chứa giá trị |
| IS NOT NULL | Trường có chứa giá trị |

Ví dụ:

Tạo một truy vấn để hiển thị danh sách những sinh viên không có học bổng bao gồm các trường: Tenlop, Hotensv, hocbong.

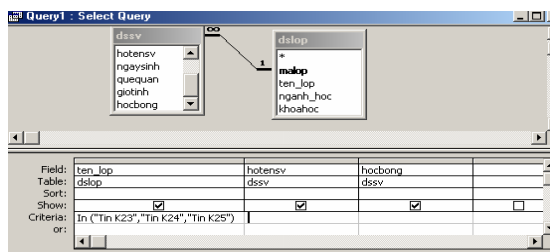


4.2.6. Chọn các bản ghi thuộc danh sách các giá trị nào đó

Chúng ta có thể sử dụng phép toán IN (Danh sách giá trị)

Ví dụ

Tạo một truy vấn để hiển thị danh sách những sinh viên thuộc lớp "Tin K23" hoặc "Tin K24" hoặc "Tin K25" bao gồm các trường: Tenlop, Hotensv, hocbong.



4.2.7. Tham chiếu đến các trường khác

Nếu trong biểu thức chọn của truy vấn, các tính toán tham chiếu đến các trường phải đặt trong dấu [], trường hợp tham chiếu đến trường của bảng khác phải chỉ rõ bảng nguồn của nó. **[Tên bảng]![Tên trường]**.

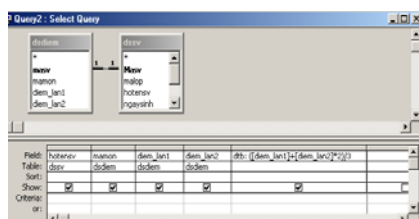
4.2.8. Tạo trường kiểu biểu thức

Ví dụ

Cho 2 bảng dữ liệu Dssv(**Masv**, hotensv, ngaysinh, quequan, gioitinh)

Dsdiem(**Masv**, mamon, diem_lan1, diem_lan2)

Tạo truy vấn để hiển thị các thông tin: Hotensv, mamon, diem_lan1, diem_lan2, dtb, trong đó $dtb = (diem_lan1 + diem_lan2 * 2) / 3$.



| hotensv | mamon | diem_lan1 | diem_lan2 | dtb |
|-------------------|-------|-----------|-----------|-------|
| Nguyễn Tèo | 01 | 10 | 10 | 10.00 |
| Tấn Bình Chi | 02 | 8 | 3 | 4.67 |
| Nguyễn Thanh Thủy | 01 | 7 | 8 | 7.67 |
| Võ Tùng | 03 | 9 | 8 | 8.33 |
| Lê Mậu Ngọ | 02 | 3 | 4 | 3.67 |
| Nguyễn Khả Linh | 03 | 6 | 7 | 6.67 |
| Lê Thủy Tiên | 01 | 4 | 5 | 4.67 |
| Tấn Thành | 02 | 6 | 7 | 6.67 |
| Nguyễn Túc | 02 | 5 | 9 | 7.67 |
| Tấn Bách strong | 03 | 3 | 9 | 7.00 |

☞ Chú ý

Sau khi thực hiện truy vấn chúng ta không thể thay đổi giá trị trong trường kiểu biểu thức, tuy nhiên nếu thay đổi giá trị trong trường tham gia biểu thức thì kết quả trong trường kiểu biểu thức cũng thay đổi theo.

4.2.9. Chọn giá trị duy nhất

Theo mặc định, access sẽ chọn tất cả các bản ghi thỏa mãn điều kiện, tuy nhiên đôi khi có nhiều giá trị giống nhau được lặp đi lặp lại, do đó để cô đọng dữ liệu thì chúng ta có thể quy định thuộc tính duy nhất trong khi hiển thị

Thuộc tính Unique-values

Chọn Yes: Không thể hiện các giá trị trùng nhau

Chọn No: Thể hiện các giá trị trùng nhau

Thuộc tính Unique-Records

Chọn Yes: Không thể hiện các bản ghi trùng nhau

Chọn No: Thể hiện các bản ghi trùng nhau

4.2.10. Chọn các giá trị đầu

Khi hiển thị truy vấn đôi khi chúng ta muốn hiển thị một số bản ghi đầu tiên nào đó thoả mãn các điều kiện thì sử dụng thuộc tính **Top values**.

5. TRUY VẤN DỰA TRÊN NHIỀU BẢNG DỮ LIỆU

Để tạo truy vấn dựa trên nhiều bảng dữ liệu thì các bảng đó phải được thiết lập mối quan hệ, nếu các bảng không thiết lập mối quan hệ thì khi truy vấn dữ liệu access sẽ cho ra những bộ dữ liệu là tích Đề-Các giữa các bộ dữ liệu trong các bảng.

5.1. Liên kết các bảng trong truy vấn

Khi các bảng dữ liệu được thiết lập mối quan hệ thì trường nối với nhau gọi là trường liên kết, trong access phân biệt 3 loại liên kết sau

5.1.1. Liên kết nội (*Inner join*)

Đây là loại liên kết rất phổ biến nhất giữa 2 bảng dữ liệu. Trong đó dữ liệu khi thể hiện trên Dynaset sẽ gồm những bản ghi mà dữ liệu chứa trong trường liên kết ở hai bảng phải giống nhau hoàn toàn.

5.1.2. Liên kết ngoại (*Outer join*)

Đây là loại liên kết cho phép dữ liệu thể hiện trên Dynaset của một trong hai bảng tham gia có nội dung trường liên kết không giống nội dung trong trường tương ứng của bảng còn lại. Liên kết ngoại được chia làm hai loại

Left Outer Join: Trong kiểu liên kết này, dữ liệu ở bảng bên "1" thể hiện toàn bộ trên Dynaset và chỉ những bản ghi bên bảng "nhiều" có nội dung trong trường liên kết giống trường tương ứng bên bảng "1".

Right Outer Join: Trong kiểu liên kết này, dữ liệu ở bảng bên "nhiều" thể hiện toàn bộ trên Dynaset và chỉ những bản ghi bên bảng "1" có nội dung trong trường liên kết giống trường tương ứng bên bảng "nhiều".

5.1.3. Tự liên kết (*Self join*)

Là kiểu liên kết của một bảng dữ liệu với chính nó. Trong đó một bản ghi trong bảng dữ liệu sẽ liên kết với những bản ghi khác trong bảng dữ liệu đó. Tự liên kết có thể hiểu như là liên kết nội hay liên kết ngoại từ một bảng vào một bảng sao chính nó. Để thực hiện việc tạo tự liên kết chúng ta phải đưa một bảng vào tham gia truy vấn 2 lần.

5.2. Tạo liên kết ngoại

Muốn tạo liên kết ngoại giữa 2 bảng dữ liệu ta thực hiện

Tools/Relationships

Double click vào đường liên kết giữa 2 bảng dữ liệu, chọn Join Type



Trong hộp thoại Join Properties chọn mục 2 hoặc mục 3.

5.3. Tạo một tự liên kết

Để tạo một tự liên kết chúng ta thực hiện

Tạo truy vấn mới và đưa bảng dữ liệu vào truy vấn 2 lần

Tạo các liên kết

Ví dụ:

Cho bảng Dsdiem(Masv, Hoten, Diem_lan1, Diem_lan2)

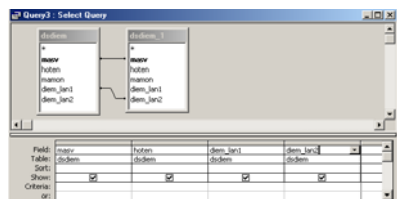
Tạo truy vấn để hiển thị danh sách những sinh viên có điểm thi Lần 1 bằng điểm thi lần 2...

Tạo truy vấn mới và đưa bảng Dsdiem vào tham gia truy vấn 2 lần

Tạo liên kết nội (Inner join) giữa 2 trường Masv

Tạo liên kết nội từ trường Diem_lan1 vào Diem_lan2

Đưa các trường vào vùng lưới QBE và xem kết quả



| masv | hoten | diem_lan1 | diem_lan2 |
|------|----------|-----------|-----------|
| a001 | Lan Anh | 10 | 10 |
| a008 | Hong Ngu | 6 | 6 |

5.3. Tự động tìm kiếm dữ liệu (Auto lookup)

Khi nhập dữ liệu vào Dynaset, chức năng tự động tìm kiếm dữ liệu cho phép chúng ta chỉ nhập dữ liệu ở các trường của bảng bên "nhiều" (Ở quan hệ 1-∞) còn Access sẽ tự động tìm kiếm dữ liệu tương ứng trên bảng "1" để hiển thị.

Chức năng Auto Lookup hoạt động trong các truy vấn mà hai bảng tham gia có mối quan hệ 1-∞.

Tạo một truy vấn có sử dụng chức năng Auto Lookup chúng ta thực hiện:

Tạo truy vấn và đưa 2 bảng vào tham gia truy vấn

Đưa trường liên kết của bảng bên nhiều vào vùng lưới QBE

Đưa các trường cần hiển thị dữ liệu của bảng bên "1".

☞ **Chú ý:** Khi nhập dữ liệu chỉ nhập dữ liệu ở các trường của bảng "nhiều"

6. TÍNH TỔNG TRONG TRUY VẤN CHỌN

Trong thực tế, chúng ta thường có những câu hỏi đặt ra về việc nhóm dữ liệu nào đó, chẳng hạn trong tháng 10 công ty xăng dầu XYZ bán được bao nhiêu lít xăng, tổng thành tiền bao nhiêu?

Trong Access chúng ta có thể thực hiện một số phép tính lên một nhóm bản ghi bằng cách dùng truy vấn tính tổng

Một số phép toán thường sử dụng

| Phép toán | Ý nghĩa |
|-------------------|---|
| <i>Sum</i> | Tính tổng các giá trị của một trường |
| <i>Avg</i> | Tính giá trị trung bình của một trường |
| <i>Min</i> | Tính giá trị nhỏ nhất của một trường |
| <i>Max</i> | Tính giá trị lớn nhất của một trường |
| <i>Count</i> | Đếm số giá trị khác rỗng có trong một trường |
| <i>First</i> | Giá trị của trường ở bản ghi đầu tiên trong bảng |
| <i>Last</i> | Giá trị của trường ở bản ghi cuối cùng trong bảng |
| <i>Where</i> | Giới hạn điều kiện khi tính tổng |
| <i>Expression</i> | Trường kiểu biểu thức |

☞ **Chú ý:** Khi thực hiện truy vấn Total, dữ liệu trong bảng kết quả của nó trình bày không thể chỉnh sửa.

6.1. Tạo truy vấn tính tổng

Tạo truy vấn chọn và đưa các bảng vào tham gia truy vấn

Thực hiện lệnh: View/Totals

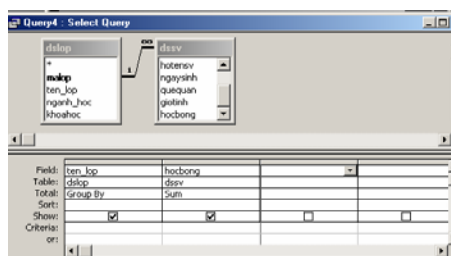
Trong vùng lưới QBE:

Tại hàng Field chọn các trường

Total chọn các phép toán tương ứng.

Criteria: Chọn điều kiện giới hạn tính tổng (Nếu có)

Lưu và thực hiện truy vấn



6.2. Tính tổng của tất cả các bản ghi

Tạo truy vấn chọn.

Đưa các bảng cần thiết vào truy vấn.

Đưa các trường cần thiết vào vùng lưới QBE

Chọn menu View/Totals, dòng Total sẽ xuất hiện trên vùng lưới.

Trong hàng total của mỗi trường chọn phương pháp tính tổng (Sum, Avg, count...).

Vì đang tính tổng của tất cả các bản ghi nên không được phép chọn “Group by” ở bất kỳ trường nào.

Chuyển sang DataSheet View để xem kết quả. (View/ Datasheet View)

Ví dụ:

Để quản lý các mặt hàng bán ra trong một cửa hàng người ta sử dụng 2 bảng dữ liệu như sau:

Dshang(**Mahang**, tenhang, dongia)

Dskhach(Mahang, tenkhach, ngaymua, diachi, soluong, thanhtien)

Tạo truy vấn để thống kê xem trong cửa hàng bán bao nhiêu mặt hàng và trung bình đơn giá của mỗi mặt hàng là bao nhiêu?

Tạo truy vấn và đưa bảng Dshang vào tham gia truy vấn

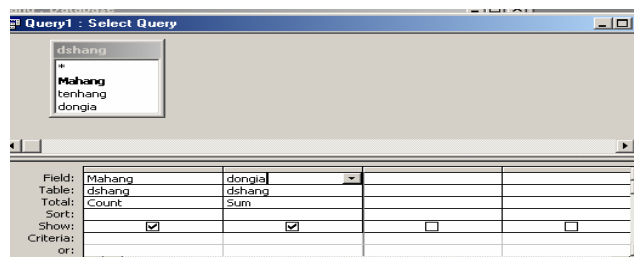
Đưa 2 trường Mahang và dongia vào vùng lưới QBE.

Chọn View/Totals

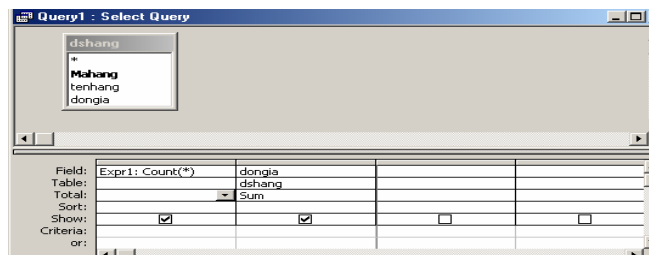
Trong hàng Total của trường Mahang chọn phép toán Count

Trong hàng Total của trường Dongia chọn phép toán Avg

Chọn View/Datasheet View để xem kết quả.



☞ **Chú ý:** Có thể đếm số bản ghi bằng cách dùng Count(*)



6.3. Tính tổng trên từng nhóm bản ghi

Trong nhiều trường hợp chúng ta có thể tính toán trên một nhóm bản ghi nào đó. Chúng ta quy định khi thiết kế truy vấn những trường nào sẽ được tính theo nhóm, trường nào sẽ được tính tổng.

Tạo truy vấn

Đưa các bảng tham gia vào truy vấn

Đưa các trường vào vùng lưới

Chọn View/Totals

Tại hàng total

Chọn “Group by” cho trường làm khóa để nhóm

Chọn các phép toán tính tổng (Sum,count..) cho các trường còn lại

Chọn View/Datasheet View để xem kết quả.

Ví dụ:

Tạo một truy vấn để tính tổng soluong, thanhtien của mỗi mặt hàng bán được là bao nhiêu?

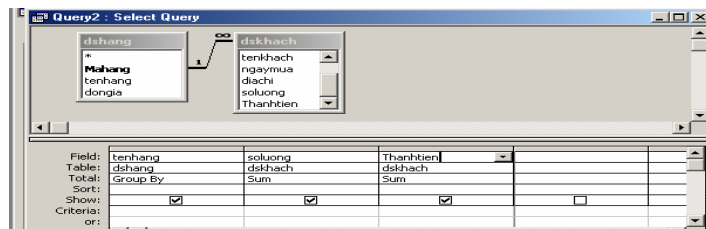
Tạo truy vấn và đưa 2 bảng dshang và dskhach vào tham gia truy vấn

Chọn View/Totals

Đưa các trường Tenhang, soluong, thanhtien vào vùng lưới QBE.

Tại hàng Total của trường Tenhang: Chọn phép toán Group by

Tại hàng Total của trường Soluong, thanhtien: Chọn phép toán SUM.



Chọn View/Datasheet View để xem kết quả.

| tenhang | SumOfsoluong | SumOfThanhTien |
|----------|--------------|----------------|
| Dau hoa | 35 | 140000 |
| Xang A83 | 83 | 415000 |
| XangA92 | 78 | 468000 |

Ta có thể thay đổi tiêu đề cột trong khi thực hiện truy vấn tính tổng như sau

| tenhang | Tong so luong | Tong thanh tien |
|----------|---------------|-----------------|
| Dau hoa | 35 | 140000 |
| Xang A83 | 83 | 415000 |
| XangA92 | 78 | 468000 |

6.4. Tính tổng trên nhiều nhóm bản ghi

Access cho phép tính tổng không chỉ trên một mà còn nhiều nhóm bản ghi.

Để làm được điều đó chúng ta chọn “Group by” trên nhiều trường và khi thực hiện

Access sẽ theo thứ tự từ trái sang phải trường bên trái là nhóm mức cao hơn, trường kế tiếp theo là nhóm mức thấp hơn.

Ví dụ:

Tạo truy vấn để tính tổng thanh tiền của mỗi mặt hàng bán được theo từng năm nào đó?

Tạo truy vấn và đưa 2 bảng dshang và dskhach vào tham gia truy vấn

Chọn View/Totals

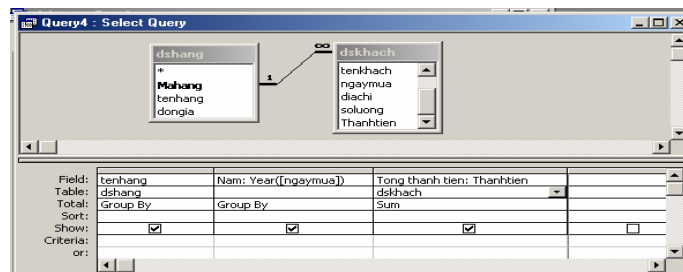
Đưa các trường tenhang, ngaymua, thanh tiền vào vùng lưới QBE

Tại hàng Total của trường tenhang chọn phép toán Group by

Tại hàng field của trường ngayban sử dụng hàm year([ngayban]) và tại hàng total chọn phép toán Group by.

Tại hàng Total của trường Thanh tiền chọn phép toán Sum.

Thay đổi tiêu đề cột trong truy vấn.



Chọn View/ Datasheet View để xem kết quả

| tenhang | Nam | Tong thanh tien |
|----------|------|-----------------|
| Dau hoa | 1997 | 24000 |
| Dau hoa | 1998 | 88000 |
| Dau hoa | 1999 | 28000 |
| Xang A83 | 1997 | 45000 |
| Xang A83 | 1998 | 250000 |
| Xang A83 | 1999 | 120000 |
| XangA92 | 1997 | 120000 |
| XangA92 | 1998 | 168000 |
| XangA92 | 1999 | 180000 |

6.5. Lập biểu thức chọn cho các trường dùng để nhóm khi tính tổng

Cũng như với những truy vấn khác, chúng ta có thể lập biểu thức chọn cho truy vấn tính tổng theo từng nhóm.

Để thực hiện công việc này, chúng ta lập biểu thức điều kiện ngay hàng Criteria của trường “group by”.

Ví dụ:

Tạo truy vấn để tính tổng thanh tiền của mỗi mặt hàng bán được trong năm 1999.

Tạo truy vấn và đưa 2 bảng dshang và dskhach vào tham gia truy vấn

Chọn View/Totals

Đưa các trường tenhang, ngaymua, thanh tien vào vùng lưới QBE

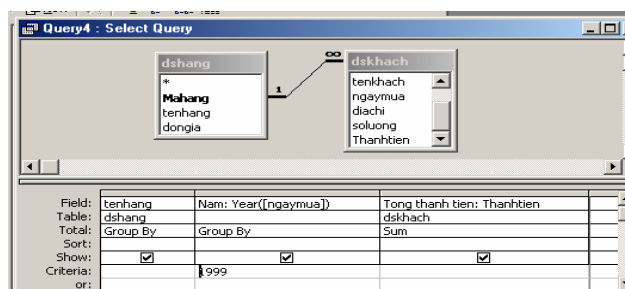
Tại hàng Total của trường tenhang chọn phép toán Group by

Tại hàng field của trường ngayban sử dụng hàm year([ngayban]) và tại hàng total chọn phép toán Group by.

Tại hàng Total của trường Thanh tien chọn phép toán Sum.

Tại hàng Criteria của trường ngaymua: gõ vào giá trị 1999

Thay đổi tiêu đề cột trong truy vấn.



Chọn View/ Datasheet View để xem kết quả

| | tenhang | Nam | Tong thanh tien |
|---|----------|------|-----------------|
| ▶ | Dau hoa | 1999 | 28000 |
| | Xang A83 | 1999 | 120000 |
| | Xang A92 | 1999 | 180000 |

Record: 1 of 3

6.6. Lập biểu thức chọn để giới hạn những bản ghi.

Trong các phần trước, chúng ta đã biết cách giới hạn các bản ghi trong truy vấn theo một điều kiện nào đó, sự giới hạn này gọi là **giới hạn sau khi tính tổng**.

Bây giờ chúng ta lập biểu thức chọn giới hạn số bản ghi trước khi đưa vào tính tổng trong truy vấn gọi là **giới hạn trước khi tính tổng**.

Cách tạo

Tạo truy vấn mới và đưa các bảng tham gia vào truy vấn

Đưa các trường vào vùng lưới QBE.

Chọn menu View/Total

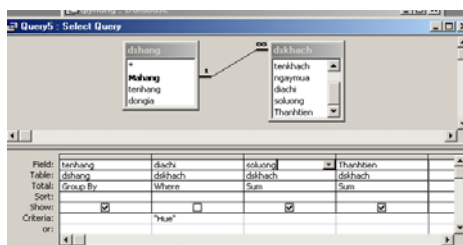
Thiết lập hàng Total thành Where đối với trường chúng ta muốn dùng để đặt biểu thức điều kiện giới hạn số bản ghi trước khi tính tổng.

Gõ biểu thức điều kiện tại hàng Criteria tương ứng.

Chuyển sang chế độ datasheet view để xem kết quả.

Ví dụ:

Tạo truy vấn để tính tổng soluong, thanhtien của mỗi mặt hàng bán được đối với khách mua hàng có Queuean ở "Huế"



Chú ý: Trong đa số trường hợp, đặt điều kiện lọc trước và sau khi tính tổng có giá trị khác nhau.

6.7. Dùng truy vấn để cập nhật bản ghi

Khi truy vấn chỉ dựa trên một bảng, hoặc chỉ bảng có quan hệ 1-1 thì tất cả các trường đều có thể thay đổi, cập nhật. Trong trường hợp có nhiều hơn hai bảng tham gia truy vấn mà có quan hệ 1-∞ thì sẽ phức tạp hơn.

6.7.1. Khi nào dữ liệu trong trường có thể sửa đổi được

Bảng sau đây liệt kê các trường hợp khi nào một trường trong kết quả truy vấn hay trong biểu mẫu có thể sửa đổi được.

| Loại truy vấn hay trường | Dữ liệu trong trường có cho phép sửa đổi hay không? |
|---|---|
| Truy vấn dựa trên 1 bảng | Có |
| Truy vấn dựa trên nhiều bảng có quan hệ 1-1 | Có |

| Truy vấn dựa trên nhiều bảng có quan hệ 1-∞ | Thông thường |
|---|--------------|
| Truy vấn Tham khảo chéo | Không |
| Truy vấn tính tổng | Không |
| Truy vấn với thuộc tính Unique values được thiết lập thành Yes | Không |
| Truy vấn hội | Không |
| Truy vấn chuyển nhượng | Không |
| Trường kiểu biểu thức | Không |
| Trường trong bản ghi đã bị xoá hoặc bị khoá bởi một người khác trong môi trường nhiều người sử dụng | Không |

6.7.1. Chỉnh sửa bản ghi trong truy vấn dựa trên hai bảng có quan hệ 1-∞

Trong truy vấn dựa trên dữ liệu là hai bảng có quan hệ 1-∞, chúng ta có thể sửa đổi tất cả các trường trừ trường liên kết của bảng bên "1". Tuy nhiên có hai trường hợp mà vẫn có thể sửa đổi dữ liệu trong trường liên kết bên "1" là:

Có thể sửa đổi dữ liệu trường liên kết bên bảng "1" trong trường hợp liên kết ngoại và tương ứng của liên kết bên bảng "nhiều" không chứa giá trị.

Có thể sửa đổi dữ liệu trường liên kết bên bảng "1" trong trường hợp đã khai báo thuộc tính tham chiếu toàn vẹn.

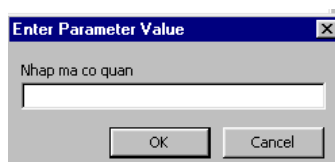
7. TRUY VẤN THAM SỐ (Parameter Query)

7.1. Khái niệm

Nếu thường xuyên chạy cùng một truy vấn, nhưng mỗi lần một tiêu chuẩn khác nhau, thay vì phải thiết kế lại truy vấn sau mỗi lần thực hiện, có thể tiết kiệm thời gian bằng cách tạo truy vấn tham số. Khi thực hiện loại này Access sẽ nhắc nhập điều kiện chọn trong hộp thoại enter parameter Value.

Ví dụ:

Giả sử thường xuyên chạy một truy vấn để liệt kê danh sách nhân viên của một cơ quan nào đó có mã cơ quan nhập vào bất kỳ.



☞ **Chú ý:** Nội dung các tham số mà chúng ta nhập vào có thể là hằng (số, chuỗi, ngày..) nhưng không được biểu thức.

7.2. Tạo truy vấn tham số

Tạo truy vấn chọn và đưa các bảng cần thiết vào tham gia truy vấn.

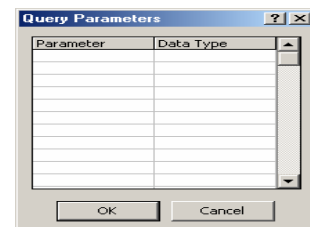
Kéo các trường cần thiết vào vùng lưới QBE.

Tại hàng Criteria gõ vào biểu thức có chứa tham số với chú ý tên tham số phải nằm giữa 2 dấu ngoặc vuông ([])

Tên tham số cũng là chuỗi nhắc nhở. Access cho phép có khoảng trắng và độ dài tối đa 255 ký tự.

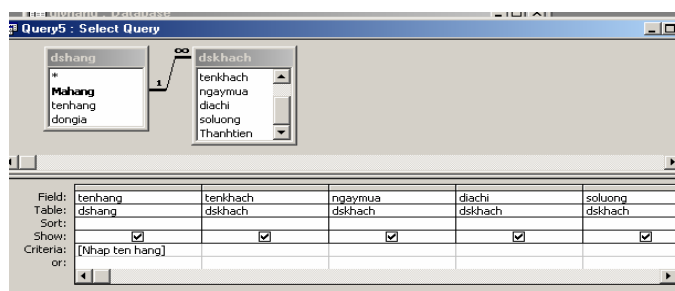
Quy định kiểu dữ liệu cho tham số: Chọn queries/ parameter query.

Trong hộp thoại query parameters: Trong mục Parameter chọn tham số, trong mục Data type chọn kiểu dữ liệu tương ứng.

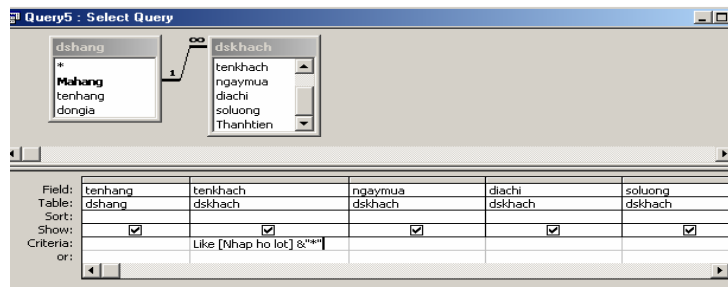


Ví dụ:

Tạo truy vấn để hiển thị danh sách các khách hàng mua một mặt hàng nào đó (mặt hàng được nhập bất kỳ từ bàn phím).



Tạo truy vấn để hiển thị danh sách các khách hàng mua hàng có họ lót được nhập từ bàn phím.



7.3. Truy vấn nhiều tham số

Có thể tạo truy vấn , khi chạy truy vấn nhập nhiều dữ liệu cho điều kiện chọn lựa. Muốn vậy tạo truy vấn nhiều tham số.

Ví dụ:

Tạo một truy vấn hiển thị danh sách các khách hàng mua hàng trong khoảng thời gian nào đó (Thời gian được nhập từ bàn phím).

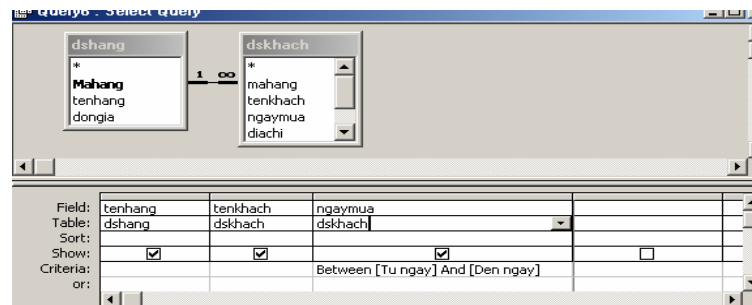
Tạo truy vấn chọn và đưa các 2 bảng dshang và dskhach vào tham gia truy vấn.

Kéo các trường tenhang, tenkhach, ngaymua vào vùng lưới QBE.

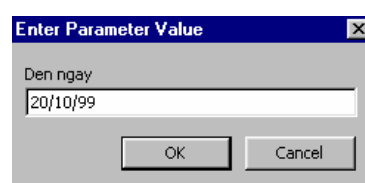
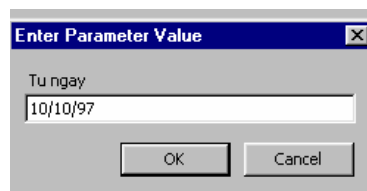
Tại hàng Criteria của trường NGAYSINH chọn:

Between [Từ ngày] and [Đến ngày]

Chọn query/Parameter khai báo kiểu dữ liệu cho 2 tham số là date/time.



Khi chạy truy vấn sẽ cho kết quả sau



7.4. Kết hợp giữa truy vấn tham số và truy vấn tính tổng

Trong nhiều bài toán quản lý người ta thường gặp những yêu cầu như: Hãy thống kê xem mỗi loại hàng trong một tháng nào đó bán được với tổng số lượng là bao nhiêu? Tổng thành tiền là bao nhiêu? (Tháng được nhập từ bàn phím). Vì vậy trước hết chúng ta phải thực hiện truy vấn tính tổng xong mới kết hợp truy vấn tham số.

Ví dụ:

Hãy tạo một truy vấn để hiển thị tổng thanh tiền của mỗi mặt hàng bán được trong một năm nào đó (Năm được nhập từ bàn phím).

Tạo một truy vấn chọn, đưa bảng Dshang và dskhach vào tham gia truy vấn.

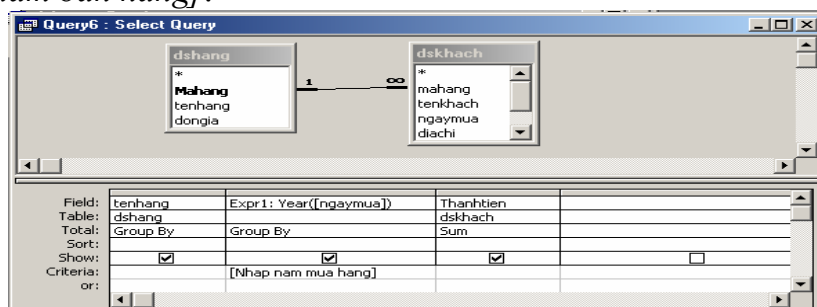
Đưa các trường tenhang, ngayban, thanh tien vào vùng lưới QBE.

Chọn View/Totals

Trong hàng Total: Chọn Group by đối với trường tenhang, và year([ngayban])

Trong hàng Criteria đối với trường Ngaysinh, ta chọn tham số sau:

[Nhập vào năm bán hàng].



8. TRUY VẤN THAM KHẢO CHÉO (Crosstab query)

8.1. Khái niệm

Truy vấn tham khảo chéo là loại truy vấn dùng để tóm lược dữ liệu và trình bày kết quả theo dạng như một bảng tính. Truy vấn tham khảo chéo cũng có thể thống kê một khối lượng dữ liệu lớn và trình bày đơn giản hơn do đó thường sử dụng để so sánh dữ liệu.

8.2. Tạo truy vấn tham khảo chéo

Muốn tạo một truy vấn tham khảo chéo chúng ta phải xác định được 3 yếu tố chính: Trường làm tiêu đề cột (Duy nhất 1 trường), trường làm tiêu đề hàng (Có thể nhiều trường), trường tính giá trị (Duy nhất 1 trường).

Cách tạo

Tạo truy vấn chọn và đưa các bảng vào tham gia truy vấn

Đưa các trường vào vùng lưới QBE

Chọn Query/Crosstab

Quy định trường làm tiêu đề cột

Tại hàng Total: Bắt buộc chọn phép toán Group by

Tại hàng Crosstab: Chọn Column heading

Quy định trường làm tiêu đề hàng

Tại hàng Total: Ít nhất một trong các trường phải chọn phép toán Group by

Tại hàng Crosstab: Chọn Row heading

Quy định trường tính giá trị

Tại hàng Total: Chọn phép toán thích hợp

Tại hàng Crosstab: Chọn Value

Ví dụ:

Cho 2 bảng dữ liệu Dstruong(**matruong**, tentruong, sodt)

Danh sach(matruong, hoten, ngaysinh, gioitinh, xeploai)

Tạo một truy vấn Crosstab để phản ánh tổng số lượng sinh viên xếp mỗi loại của trong từng trường bao nhiêu.?

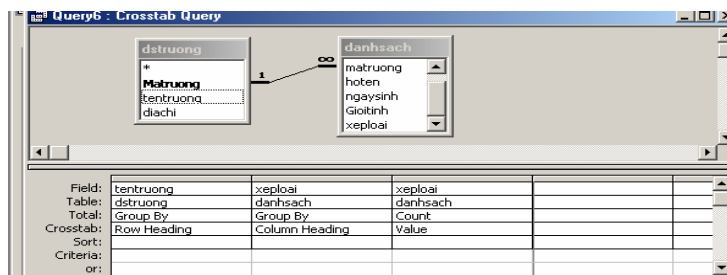
Tạo truy vấn và đưa 2 bảng dữ liệu vào tham gia truy vấn, đưa các trường tentruong và xeploai vào vùng lưới QBE. (Trường Xeploai đưa vào 2 lần)

Chọn Query/ crosstab query

Tại hàng Total của trường tentruong: Chọn phép toán Group by, hàng crosstab: chọn Row heading

Tại hàng Total của trường Xeploai: Chọn phép toán Group by, hàng Crosstab chọn Column heading.

Tại hàng Total của trường Xeploai: Chọn phép toán Count, hàng Crosstab chọn Value.



Chọn View/ Datasheet View để xem kết quả

| tentruong | Gioi | Kha | Trung binh | Yeu |
|-----------|------|-----|------------|-----|
| DHKH | | 3 | 1 | |
| DHNL | | 1 | 1 | 1 |
| DHSP | 1 | 1 | | 1 |
| DHYK | | 1 | 1 | 1 |

8.3. Định dạng cho tiêu đề cột

Với truy vấn Crosstab, chúng ta có thể can thiệp nhiều hơn về cách trình bày tiêu đề cột trong bảng. Chúng ta có thể thay đổi bằng cách đặt lại thuộc tính Column Heading của truy vấn. Thuộc tính này cho phép chúng ta: Chỉ định sắp xếp các tiêu đề cột.

Muốn định dạng tiêu đề cột thực hiện các bước sau:

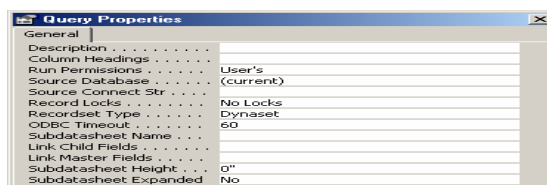
Tạo truy vấn Crosstab.

Chuyển sang chế độ Design View.

Mở bảng thuộc tính truy vấn.

Tại hàng Column Heading : Gõ các tiêu đề cột theo thứ tự mà chúng ta muốn

Các giá trị này phải cách nhau bởi dấu chấm phẩy (;)



9. TRUY VẤN HÀNH ĐỘNG

9.1. Các loại truy vấn hành động

Truy vấn hành động giúp người sử dụng tạo bảng mới hay sửa đổi dữ liệu trong các bảng. Có 4 loại truy vấn hành động:

Truy vấn tạo bảng (Make table query): Tạo bảng mới từ một bảng hay nhiều bảng đã tồn tại dữ liệu.

Truy vấn cập nhật (Update query): Dùng để cập nhật dữ liệu cho một hoặc nhiều trường trong bảng dữ liệu.

Truy vấn xoá (Delete query): Xoá các bản ghi thoả mãn các điều kiện từ một hay nhiều bảng dữ liệu.

Truy vấn nối (Append query): Nối một số bản ghi từ một hoặc nhiều bảng dữ liệu vào sau một hoặc nhiều bảng dữ liệu khác.

9.2. Truy vấn tạo bảng

Truy vấn tạo bảng sẽ tạo ra một bảng mới bằng cách rút các bản ghi thoả mãn các điều kiện nào đó.

Cách tạo truy vấn

Để tạo truy vấn tạo bảng chúng ta tạo truy vấn chọn và đưa bảng vào tham gia truy vấn. Đưa các trường vào vùng lưới QBE

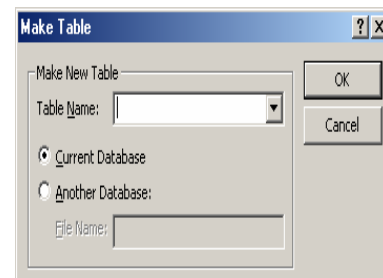
Chọn query/ make table query

Trong mục Table name: Đặt tên mới cho bảng muốn tạo.

Chọn Current Database: CSDL hiện thời

Another Database: Tạo bảng trong CSDL khác.

Chọn các điều kiện (Nếu có).



9.3. Truy vấn xoá

Truy vấn xoá giúp chúng ta loại bỏ các bản ghi thoả mãn một số điều kiện nào đó

Cách tạo truy vấn

Để tạo truy vấn xoá chúng ta tạo truy vấn chọn và đưa bảng vào tham gia truy vấn.

Chọn query/ Delete query

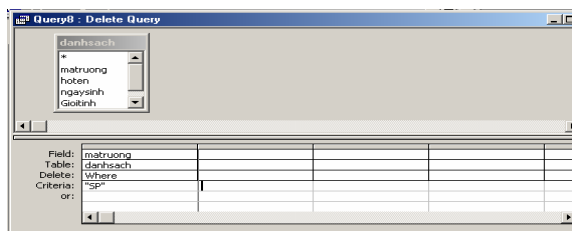
Trong vùng lưới QBE tại hàng Field chọn các trường cần so sánh với điều kiện xoá

Tại hàng Delete: Chọn phép toán Where

Tại hàng Criteria: Chọn điều kiện xoá

Ví dụ:

Tạo truy vấn để xoá những sinh viên có matruong là "SP"



9.3. Truy vấn cập nhật

Truy vấn này dùng để cập nhật giá trị hoặc sửa đổi giá trị của các trường trong bảng dữ liệu.

Cách tạo truy vấn

Tạo một truy vấn chọn và đưa bảng vào tham gia truy vấn

Chọn Query/Update query

Tại hàng Field: Chọn trường cần cập nhật dữ liệu

Tại hàng Update to: Chọn Biểu thức cần tính giá trị

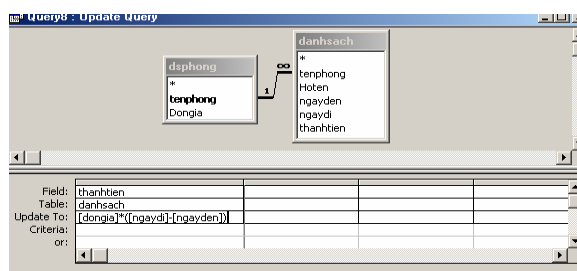
Tại hàng Criteria: Chọn điều kiện (nếu có).

Ví dụ:

Cho 2 bảng dữ liệu Dsphong(tenphong, dongia)

Dskhachtro (tenphong, ngayden, ngaydi, thanh tien)

Tạo truy vấn để cập nhật giá trị cho trường Thanh tien = (ngaydi - ngayden) * dongia



9.4. Truy vấn nối dữ liệu

Truy vấn nối dữ liệu dùng để nối dữ liệu từ một bảng này vào sau một bảng khác.

Cách tạo truy vấn

Tạo truy vấn chọn và đưa bảng dữ liệu vào để nối với bảng khác tham gia truy vấn.

Chọn Queries/Append query

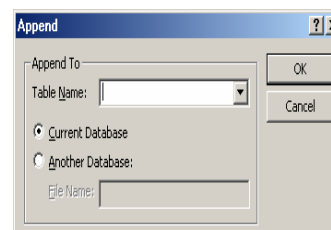
Trong mục Table name: Chọn bảng cần nối vào và chọn OK.

Chọn Current Database: CSDL hiện thời

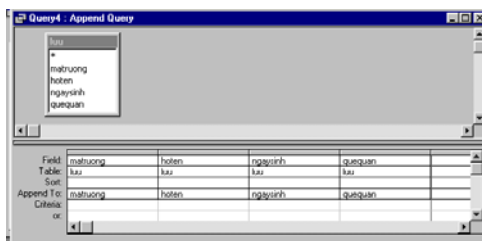
Another Database: Tạo bảng trong CSDL khác.

Trong vùng lưới QBE của truy vấn tại hàng Field:

Đưa các trường của bảng gốc vào.



Trong hàng Append to: Đưa các trường tương ứng của bảng cần nối và đặt điều kiện nếu cần thiết.



☞ Chú ý

Trong truy vấn nối dữ liệu thì các trường được nối với nhau tương ứng phải cùng kiểu dữ liệu. Nếu các trường tương ứng không có cùng kiểu dữ liệu thì sẽ không được nối. Nếu các trường có Field size không phù hợp thì tùy theo việc nối dữ liệu mà sẽ cắt bớt hoặc thêm vào ký tự trống.

1. GIỚI THIỆU VỀ NGÔN NGỮ SQL

SQL là một ngôn ngữ dùng để truy xuất dữ liệu, cập nhật, thay đổi và quản lý các CSDL quan hệ.

Khi tạo một truy vấn thì ACCESS tự động xây dựng các câu lệnh SQL tương ứng.

Một số truy vấn của ngôn ngữ SQL như sau:

Truy vấn hội (Union Query)

Truy vấn chuyển nhượng (pass through Query)

Truy vấn định nghĩa dữ liệu (Data Definition Query)

Truy vấn con (Sub Query)

☞ Chú ý

Trong SQL mỗi câu lệnh có thể được viết trên nhiều hàng và kết thúc mỗi câu lệnh là dấu chấm phẩy (;)

2. SQL XỬ LÝ TRÊN BẢNG DỮ LIỆU

2.1. Tạo bảng mới

Cú pháp

*Create table <Table name>(<Field1> <Type>[(Size)][, <Field2>
<Type>[(Size)].....*

[Constraint < Primary key name> primary key <Fieldname key>]

[Constraint <Index name> Unique <Field name Index>]

Chức năng: Tạo cấu trúc của một bảng mới.

Giải thích chức năng các tham số

Table name: Tên bảng cần tạo

Fieldname1, Fieldname2.....: Các trường trong bảng cần tạo, tối thiểu 1 trường.

Type: Các kiểu dữ liệu tương ứng.

Size : Kích thước của trường

Primary key name : Tên khoá chính

Fieldname key : Trường làm khoá chính

Index name : Tên chỉ mục

Fieldname Index : Trường làm chỉ mục

Constraint.....Primary key : Dùng để thiết lập khoá chính

Constraint..... Unique: Thiết lập chỉ mục (Không trùng nhau)

☛ **Chú ý:** Đối với các kiểu dữ liệu khi sử dụng trong SQL

| Kiểu dữ liệu chuẩn | Khai báo tương ứng trong SQL |
|-------------------------|------------------------------|
| Text | Text(Size) |
| Byte | Byte |
| Integer | Short |
| Long Integer | Long |
| Single | Single |
| Double | Double |
| Date/Time | Datetime |
| Yes/No | Yesno |
| OLE Object | OLE Object |
| Currency, Memo, Counter | Currency, Memo, Counter |

Ví dụ : Tạo bảng DSTRUONG có cấu trúc sau

| | | |
|-----------|-----------|------|
| Fieldname | Data Type | size |
| Matruong | Text | 2 |
| Tenruong | Text | 20 |
| SODT | Text | 9 |

Create table dstruong(matruong text(2), tentruong text(20), Sodt text(9));

Ví dụ Tạo bảng DSHOCVIEN có cấu trúc sau

| | | |
|-----------|-----------|----------------|
| Fieldname | Data Type | size |
| Mahv | Text | 4 (Khoá chính) |
| Tenhv | Text | 30 |
| Ngaysinh | Date/time | |

SODT Text 9 (Lập chỉ mục)

Create table dshocvien(mahv text(4), tenhv text(30), ngaysinh Datetime, sodt text(9),
Constraint khoa primary key (mahv), Constraint chimuc unique (sodt));

☞ **Chú ý:** Nếu tên trường có ký tự trắng hoặc tên bảng, tên khoá chính, tên chỉ mục thì phải đặt trong cặp dấu [...]

Ví dụ

Create table [Bang NV] ([Ma nv] text(2), [ho va ten] text(30));

2.2. Thay đổi cấu trúc của bảng

2.2.1. Thêm hoặc loại bỏ trường

Cú pháp

Alter table <Table name>[add column <fieldname><type>]

[Drop column <Field name>]

[Add Constraint <Index name> unique <fieldname>]

[Drop Constraint <Index name>]

Chức năng: Thay đổi cấu trúc của bảng

Giải thích:

ADD Column...: Thêm trường vào bảng

Drop column...: Loại bỏ trường ra khỏi bảng

Add Constraint.....: Thêm tên chỉ mục

Drop Constraint.....: Loại bỏ tên chỉ mục

Ví dụ: Giả sử đã tồn tại bảng MATHANG cấu trúc sau

| <i>Fieldname</i> | <i>Data Type</i> | <i>size</i> |
|------------------|------------------|-------------|
| MAHANG | Text | 4 |
| TENHANG | Text | 20 |
| GIA | Integer | |
| MAXN | Text | 2 |
| Ngaynhap | Date/time | |

Thêm trường SOLUONG có kiểu byte vào bảng MATHANG

Alter Table mathang soluong byte;

Ví dụ : Thêm chỉ mục có tên cmx cho trường MAXN

Alter table mathang add constraint cmx unique Maxn

Ví dụ : Loại bỏ chỉ mục cmx

Alter table mathang drop constraint cmx

Ví dụ : Loại bỏ trường ngaynhap ra khỏi bảng MATHANG

Alter table mathang drop column ngaynhap

2.2.2. Loại bỏ chỉ mục

Cú pháp

Drop Index <Index name> on <Table name>

Chức năng: Loại bỏ 1 chỉ mục nào đó.

2.3. Xóa bảng

Cú pháp

Drop table <Table name>

Chức năng: Xóa bảng dữ liệu nào đó.

Ví dụ: Xóa bảng MATHANG

Drop table MATHANG

3. SQL XỬ LÝ TRÊN TRUY VẤN

3.1. Truy vấn chọn (Select query)

Cú pháp

Select <Scope> <Fieldname1> [AS <New name>].....

From <Table name>

[Where <Condition>]

Chức năng: Tạo truy vấn chọn

Trong đó:

Scope: Phạm vi (Mặc định là ALL, Top n: Lấy n bản ghi đầu tiên)

Nếu có AS <New name> thì sẽ thay thế tên cho Fieldname tương ứng.

Table name: Tên bảng cần lấy dữ liệu.

Condition: Điều kiện để hạn chế dữ liệu.

Ví dụ: Cho bảng dữ liệu DOCGIA sau

| Fieldname | Data Type | Size | Description |
|-----------|-----------|------|-------------------------|
| MADG | Text | 2 | Mã độc giả (Khoá chính) |
| MASACH | Text | 4 | |
| HOTEN | Text | 30 | |
| QUEQUAN | Text | 30 | |
| NGAYSINH | Date/time | 8 | |
| NGAYMUON | Date/time | 8 | |

Chọn 2 trường MADG và HOTEN

Select MADG, hoten

From docgia;

Chọn 2 trường Masach và hoten mà chỉ những masach bắt đầu là T

Select MADG, hoten

From docgia where masach like “T*”;

Chọn Hoten, quequan, madg cho những độc giả có quê quán ở Huế và đổi tên trường hoten thành Họ và tên

Select hoten AS [Họ và tên], queuqan, madg

From docgia where quequan=”Huế”;

Chọn những độc giả mượn sách trong tháng 8 hoặc năm 1999.

Select * from docgia where month([ngaymuon])=8 or year([ngaymuon])=1999;

3.2. Truy vấn tính tổng (Total query)

Cú pháp

Selectfrom.....[where < Condition>]

Group by [Group fieldname]

[Having <Group Condition>];

Chức năng: Tạo một truy vấn tính tổng.

Ví dụ: Cho bảng dữ liệu BANHANG có cấu trúc

| Fieldname | Data type | size |
|-----------|-----------|------|
| Mahang | text | 2 |
| Soluong | integer | |

Ngàyban Date/time 8

Tạo một truy vấn để thống kê xem mỗi loại hàng bán được với số lượng là bao nhiêu?

```
Select mahang, sum([soluong])
```

```
From banhang
```

```
Group by mahang;
```

Tạo truy vấn để thống kê xem mỗi loại hàng trong tháng 7 bán được với số lượng bao nhiêu? Chỉ hiển thị những loại hàng mà số lượng bán trên 20.

```
Select mahang, sum([soluong])
```

```
From banhang
```

```
where month([ngayban])=7
```

```
Group by mahang
```

```
having sum([soluong])>20;
```

3. 3.Truy vấn tham khảo chéo

Cú pháp

Transform <Value Express>

Select.....From.....Where.....

Group by <Row Heading Field>

Pivot <Column heading Field>

Chức năng: Tạo truy vấn tham khảo chéo

Ví dụ

```
Transform sum([soluong])
```

```
select Mahang, tenhang, sum([soluong]) from dskhang
```

```
Group by tenhang
```

```
Pivot Mahang;
```

3.4. Truy vấn tạo bảng

Cú pháp

Select <Field select> into <New Table name>

From <Old Table name>

[Where <Condition>]

Chức năng: Tạo một truy vấn tạo bảng

Ví dụ

```
Select Mahang, tenhang into Luu
From Dskhang
Where Mahang Like “A*”;
```

3.5. Truy vấn nối dữ liệu

Cú pháp

```
Insert into <append Table name>
Select <field select>
From <Table name>
[Where <Condition>]
```

Chức năng: Tạo truy vấn nối dữ liệu

Nếu chỉ thêm 1 bản ghi với các giá trị cụ thể thì ta thực hiện câu lệnh

```
Insert into <Table name and Field list>
values <append values>
```

Ví dụ:

```
Insert into luu1(Hoten, quequan)
Values (“Nguyen an”, “Hue”)
```

3.6. Truy vấn cập nhật dữ liệu

Cú pháp

```
Update <Update Table name>
Set <Field name>=<Express>
[Where <Condition>]
```

Chức năng: Tạo một truy vấn dùng để cập nhật dữ liệu

Ví dụ:

```
Update dssv
set [hocbong]=[hocbong]+200000
Where Uutien=”1”;
```

3.7. Truy vấn xoá

Cú pháp

```
Delete <Table.*>  
From <Delete Table name>  
[Where <Condition>]
```

Chức năng: Dùng để tạo một truy vấn xoá các bản ghi trong bảng theo một hoặc nhiều điều kiện nào đó.

☞ **Chú ý:** nếu mệnh đề From chỗ có 1 bảng duy nhất thì không cần liệt kê các bảng trong mệnh đề DELETE.

Ví dụ:

Có 2 bảng dữ liệu DSHS và DSDTHI có quan hệ 1-1 trên trường MAHS. Hãy xoá những học sinh có điểm thi <5 trong bảng DSHS và DSDTHI.

```
Delete DSHS.*  
From DSHS INNER JOIN DSDTHI ON DSHS.MAHS=DSDTHI.MAHS  
Where diem<5.;
```

3.8. Tạo mối quan hệ giữa các bảng

Muốn tạo một truy vấn để truy xuất dữ liệu từ 2 hay nhiều bảng thì phải tạo các mối quan hệ giữa các bảng đó.

Cú pháp

```
..... From <Table name 1> inner join <Table name 2> ON <Table name 1>.<Field  
name1>=<Table name 2>.<Field name 2>.....
```

Hoặc

```
.....From <Table name 1> Left join/ Right join <Table name 2> ON <Table name  
1>.<Field name1>=<Table name 2>.<Field name 2>.....
```

Ví dụ:

Tạo truy vấn gồm : Matruong, tentruong, hoten từ 2 bảng Dstruong và DSHS dựa vào trường liên kết Matruong.

```
Select Matruong, tentruong, hoten From Dstruong inner join dshs on  
Dstruong.matruong=dshs.matruong;
```

Ví dụ:

Cho 3 bảng dữ liệu Dstruong(**Matruong**, tentruong)

Dskhoa(Matruong, tenkhóa, **Makhóa**)

DSSV(Makhóa, Hoten, Ngaysinh, quequan)

Tạo một truy vấn để hiển thị danh sách sinh viên thuộc mỗi khoa của mỗi trường

Select Distinctrow

Matruong, tentruong, tenkhóa, hoten

*From Dstruong inner join (Dskhoa inner join DSSV ON Dskhoa.Makhóa=DSSV.Makhóa)
ON Dstruong.Matruong=Dskhoa.Matruong;*

☞ Chú ý

Có thể sử dụng liên kết ngoại trái Left join hoặc phải Right join

3.9. Truy vấn con (Sub query)

Truy vấn con là một mệnh đề Select.....From.....Wheres được lồng ghép vào một trong các mệnh đề sau:

Select.....From.....Where

Select.....Into.....

Insert.....Into.....

Delete.....

Update.....

Cú pháp

Select.....From.....Where.....

<Biểu thức so sánh> ANY|SOME|ALL <Mệnh đề truy vấn con>

<Biểu thức tìm kiếm> IN | NOT IN <Mệnh đề truy vấn con>

EXIST | NOT EXISTS <Mệnh đề truy vấn con>;

Chức năng: Tạo một truy vấn con

Giải thích các tham số

<Biểu thức so sánh>: Là một biểu thức và một phép toán so sánh.

<Biểu thức tìm kiếm>: Là một biểu thức mà tập hợp kết quả của truy vấn con sẽ được tìm kiếm.

<Mệnh đề truy vấn con>: Là dạng mệnh đề ở trong cú pháp và đặt giữa hai dấu ().

ANY, SOME: Các bản ghi trong truy vấn chính thoả mãn điều kiện so sánh với bất kỳ hoặc một vài các bản ghi nào truy xuất được từ truy vấn con.

ALL: Các bản ghi trong truy vấn chính thoả mãn với điều kiện so sánh với tất cả bản ghi nào truy xuất được từ truy vấn con.

IN: Các bản ghi trong truy vấn chính mà có tồn tại một vài bản ghi trong truy vấn con có giá trị bằng nó.

NOT IN: Các bản ghi trong truy vấn chính mà không tồn tại một vài bản ghi trong truy vấn con có giá trị bằng nó.

EXISTS (NOT EXISTS): Phép so sánh True/ False để xác định nhận truy vấn con có kết quả là bản ghi nào không.

Ví dụ:

Cho 2 bảng dữ liệu KHO(Mahang, Tenhang, Dongia)

NKBAN (Mahang, Hoten, Ngaymua, Giamgia, Dongia)

Tìm tất cả những mặt hàng mà đơn giá lớn hơn vài mặt hàng được bán với Giamgia là 20%.

*Select * From Kho*

Where dongia > ANY (select dongia From NKBAN Where giamgia=20%);

Tìm những mặt hàng bán ra với giảm giá >=10%.

*Select * from Kho*

Where Mahang IN (Select mahang From NKBAN Where giamgia>=0.1);

Ví dụ:

Cho 2 bảng danh sách DSKH(MAKHACH, HOTEN, QUEQUAN, SDT)

DATHANG(MAKHACH, SOLUONG, NGAYDAT)

Tìm những người khách đặt hàng trước 10/10/99 bao gồm Hoten, Quequan.

Select Hoten, quequan From DSKH

Where Makhach IN (Select makhach from DATHANG Where NGAYDAT<=#10/10/99#;.

3.10. Truy vấn hội (Union Query)

Dùng để nối (Kết hợp) dữ liệu các trường tương ứng từ 2 hay nhiều bảng hoặc truy vấn vào trường.

Cú pháp: *SelectFrom.....Where.....*

UNION | UNION ALL

Select.....

Chức năng: Tạo truy vấn hội

Giải thích:

UNION: Không muốn các bản ghi trùng nhau hiển thị

UNION ALL: Hiển thị các bản ghi trùng nhau

Ví dụ:

Cho 2 bảng dữ liệu DHSVIEN(Hoten, Lop, Matruong, Diachi)

CDHSVIEN(Hoten, Lop, Matruong, Diachi)

Sử dụng truy vấn hội để liệt kê Hoten, Lop, Matruong của sinh viên 2 hệ (Địa học, Cao đẳng).

Select Hoten, Lop, Matruong from DHSVIEN

UNION Select Hoten, Lop, Matruong from DHSVIEN

Liệt kê Hoten, Lop của những sinh viên 2 hệ và có quê quán ở Huế

Select Hoten, Lop from DHSVIEN

UNION Select Hoten, Lop from DHSVIEN Where Diachi= "Huế";

Chương 5

BIỂU MẪU

Từ trước đến nay chúng ta vẫn làm việc một cách đơn điệu với các bảng, truy vấn với cách trình bày dữ liệu hiệu quả nhưng không đẹp mắt. Với biểu mẫu (form) trong Access sẽ giúp chúng ta khắc phục điều này. Biểu mẫu trong Access rất linh động, chúng ta có thể dùng biểu mẫu để nhập, xem, hiệu chỉnh dữ liệu. Hoặc là dùng biểu mẫu để tạo ra các bảng chọn công việc làm cho công việc của chúng ta thuận lợi và khoa học hơn. Hoặc dùng biểu mẫu để tạo ra các hộp thoại nhằm thiết lập các tùy chọn cho công việc quản lý của mình.

1. KHÁI NIỆM VỀ BIỂU MẪU

Nếu chúng ta đã quen điền các tờ biểu, mẫu trong cuộc sống hàng ngày thì chúng ta có thể hình dung một biểu mẫu trong Access cũng vậy. Một biểu mẫu trong Access định nghĩa một tập dữ liệu chúng ta muốn lấy và từ đó đưa vào CSDL. Cũng vậy biểu mẫu cũng có thể dùng để xem xét dữ liệu hay in ra máy in.

Trong môi trường của Hệ QTCSDL Access chúng ta có thể thiết kế các biểu mẫu có hình thức trình bày đẹp, dễ sử dụng và thể hiện đúng các thông tin cần thiết. Chúng ta có thể đưa vào biểu mẫu các đối tượng như văn bản, hình ảnh, đường vẽ kết hợp với các màu sắc sao cho biểu mẫu của chúng ta đạt được nội dung và hình thức trình bày ưng ý nhất. Hình thức và cách bố trí các đối tượng ra sao trên biểu mẫu hoàn toàn tùy thuộc vào khả năng thẩm mỹ và ng khiếu trình bày của chúng ta.

2. TÁC DỤNG VÀ KẾT CẤU CỦA BIỂU MẪU

2.1. Tác dụng của biểu mẫu

Biểu mẫu cung cấp một khả năng thuận lợi để hiển thị dữ liệu. Chúng ta có thể xem mọi thông tin của một bản ghi thay vì ở chế độ Datasheet nghèo nàn trước đây bằng chế độ Form View, một phương cách tiên tiến hơn.

Sử dụng biểu mẫu tăng khả năng nhập dữ liệu, tiết kiệm thời gian và ngăn ngừa các lỗi do đánh sai. Chẳng hạn thay vì gõ vào các giá trị của tất cả các trường chúng ta có thể tạo những danh sách (gọi là combo box) để chọn trên biểu mẫu (đây là phương cách áp dụng rất hiệu quả để tránh đánh sai dữ liệu).

Biểu mẫu cung cấp một hình thức trình bày hết sức tiện nghi để xem, nhập và hiệu chỉnh các bản ghi trong CSDL. Access cung cấp các công cụ thiết kế biểu mẫu hỗ trợ rất đặc lực cho chúng ta trong việc thiết kế những biểu mẫu để sử dụng mà lại có thể tận dụng được các khả năng:

Hình thức thể hiện dữ liệu đẹp, trình bày lôi cuốn với các kiểu font và hiệu ứng đồ họa đặc biệt khác ...

Quen thuộc với người sử dụng vì nó giống các biểu mẫu trên giấy thông thường.

Có thể tính toán được.

Có thể chứa cả biểu đồ.

Có thể hiển thị dữ liệu từ nhiều bảng (hoặc truy vấn)

Tự động hóa một số thao tác phải làm thường xuyên.

2.2. Kết cấu của biểu mẫu

Các thông tin trên biểu mẫu có thể lấy dữ liệu từ một bảng hay truy vấn nào đó, nhưng cũng có thể độc lập đối với cả bảng lẫn truy vấn, chẳng hạn như các đối tượng đồ họa. Dáng vẻ trình bày của biểu mẫu được thực hiện trong quá trình thiết kế.

Tất cả các thông tin thể hiện trên biểu mẫu được chứa trong những đối tượng gọi là *điều khiển* (control). Điều khiển có thể dùng để thể hiện dữ liệu hoặc thực hiện các hàng động hoặc trang trí cho biểu mẫu.

Một số điều khiển được buộc vào với các trường của bảng hay truy vấn, gọi là bảng cơ sở hay truy vấn cơ sở. Do đó chúng ta có thể dùng biểu mẫu để nhập dữ liệu vào

các trường hay lấy dữ liệu từ các trường đó ra để xem. Ví dụ dùng Text box để nhập hay hiển thị chuỗi và số, dùng Object frame để thể hiện hình ảnh.

Một số điều khiển khác trình bày thông tin được lưu trữ trong thiết kế bảng. Ví dụ dùng Label (nhãn) để thể hiện thông tin có tính chất mô tả; đường và các hình khối để tổ chức dữ liệu và làm biểu mẫu có hình thức hấp dẫn hơn.

3. TẠO BIỂU MẪU

3.1. Tạo biểu mẫu tự động với Autoform

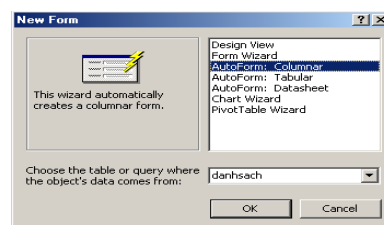
Access cung cấp chức năng Autoform cho phép chúng ta tạo biểu mẫu dựa trên các bảng hoặc truy vấn đã được xây dựng trước đó.

Cách tạo

Trong cửa sổ Database, chọn form, chọn New

Trong mục *Choose the table or query Where the object's data comes from:*

Chọn bảng hoặc truy vấn làm nguồn dữ liệu cho form.



Chọn **Autoform Columnar**: Nếu muốn tạo lập biểu mẫu dạng cột, trong đó mỗi trường trong bảng hay truy vấn là một dòng.

Chọn **Autoform Tabular**: Nếu muốn tạo lập biểu mẫu dạng hàng, trong đó mỗi trường trong bảng hay truy vấn là một cột và một bản ghi trong một dòng.

Chọn **Autoform Datasheet**: Nếu muốn tạo lập biểu mẫu theo dạng bảng, trong đó mỗi cột tương ứng một trường và mỗi dòng là một bản ghi.

Chọn OK.

Ví dụ

Cho bảng Danhsach(Tenphong, hoten, ngayden, ngaydi, thanh tien)

Hãy thiết kế biểu mẫu dựa trên chức năng Autoform sử dụng bảng Danhsach làm nguồn dữ liệu.

Biểu mẫu dạng **Autoform Columnar**

Biểu mẫu dạng **Autoform tabular**

| | tenphong | Hoten | ngayden | ngaydi | thanh tien |
|----|-------------|-------|----------|----------|------------|
| T1 | Thanh Loan | | 10/10/90 | 20/10/90 | 1000000 |
| T1 | Thuy Phuong | | 11/11/91 | 22/11/91 | 3000000 |
| T1 | Tran Sau | | 02/02/98 | 20/02/98 | 3000000 |
| T2 | Thanh Thanh | | 01/01/98 | 20/10/98 | 8000000 |
| T2 | Le Lai | | 11/02/99 | 22/02/99 | 1000000 |

Biểu mẫu dạng **Autoform Datasheet**

3.2. Tạo biểu mẫu sử dụng Wizard

Tạo biểu mẫu sử dụng công cụ Autoform

thì Access không cho phép người sử dụng can thiệp

vào quá trình tạo biểu mẫu, chẳng hạn như hạn chế số trường..... thì Form Wizard cho phép người sử dụng can thiệp vào quá trình tạo biểu mẫu.

Cách tạo

Trong cửa sổ Database chọn Form, chọn New

Trong mục *Choose the table or query Where the object's data comes from:*

Chọn bảng hoặc truy vấn làm nguồn dữ liệu cho form.

Chọn Form Wizard

Chọn OK

Trong mục Available Field: Chọn các trường đưa vào biểu mẫu, nhấn nút >>

Chọn nút **Next**.

Chọn **Columnar** : Biểu mẫu hiển thị theo dạng cột

Tabular : Biểu mẫu hiển thị theo dạng hàng

Datasheet : Biểu mẫu hiển thị theo dạng bảng

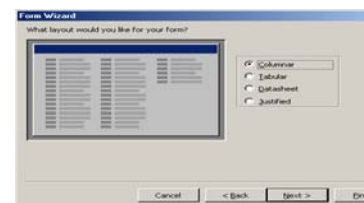
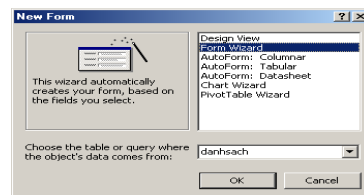
Justified : Biểu mẫu hiển bình thường (đều).

Chọn **Next**

Chọn loại biểu mẫu

Chọn **Next**

Đặt tiêu đề cho Form



Chọn *Open the form to view or*

enter information: Nếu muốn mở Form sau khi chọn Finish.

Chọn *Modify the form's design*: Nếu muốn form ở dạng thiết kế.

Chọn **Finish** .

Lưu form.

3.3. Tạo biểu mẫu không sử dụng Wizard (Do người sử dụng tự thiết kế)

Tạo biểu mẫu sử dụng công cụ Autoform và Form wizard người sử dụng có thể nhanh chóng thiết kế các biểu mẫu nhờ vào các đặc tính hỗ trợ của Access. Nhưng đối với hai cách trên chỉ cung cấp một số hạn chế các phương án xây dựng biểu mẫu mà không thoả mãn yêu cầu của người sử dụng khi muốn thiết kế biểu mẫu theo ý của riêng mình. Do đó người sử dụng phải tự thiết kế một biểu mẫu không cần sự hỗ trợ của Access.

Cách tạo

Trong cửa sổ Database chọn Form, chọn New

Chọn bảng dữ liệu hoặc truy vấn làm nguồn

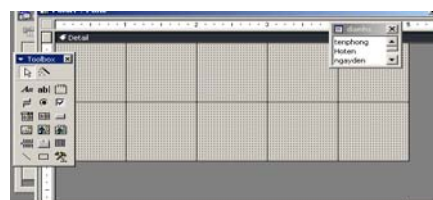
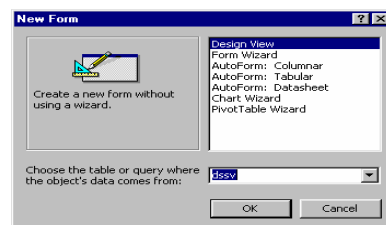
dữ liệu cho form, chọn **OK**

Xây dựng các điều khiển cho biểu mẫu

(Đưa các trường trong bảng dữ liệu vào biểu mẫu).

Thiết lập các thuộc tính cho các điều khiển.

Lưu biểu mẫu.



4. CÁC CHẾ ĐỘ HIỂN THỊ VÀ CÁC THÀNH PHẦN CỦA BIỂU MẪU

4.1. Các chế độ hiển thị

Có 4 chế độ hiển thị của biểu mẫu

4.1.1. Chế độ *Design View*

Dùng để tạo biểu mẫu mới hay thay đổi cấu trúc của một biểu mẫu đã tồn tại.

Khi đang ở trong cửa sổ Database: Chọn form /chọn tên form/ chọn Design (Có

thể click chuột phải rồi chọn Design).

Khi đang ở chế độ Form view: Chọn View/ Form Design

4.1.2. Chế độ Form View

Dùng để nhập, thay đổi và xem dữ liệu. Trong chế độ Form View người sử dụng có thể xem tất cả các trường của một bản ghi tại một thời điểm.

Khi đang ở trong cửa sổ Database: Chọn form /chọn tên form/ chọn Open (Có thể click chuột phải rồi chọn Open).

Khi đang ở chế độ Design view: Chọn View/ Form View.

4.1.3. Chế độ hiển thị biểu mẫu dưới dạng bảng (Datasheet View)

Dùng để nhập, thay đổi và xem dữ liệu trong biểu mẫu theo dạng bảng biểu.

Để mở chế độ hiển thị dạng Datasheet View chúng ta thực hiện như sau:

Khi đang ở chế độ Design View: Chọn View/Datasheet.

Khi đang ở chế độ Form View: Chọn View/ Datasheet View.

4.1.4. Chế độ hiển thị Print Preview

Dùng để xem biểu mẫu trước khi quyết định in ấn. Trong chế độ Print Preview sẽ duy trì hình dạng trình bày dữ liệu đã được thiết kế trước đó.

Khi đang ở trong cửa sổ Database: Chọn form /chọn tên form/ Chọn File/Print Preview.

4.2. Các thành phần biểu mẫu trong chế độ Design View

Khi muốn thiết kế biểu mẫu thì người sử dụng phải làm việc trong chế độ Design View khi đó biểu mẫu có các thành phần chính sau:

Thước(Ruler): Điều chỉnh kích thước của các điều khiển.

Tiêu đề form (form header):Sử dụng để trình bày tiêu đề của form, tiêu đề form luôn được trình bày phần trên cùng, đầu tiên của biểu mẫu và trang in biểu mẫu.

Chân form (Form Footer): Sử dụng để trình bày chân của form, chân form luôn được trình bày phần dưới cùng, xuất hiện cuối biểu mẫu và trang in biểu mẫu.

Tiêu đề trang (Page header): Sử dụng để chứa tiêu đề trang

Chân trang (Page footer): Sử dụng để chứa chân trang nhưng xuất hiện phần trước của Form footer trong trang biểu mẫu in.

Chú ý

Page header và Page footer chỉ xuất hiện trong trang biểu mẫu in nên chúng không có những tính chất thông thường như Form header và Form footer.

Chọn View/ Page header/ footer (Nếu 2 thành phần này chưa xuất hiện trên biểu mẫu).

Chi tiết form (Detail): Đây là phần rất quan trọng chứa các điều khiển nhằm trình bày các dạng dữ liệu từ các bảng dữ liệu hoặc các truy vấn. Các loại điều khiển có thể là điều khiển buộc, không buộc hoặc tính toán.

5. CÁC LOẠI ĐIỀU KHIỂN

Tất cả thông tin trên biểu mẫu được chứa trong những đối tượng gọi là điều khiển (Control). Điều khiển có thể dùng để thể hiện dữ liệu, thực hiện các hành động hoặc thiết kế biểu mẫu đẹp mắt. Trong ACCESS hệ thống định nghĩa một số loại điều khiển như sau:

Điều khiển nhãn (Label).

Điều khiển hộp văn bản (Text box).

Điều khiển nhóm lựa chọn (Option group).

Điều khiển loại hộp Combo (Combo box) và hộp danh sách (List Box).

Ngoài ra còn có một số điều khiển khác như command button.....

Khi tạo lập điều khiển, chúng ta thường xác định hình thức dữ liệu trình bày trong chúng. Có những điều khiển lấy dữ liệu từ các trường trong bảng hay truy vấn, có điều khiển chỉ dùng vào mục đích trang trí, làm tiêu đề, có những điều khiển lấy dữ liệu từ một biểu thức nào đó. Vì vậy người ta phân ra thành ba nhóm điều khiển chính:

Điều khiển buộc (Bound control)

Điều khiển không buộc (Unbound control)

Điều khiển tính toán (Calculated control)

5.1. Điều khiển bị buộc (bound), không buộc (unbound) và tính toán được (calculated).

Khi tạo một điều khiển trong biểu mẫu thì phải xác định nó lấy dữ liệu từ nguồn nào để thể hiện.

Ví dụ

Tạo một điều khiển loại hộp văn bản (Text box) để hiển thị tên các mặt hàng, chúng ta phải chỉ định cho điều khiển lấy dữ liệu trong trường TEN_HANG của bảng MAT_HANG. Hộp văn bản này gọi là bị buộc.

Điều khiển cũng có thể thể hiện những thông tin không có trong CSDL (Không bị buộc).

Ví dụ: Tạo tiêu đề cho biểu mẫu...

Tóm lại

Điều khiển bị buộc (Bound Control) là điều khiển mà nguồn dữ liệu của nó lấy từ một trường trong bảng hoặc truy vấn. Trong biểu mẫu dùng điều khiển buộc vào các trường để hiển thị nội dung hoặc cập nhật các trường của CSDL, các giá trị cập nhật có thể là: Văn bản, Date, Number, yes/No, Picture, chart trong đó dạng văn bản là phổ biến nhất.

Điều khiển không bị buộc (Unbound Control) là điều khiển không lấy dữ liệu từ một nguồn nào cả là điều khiển không bị buộc. Dùng điều khiển không buộc để trình bày thông tin không có trong các bảng hay rút được từ truy vấn.

Điều khiển tính toán (Calculated Control) là điều khiển mà nguồn dữ liệu của nó không phải là một trường mà là một biểu thức gọi là điều khiển tính toán (Calculated Control). Chúng ta qui định giá trị xuất hiện trong điều khiển bằng cách lập biểu thức cho nó. Biểu thức này là nguồn dữ liệu của điều khiển. Trong biểu thức có thể dùng các toán tử (+, -, =, ...) với tên điều khiển.

Ví dụ: Tạo một điều khiển $THANH TIEN = SOLUONG * DONGIA$

5.2. Tạo điều khiển loại hộp văn bản (text box)

Text box có thể là một điều khiển bị buộc, không buộc hoặc tính toán.

Tạo hộp văn bản bị buộc (Bound Text box).

Chúng ta buộc điều khiển Text box vào một trường bằng cách chỉ định điều khiển đó lấy dữ liệu trên trường nào. Chọn trường để buộc vào điều khiển bằng cách Click biểu

tượng Field List để mở danh sách các trường của bảng hay truy vấn làm nền tảng cho biểu mẫu. Theo mặc định như vậy thì hệ thống sẽ tạo một điều khiển loại Text box.

Một cách khác dùng hộp dụng cụ Toolbox để tạo điều khiển và sau đó gõ tên trường muốn buộc vào hộp văn bản.

Mở hộp danh sách trường

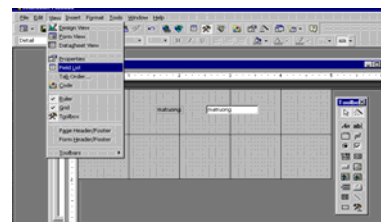
Mở biểu mẫu trong chế độ Design (Tạo biểu mẫu trước).

Trong cửa sổ Database Chọn Form/New (Cộn bảng hoặc truy vấn làm nền cho biểu mẫu).

Chọn View/Field List hoặc click vào biểu tượng Field List trên thanh công cụ.

Tạo một Text Box bị buộc

Từ danh sách trường chọn một hoặc nhiều trường kéo và đưa vào biểu mẫu.



5.3. Tạo một điều khiển khác dùng hộp công cụ

Dùng hộp công cụ (Toolbox) để tạo những điều khiển không buộc (Unbound control) hoặc để tính toán. Đặc biệt dùng các tính năng của hộp này để tạo những điều khiển bị buộc khác ngoài buộc khác ngoài (Text box). Để bật hộp công cụ ta chọn View/Toolbars.

Tạo điều khiển dùng hộp công cụ

Click vào biểu tượng công cụ tương ứng với điều khiển muốn tạo.

Tạo điều khiển bị buộc bằng cách chọn một trường trong danh sách trường (Field List) và kéo vào biểu mẫu.

Hoặc tạo điều khiển không buộc hay dùng để tính toán bằng cách click vào một vị trí trên biểu mẫu.

Tạo điều khiển dùng để tính toán

Nếu muốn trình bày kết quả của một phép toán trong biểu mẫu, khai báo nguồn dữ liệu của điều khiển là một biểu thức. Sau này mỗi lần mở biểu mẫu, ACCESS tính toán lại kết quả của vùng dữ liệu được cập nhật mới nhất từ các bảng.

Tạo điều khiển nhãn (Label Control)

Muốn trình bày một chuỗi ký tự trên biểu mẫu như để làm tiêu đề... chúng ta dùng loại điều khiển gọi là điều khiển nhãn (label control). Nhãn không thể hiện dữ liệu của một trường hay biểu thức nào, chúng luôn luôn là không bị buộc.

Click vào biểu tượng Label trong Toolbox

Click vị trí muốn đặt nhãn trên biểu mẫu.

Thay đổi các thuộc tính của điều khiển.

Sau khi tạo biểu mẫu, biến đổi một số thuộc tính của điều khiển có thể hoàn thiện thêm về thiết kế và hình thức trình bày của số liệu.

Nhấp kép vào điều khiển.

Trong hộp lựa chọn trên đầu bảng thuộc tính, chọn một lớp thích hợp để làm việc.

All Properties: Trình bày tất cả các thuộc tính của điều khiển

Data Properties: Ấn định các đặc tính thể hiện dữ liệu trong điều khiển như giá trị mặc nhiên, định dạng số.

Even Properties: Qui định một tập lệnh (Macro) hay thủ tục (Procedure)..

Layout Properties: Định nghĩa các hình thức của điều khiển như cao, rộng.

Other Properties: Một số thuộc tính khác như tên điều khiển, thông tin mô tả ở dòng trạng thái.

Click chọn một trong các thuộc tính của bảng để thực hiện.

5.4. Thiết lập một số thuộc tính bổ sung

5.4.1. Các thuộc tính hỗ trợ nhập liệu

Thuộc tính **Default value**: Gán giá trị mặc định vào nội dung trình bày trong điều khiển.

Thuộc tính **ValidationRule** và **ValidationRule Text**: Kiểm tra tính hợp lệ khi nhập dữ liệu cho một điều khiển và thông báo lỗi nếu dữ liệu không hợp lệ

5.4.2. Các thuộc tính giống lề

Thuộc tính **General**: Giống hàng văn bản theo lề trái, dữ liệu số và ngày tháng theo lề phải.

Thuộc tính **Left**: Giống hàng văn bản theo lề trái.

Thuộc tính **Center**: Xác lập hàng văn bản ở chính giữa.

Thuộc tính **Center**: Gióng hàng văn bản theo lề phải.

5.4.3. Các thuộc tính màu sắc

Thuộc tính **BackColor**: Thiết lập màu nền cho điều khiển hay cho biểu mẫu

Thuộc tính **ForeColor**: Thiết lập màu cho hàng chữ trong điều khiển.

Thuộc tính **Bodercolor**: Thiết lập màu cho khung bao quanh trong điều khiển.

5.4.3. Các thuộc tính khung bao

Thuộc tính **BoderStyle**: Thiết lập loại khung cho điều khiển

Thuộc tính **BoderWidth**: Thiết lập độ dày hay đậm của khung bao.

Thuộc tính **BoderColor**: Thiết lập màu của khung bao.

6. NÂNG CẤP BIỂU MẪU

Access cung cấp nhiều loại điều khiển để làm cho biểu mẫu dễ sử dụng và có nhiều hiệu ứng tốt. Có thể thay thế một Textbox với một List box hay Combo box để chọn từ các giá trị có sẵn thay vì buộc người sử dụng phải nhớ để nhập giá trị vào.

6.1. Dùng điều khiển List box và Combo box để tạo danh sách chọn lựa.

Trong nhiều trường hợp, chọn một danh sách có sẵn thường tiện lợi hơn phải nhớ để gõ vào từ bàn phím. Access cung cấp hai khả năng điều khiển tạo danh sách chọn lựa: List box và Combo box.

List box đơn giản là một danh sách để chọn, combo box tương tự như một text box và một combo box kết hợp vào một điều khiển, có nghĩa là có thể gõ thẳng giá trị vào text box hay chọn từ một danh sách có sẵn.

Ưu điểm của List box: Danh sách luôn được thể hiện và người dùng chỉ được phép chọn trong danh sách, do đó dữ liệu nhập luôn luôn là hợp lệ.

Ưu điểm của Combo box: Danh sách không được thể hiện cho đến khi người dùng mở hộp điều khiển, do đó ít tốn chỗ trên biểu mẫu hơn.

6.2. Tạo List box và Combo box không sử dụng Wizard

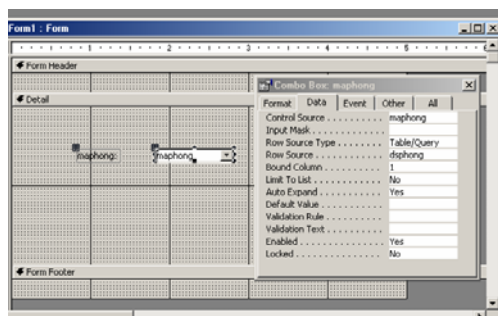
Tắt chức năng Control Wizard trong menu View hoặc trong thanh công cụ.

Click biểu tượng List box hoặc Combo box trong hộp công cụ.

Tạo điều khiển buộc bằng cách click biểu tượng Field list trên thanh công cụ để mở danh sách trường làm nền cho biểu mẫu. Chọn một trường trong danh sách kéo vào biểu mẫu, hoặc tạo điều khiển không buộc bằng cách click biểu mẫu nơi muốn đặt điều khiển.

Lập các thuộc tính của điều khiển để định nghĩa các hàng dùng làm chọn lựa trong danh sách.

| Muốn danh sách thể hiện | Lập thuộc tính Row Source Type thành | Lập Row Source thành |
|---|---|---|
| Các hàng từ một bảng hay truy vấn | Table/Query (Default) | Tên của bảng hay truy vấn đó |
| Các hàng lấy từ lệnh Select của SQL | Table/Query (Default) | Một câu lệnh SQL |
| Một danh sách với các giá trị do người dùng tự đặt | Value List | Danh sách các giá trị đó phân cách nhau bởi dấu chấm phẩy |
| Tên các trường trong một bảng hoặc truy vấn | Field List | Tên của bảng hoặc truy vấn đó |
| Các giá trị định nghĩa bởi một hàm trong Access Basic | Tên hàm xây dựng | Để trống |



6.2.1. Định nghĩa các hàng trong danh sách

Định nghĩa nguồn dữ liệu để hình thành các hàng dùng làm trong danh sách lựa chọn trong List box và Combo box bằng cách lập thuộc tính thích hợp cho RowSource Type và RowSource của điều khiển, hai thuộc tính này phối hợp với nhau để tạo nên các hàng trong danh sách.

Thuộc tính RowSource Type cho biết danh sách sẽ hình thành từ nguồn dữ liệu loại nào, thuộc tính RowSource là thuộc tính xác định nguồn dữ liệu của danh sách.

6.2.2. Danh sách lấy dữ liệu từ bảng hay truy vấn

Có thể tạo List box hay Combo box lấy dữ liệu từ một hoặc nhiều trường trong bảng (hoặc truy vấn) để lập thành danh sách chọn lựa cho điều khiển. Thực hiện điều này bằng cách đặt thuộc tính RowSource Type thành Table/Query và RowSource thành tên bảng hay tên truy vấn có chứa dữ liệu để lập thành danh sách đó.

6.2.3. Tạo danh sách nhiều cột

Có thể tạo danh sách gồm hai hay nhiều cột để có thêm thông tin lựa chọn. Xác định số cột bằng cách lập thuộc tính ColumnCount và ColumnWidth cho điều khiển. Thuộc tính columnCount quy định số cột và nếu trong RowSource là tên một bảng hay truy vấn thì các trường đầu tiên tương ứng số cột của bảng hay truy vấn đó (tính từ trái sang phải) sẽ được đưa vào danh sách. Thuộc tính columnwidth quy định bề rộng mỗi cột tính theo inch hay cm.

6.2.4. Sử dụng danh sách giá trị

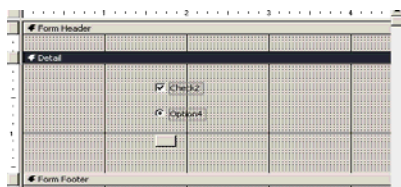
Trường hợp điều khiển chỉ có một số nhỏ lựa chọn và các giá trị này không thay đổi, có thể dùng phương pháp đơn giản là danh sách giá trị. Danh sách này chúng ta tự lập bằng cách đưa các giá trị dùng làm lựa chọn vào thuộc tính RowSource của điều khiển. Sử dụng dấu chấm phẩy để phân cách các mục lựa chọn của danh sách.

6.3. Sử dụng hộp kiểm tra (check box), nút chọn lựa (Option button), nút bật tắt (Toggle button).

Hộp kiểm tra, nút chọn lựa, nút bật tắt thường được sử dụng như các điều khiển độc lập để nhận các lựa chọn Yes/No. Các điều khiển này thực chất chỉ khác nhau về hình thức, do đó chúng ta có thể sử dụng bất cứ nút nào.

Khi được chọn điều khiển này biểu diễn trị Yes (hay True), ở trạng thái không chọn thì nhận giá trị No (hay False).

- Check box** : Được chọn khi có dấu x trong hộp.
- Option button** : Được chọn khi có hạt đậu trong nút.
- Toggle button** : Được chọn khi nó trông như bị nhấn xuống.



6.4. Sử dụng nhóm chọn lựa (Option group)

Chúng ta có thể dùng nhóm chọn lựa để trình bày một số nhỏ các chọn lựa trên biểu mẫu. Nhóm chọn lựa gồm một khung nhóm (Group frame) có gắn nhãn và một số check box, option button hay toggle button.



Nếu nhóm lựa chọn được buộc vào một trường, chỉ có khung nhóm là bị buộc vào trường đó, không phải các thành phần để chọn lựa trong khung. Khung nhóm có thể không bị buộc vào trường nào hoặc chứa một biểu thức, mỗi lần chỉ có thể chọn một thành phần trong nhóm.

Khi sử dụng hộp kiểm tra, nút chọn lựa và nút bật tắt trong nhóm chọn lựa, thuộc tính của chúng khác với như khi được dùng với các điều khiển độc lập. Mỗi phần tử trong nhóm chọn lựa không có thuộc tính Control Source mà thay bằng thuộc tính Option Value. Lập thuộc tính Option Value của mỗi phần tử dùng làm chọn lựa trong nhóm thành một trị số có nghĩa đối với trường mà khung nhóm được buộc vào. Khi một phần tử của nhóm được chọn, Access đưa giá trị lập trong Option value của phần tử đó cho điều khiển Option Group. Đến lượt Option Group trả lại giá trị đó cho trường mà nó bị buộc vào.

7. BIỂU MẪU DỰA TRÊN NHIỀU BẢNG

7.1. Biểu mẫu phụ (Subform)

Biểu mẫu phụ là phương pháp để đưa thông tin từ nhiều bảng vào một biểu mẫu. Biểu mẫu phụ có nghĩa là một biểu mẫu được lồng trong biểu mẫu khác. Trong Access biểu mẫu chính gọi là Main form, biểu mẫu nằm trong Main form gọi là biểu mẫu phụ (Sub form). Khi dùng biểu mẫu phụ chúng ta dễ nhận thấy mối quan hệ giữa các bản ghi của hai hay nhiều bảng.

Biểu mẫu phụ đặc biệt hữu hiệu khi dùng để hiển thị dữ liệu từ nhiều bảng hay truy vấn có quan hệ một-nhiều với nhau. Biểu mẫu chính đại diện cho bên một, biểu mẫu phụ đại diện cho bên nhiều.

7.2. Các loại biểu mẫu phụ

Khi tạo biểu mẫu phụ chúng ta có thể thiết kế nó thành dạng bảng, hoặc dạng biểu mẫu, hoặc cả hai dạng trên.

Biểu mẫu dạng bảng: Là loại dễ tạo nhất và có thể sử dụng như bất kỳ bảng nào khác như sắp xếp....

Biểu mẫu phụ dạng biểu mẫu: Cho chúng ta thực sự linh hoạt và mềm dẻo khi thiết kế hơn.

7.3. Thiết kế biểu mẫu phụ

Thông thường chúng ta dùng bảng hay truy vấn làm nguồn dữ liệu cho biểu mẫu chính, một bảng hay truy vấn khác làm nguồn dữ liệu cho biểu mẫu phụ. Nếu dữ liệu trong biểu mẫu chính và biểu mẫu phụ có liên quan với nhau, chúng ta cần đánh giá một số vấn đề sau:

Các bảng hoặc truy vấn có quan hệ một-nhiều với nhau không? Nếu dùng biểu mẫu phụ để thể hiện quan hệ một-nhiều, chúng ta nên dùng bảng bên một đối với bảng chính, bảng bên nhiều đối với bảng phụ.

Các bảng hoặc truy vấn làm nguồn dữ liệu cho biểu mẫu chính/phụ có các trường liên kết không? Access dùng trường kết nối để giới hạn số lượng bản ghi thể hiện trong biểu mẫu phụ.

Cách tạo biểu mẫu chính/phụ

Thiết kế hai biểu mẫu riêng biệt, sau đó kéo biểu mẫu phụ vào biểu mẫu chính.

Thiết kế biểu mẫu chính

Tạo biểu mẫu chính, dành chỗ trên biểu mẫu này để chứa biểu mẫu phụ.

Lưu và đóng biểu mẫu chính.

Thiết kế biểu mẫu phụ

Có thể thiết kế biểu mẫu phụ để chỉ thể hiện dữ liệu dưới dạng bảng, biểu mẫu này có cả hai khả năng trên.

Tạo biểu mẫu mới, lập hai thuộc tính ViewAllowed và Default View của biểu mẫu tùy theo yêu cầu sử dụng như sau:

Biểu mẫu phụ chỉ trình bày dưới dạng bảng: Đặt các trường trên biểu mẫu theo thứ tự chúng ta muốn chúng xuất hiện trong bảng. Lập cả hai thuộc tính ViewAllowed và Default View thành Datasheet.

Biểu mẫu phụ chỉ trình bày dữ liệu dưới dạng biểu mẫu: Sắp đặt các điều khiển như trên. Lập thuộc tính ViewAllowed thành Form và Default View thành Single form hay Continuous form.

Đưa biểu mẫu phụ vào biểu mẫu chính

Mở biểu mẫu chính ở chế độ Design View

Chuyển sang cửa sổ Database, nhấn F11

Kéo biểu mẫu phụ từ cửa sổ Database và đặt vào một vị trí trên biểu mẫu chính.

Di chuyển biểu mẫu phụ đến vị trí khác, thay đổi nội dung nhãn hoặc kích thước nếu cần. Chuyển sang chế độ Form View để xem kết quả.

| manvien | hotennv | nsinh | gioitinh | quequar |
|---------|-----------|----------|-------------------------------------|---------|
| A001 | Nguyen An | 10-10-76 | <input checked="" type="checkbox"/> | Hue |
| A002 | Le Lai | 02-02-76 | <input type="checkbox"/> | Ha Noi |

7.4. Liên kết biểu mẫu chính và biểu mẫu phụ

Trong chế độ Design View của biểu mẫu chính, mở bảng thuộc tính của điều khiển biểu mẫu phụ.

Lập thuộc tính LinkChildFields thành tên trường nối kết trong biểu mẫu phụ, nếu có nhiều trường nối kết, phân cách nhau bởi dấu phẩy.

Lập thuộc tính LinkMasterFields thành tên trường nối kết hoặc tên điều khiển trong biểu mẫu chính, nếu có nhiều trường nối kết, phân cách nhau bởi dấu phẩy.

1. TỔNG QUAN VỀ BÁO CÁO

Báo cáo là phương thức hữu hiệu giúp người sử dụng trình bày dữ liệu dưới dạng đầy đủ và dễ hiểu, nhanh chóng, đẹp mắt để khi in ấn. Người sử dụng có thể tích hợp trong báo cáo các dạng thức trình bày dữ liệu khác nhau như: Hình ảnh, biểu đồ, văn bản.....

Báo cáo được xây dựng trên một nguồn dữ liệu đó là bảng hoặc truy vấn, một câu lệnh SQL hoặc một dạng biểu mẫu nào đó

1.1. Các dạng mẫu của báo cáo

Báo cáo dạng cột (columnar): báo cáo dạng này sẽ được trình bày theo dạng một cột và kèm theo phần nhãn của mỗi cột dữ liệu bên trái, mỗi dòng tương ứng với một trường dữ liệu.

Báo cáo dạng hàng (Tabular): Báo cáo sẽ trình bày dữ liệu theo dạng bảng bao gồm nhiều hàng và nhiều cột.

Báo cáo dạng nhóm/ Tổng (Group/Total): Báo cáo dạng này sẽ tổ chức dữ liệu thành các nhóm, mỗi nhóm sẽ trình bày dữ liệu theo dạng Tabular. Người sử dụng có thể nhóm dữ liệu theo cấp và có thể tính toán giá trị tổng cho mỗi nhóm và một giá trị tính tổng cho toàn bộ các nhóm

Báo cáo dạng biểu đồ (Chart)

Báo cáo dạng nhãn (Label Report)

Báo cáo với báo cáo con

1.2. Các chế độ hiển thị của báo cáo

Báo cáo có thể được trình bày theo 3 chế độ sau

Report design: Chế độ thiết kế báo cáo.

Layout PreView: Chế độ trình bày dữ liệu trong báo cáo.

Print PreView: Chế độ xem hình thức báo cáo trước khi in ấn.

2. TẠO BÁO CÁO SỬ DỤNG CÔNG CỤ AUTO REPORT VÀ REPORT WIZARD

2.1. Tạo báo cáo sử dụng Auto report

Click biểu tượng Report trong cửa sổ database (hoặc chọn View/Report)

Chọn New

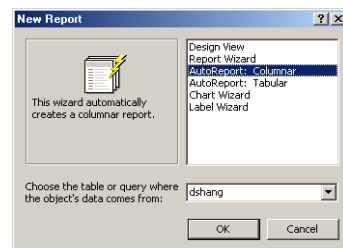
Chọn bảng hoặc Truy vấn làm nguồn dữ liệu báo cáo .

Chọn AutoReport Columnar: Nếu muốn báo cáo hiển thị dạng cột.

AutoReport Tabular: Nếu muốn báo cáo hiển thị dạng hàng

Chọn OK

Lưu Báo cáo.



2.2. Tạo báo cáo sử dụng Report Wizard

Click biểu tượng Report trong cửa sổ database (hoặc chọn View/Report)

Chọn New

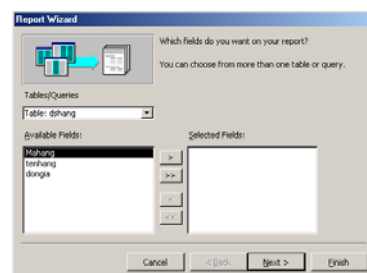
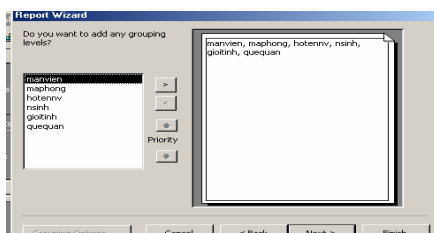
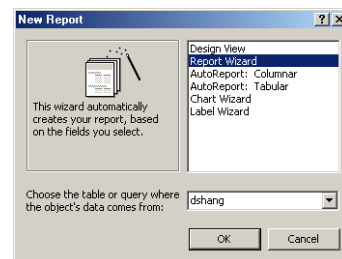
Chọn bảng hoặc Truy vấn làm nguồn dữ liệu báo cáo .

Chọn Report Wizard

Chọn OK

Chọn các trường cần thiết cho báo cáo.

Chọn Next



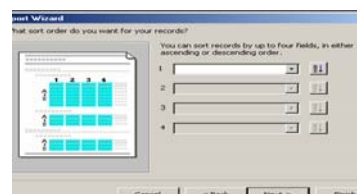
Chọn các trường cần nhóm, chọn Next

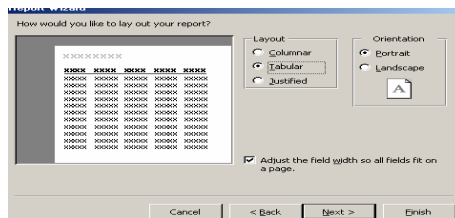
Chọn các trường cần sắp xếp, chọn Next

Chọn dạng thể hiện của Report, chọn Next

Chọn nền thể hiện của Report, chọn Next

Đặt tiêu đề cho Report, chọn Finish





3. TẠO ĐIỀU KHIỂN TRONG BÁO CÁO

3.1. Tạo điều khiển Text Box (Hộp văn bản)

Text box có nhiều tính năng, có thể là điều khiển bị buộc, không buộc hay dùng tính toán. Nếu trong báo cáo có cả 3 loại điều khiển này, chúng ta nên ưu tiên điều khiển bị buộc trước.

Tạo một text box bị buộc và không buộc

Chúng ta buộc điều khiển Text box vào một trường bằng cách chỉ định trường cho text box đó lấy dữ liệu. Text box đó lấy dữ liệu có thể thực hiện bằng cách kéo trường muốn buộc vào điều khiển từ danh sách trường (Field List) vào biểu mẫu đang thiết kế.

Cách khác để tạo Text box là dùng hộp công cụ (Toolbox), sau đó gõ tên trường muốn buộc vào hộp văn bản hoặc bảng thuộc tính của điều khiển.

Dùng danh sách trường là phương pháp tốt nhất để tạo một điều khiển Text box bị buộc vì hai lý do sau:

Điều khiển đó được hệ thống tự động gắn nhãn và nhãn lấy tên trường được kéo làm tiêu đề.

Text box bị buộc đó thừa kế các thiết lập thuộc tính của trường từ bảng hay truy vấn.

Muốn chuyển một điều khiển không buộc thành bị buộc, lập thuộc tính Control Source của điều khiển thành một trường.

Một số thao tác khi thực hiện thiết kế báo cáo.

Mở báo cáo trong chế độ Design View

Chọn Field List từ menu View (Hoặc click biểu tượng Field List trên thanh công cụ)

Tạo điều khiển Text box bị buộc.

Mở bảng danh sách trường, chọn trường hoặc các trường muốn đặt vào báo cáo

Chọn một trường, click vào trường đó.

Chọn nhiều trường liên nhau, click trường đầu, giữ phím shift, click trường cuối.

Chọn nhiều trường không liên nhau, giữ phím Ctrl và lần lượt click từng trường.

Chọn tất cả các trường trong danh sách, nhấp kép vào thanh tiêu đề của danh sách trường.

Kéo trường (hoặc các trường) được chọn và đặt vào một vị trí trên mẫu báo cáo.

Click biểu tượng Simple preview trên thanh công cụ để xem kết quả.

3.2. Tạo các điều khiển khác dùng hộp công cụ

Muốn tạo các điều khiển không buộc hay dùng để tính toán, phải dùng công cụ trong Toolbox.

Chọn View/Toolbars để hiển thị thanh công cụ

Tạo điều khiển dùng Toolbox

Click công cụ tương ứng loại điều khiển muốn tạo trong báo cáo.

Tạo điều khiển bị buộc bằng cách chọn một trường trong Field list và kéo nó vào trong báo cáo.

3.3. Tạo điều khiển dùng tính toán

Click vào biểu tượng Text box trong hộp công cụ

Click vào một vị trí trên báo cáo. Access tự động gán nhãn cho điều khiển vừa tạo, tiêu đề mặc nhiên thường có dạng “Field0”, có thể thay đổi tiêu đề này theo ý thích.

Đưa con trỏ vào bên trong Text box.

Gõ dấu = , theo sau là biểu thức muốn lập

Ví dụ: =[SOLUONG]*[DONGIA]

Click vào Sample Preview để xem kết quả.

3.4. Tạo điều khiển nhãn

Nhãn là một điều khiển không buộc, nội dung nhãn không thay đổi từ trang này qua trang khác hay từ bản ghi này qua bản ghi khác.

Click biểu tượng Label trong Toolbox.

Click vào một vị trí trên báo cáo để tạo nhãn. Nhãn được tạo như vậy sẽ có kích thước tự mở rộng khi gõ nội dung vào.

Nếu muốn trình bày văn bản thành nhiều dòng, bấm **Ctrl+Enter** cuối dòng thứ nhất.

4. CÁC THUỘC TÍNH CỦA ĐIỀU KHIỂN TRONG BÁO CÁO

Thuộc tính xác định các đặc trưng của đối tượng, mỗi điều khiển trong báo cáo cũng có những thuộc tính riêng. Muốn mở bảng thuộc tính của điều khiển, chọn điều khiển đó và click biểu tượng Properties trên thanh công cụ.

Ta xét một số thuộc tính sau:

Cangrow: Dùng thuộc tính này để làm cho Text box có thể tự điều chỉnh kích thước theo phương dọc đối với khối dữ liệu chứa trong trường nó bị buộc.(chọn Yes).

CanShrink: Khi Text box không có dữ liệu hoặc dữ liệu là chuỗi rỗng Access sẽ chừa trống chỗ đó trên giấy. Điều này có thể làm cho báo cáo quá trống trải nếu có nhiều chỗ như vậy. Chúng ta lập thuộc tính của Text box này thành Yes.

HideDuplicate: Dùng thuộc tính này để che Text box khi giá trị trong đó trùng bản ghi trước

5. SẮP XẾP VÀ TẬP HỢP DỮ LIỆU THEO NHÓM

Sắp xếp là phương pháp phổ biến nhằm tổ chức dữ liệu theo một trật tự nào đó để tìm kiếm và phân loại thông tin.

5.1. Sắp xếp dữ liệu

Khi in báo cáo người dùng thường muốn tổ chức các bản ghi theo một trật tự nào đó. Ví dụ in danh sách cán bộ theo thứ tự giảm dần của lương.

Các bước thực hiện sắp xếp trên báo cáo.

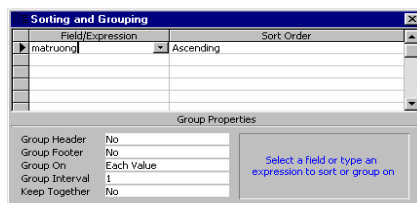
Mở báo cáo ở chế độ Design View.

Chọn Sorting And Grouping trong menu View.

Trong hộp thoại.

Field/Expression: Chỉ định sắp xếp theo trường hoặc biểu thức nào đó.

Sort Order: Chọn Tăng dần hoặc giảm dần.



5.2. Nhóm dữ liệu

Trong nhiều báo cáo, sắp xếp các bản ghi không cũng chưa đủ mà cần phân thành các nhóm. Nhóm là tập hợp các bản ghi cùng với thông tin tóm lược tiêu biểu cho một thể loại thông tin. Một nhóm thường được cấu tạo như sau:

Tiêu đề nhóm (group header), nhóm con (nếu có), các bản ghi chi tiết và chân nhóm (Group footer)

Tiêu đề nhóm 1

Tiêu đề nhóm 2

Tiêu đề nhóm 3

.....

.....

Tiêu đề nhóm 10

Các bản ghi chi tiết

Chân nhóm 10

.....

.....

Chân nhóm 3

Chân nhóm 2

Chân nhóm 1

Chương 7

MACRRO

1. KHÁI NIỆM

Macro trong MS Access là tập hợp các lệnh (Hành động, hành động.....) được định sẵn nhằm tự động thực hiện chuỗi các tác vụ nào đó mà không cần sự can thiệp từng bước của người sử dụng. Macro có thể liên kết các đối tượng trong tập tin cơ sở dữ liệu (CSDL) như: Table, Query, form, report..... nhằm tạo ra các ứng dụng để khai thác có hiệu quả..

Macro được dùng khi có các hành động nào thường xuyên lặp lại trong MS Access hoặc được dùng khi cần kết hợp các hành động đơn giản nhằm giải quyết một vấn đề nào đó khi xây dựng các ứng dụng. Việc tự động hoá các hành động này bởi macro sẽ được thực hiện một cách nhanh chóng và chính xác.

2.TẠO VÀ THI HÀNH MỘT MACRO

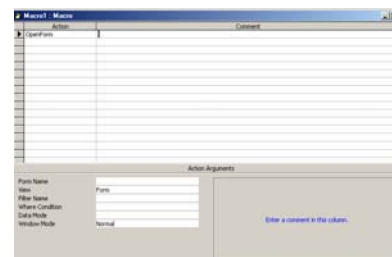
2.1.Tạo một macro

Tại cửa sổ database, chọn đối tượng Macro/ New

Trong mục Action : chọn các hành động cho Macro.

Tong mục Action arguments: Chọn các tham số tương ứng cho hành động.

Lưu trữ Macro: File/save.



2.2.Thi hành macro

Tại cửa sổ database: Chọn đối tượng Macro/Run.

3. CÁC HÀNH ĐỘNG VÀ CÁC THAM SỐ

Ms Access cung cấp một số hành động để tạo macro, mỗi hành động thực hiện một tác vụ nào đó trên CSDL và tùy thuộc vào tham số của mỗi hành động.

Open table: Mở 1 bảng dữ liệu

Table name: Tên bảng cần mở

View: Chọn dạng thể hiện của bảng (Datasheet/ Design/ Print Preview)

Data mode:

Add: Dùng để nhập dữ liệu

Edit: Dùng để thêm, xoá, sửa dữ liệu trong bảng

Read Only: Mở bảng để chỉ xem dữ liệu

Open Query: Mở 1 truy vấn

Query name: Tên truy vấn cần mở

View: Chọn dạng thể hiện của bảng (Datasheet/ Design/ Print Preview)

Data mode:

Add: Dùng để nhập dữ liệu

Edit: Dùng để thêm, xoá, sửa dữ liệu trong bảng

Read Only: Mở bảng để chỉ xem dữ liệu

Open Form: Mở 1 biểu mẫu

Form name: Tên biểu mẫu

View: Chọn dạng thể hiện (Form/ Design/ Print preview/ Datasheet)

Filter name: Tên Query lọc các dữ liệu để hiển thị trong form.

Where condition: Điều kiện lọc dữ liệu hiển thị trong form.

Data mode:

Add : Dùng để nhập dữ liệu

Edit : Dùng để thêm, xoá, sửa dữ liệu trong bảng

Read Only : Mở bảng để chỉ xem dữ liệu

Window mode:

Normal : Dạng cửa sổ form bình thường.

Hidden : Dạng cửa sổ form được ẩn đi.

Icon : Cửa sổ form thu nhỏ thành 1 biểu tượng.

Dialog : Dạng hộp thoại.

Open Report: Mở 1 báo cáo

Report name: Tên báo cáo

View: Chọn kiểu in

Print preview: In ra màn hình.

Design: Dạng thiết kế báo cáo

Print: In ra máy in

Filter name: Tên Query lọc các dữ liệu để hiển thị trong Report.

Where condition: Điều kiện lọc dữ liệu hiển thị trong Report.

Run macro: Thực hiện một tập lệnh

Macro name: Tên macro cần thực hiện

Repeat count: Số lần thực hiện macro sẽ lặp lại.

Repeat Expression: Biểu thức điều kiện để lặp lại khi thực hiện macro. Macro chỉ dừng khi khi biểu thức điều kiện nhận giá trị False.

Open module: Mở cửa sổ soạn thảo thủ tục trong 1 module.

Module name: Tên module chứa thủ tục cần mở.

Procedure name: Tên thủ tục sẽ mở.

Run code: Gọi thực hiện một hàm của Access Basic

Function name: Tên hàm cần thực hiện và các đối số của hàm.

Run App: Cho thực hiện một ứng dụng nào đó trong môi trường Windows

Command line: đường dẫn đến tập tin của một ứng dụng.

Run SQL: Cho thực hiện câu lệnh SQL

SQL Statement: Nội dung câu lệnh SQL

Maximize: Cực đại cửa sổ hiện thời

Minimize: Cực tiểu cửa sổ hiện thời thành một biểu tượng.

Restore: Phục hồi cửa sổ trở về kích thước cũ.

Move size: Di chuyển hoặc thay đổi kích thước cửa sổ hiện thời.

Right: Khoảng cách từ góc trên trái của cửa sổ này đến cạnh trái của cửa sổ chứa nó.

Down: Khoảng cách từ góc trên trái của cửa sổ này đến cạnh trên của cửa sổ chứa nó.

Width: Chiều rộng của cửa sổ này.

Height: Chiều cao của cửa sổ này.

Stop Macro: Dừng macro đang thực hiện

Beep: Phát tiếng kêu bíp

Hourglass: Đổi dạng con trỏ thành đồng hồ cát trong khi macro đang chạy

Hourglass On: Yes/No (Đổi/ Không đổi)

Close: Đóng một cửa sổ đang hoạt động

Object Type: Loại cửa sổ của đối tượng cần đóng như Table, Query, form, Report, Macro hoặc Module.

Object name: Tên của đối tượng cần đóng.

Quit: Thoát khỏi MS Access và trở về Windows

Option

Prompt: Hiện thị hộp thoại có lưu trữ không? Nếu đối tượng có thay đổi.

Save all: Lưu trữ tất cả mọi đối tượng.

Exit: Thoát mà không cần lưu trữ.

Print: In đối tượng hiện thời

Print Range: Phạm vi cần in ấn.

All: In tất cả các đối tượng

Selection: In phần trang được chọn

Pages: In các trang được chọn

Page from: Trang bắt đầu in

Page to: Trang kết thúc in

Print Quality: Chất lượng in

Copies: Số bản cần in

Collate Copies: Có sắp xếp thứ tự các bản in theo trang.

Msg Box: Hiện thị hộp thông báo

Message: Câu thông báo cần hiển thị

Beep: Yes/ No: Có/ Không phát ra tiếng Bíp khi hiển thị hộp thông báo.

Type: Loại hộp thông báo.

Title: Tiêu đề của hộp thông báo.

CancelEvent: Huỷ bỏ một sự kiện đang thực hiện

Requery: Cập nhật dữ liệu cho một đối tượng đang hoạt động bằng cách cập nhật lại dữ liệu nguồn của đối tượng đó.

Control name: Tên của đối tượng cần cập nhật dữ liệu (Nếu không chỉ ra thì sẽ cập nhật lại dữ liệu nguồn của chính đối tượng đang hoạt động).

Select Object: Chọn đối tượng trong CSDL

Object Type: loại đối tượng cần chọn.

Object name: Tên đối tượng cần chọn

In Database Window: (Yes/No) Xác định MS access có chọn đối tượng trong cửa sổ CSDL không, mặc định là No.

Set value: Gán một giá trị cho 1 trường, 1 điều khiển, hoặc một thuộc tính trên một Form hoặc 1 Report.

Item: Tên trường, đối tượng hay thuộc tính muốn gán giá trị.

Expression: Biểu thức cần gán giá trị cho Item.

☛ **Chú ý:** Nếu tên trường, tên đối tượng, tên thuộc tính ở 1 Form hoặc 1 Report khác thì phải mô tả đầy đủ.

Trong Form khác: [Forms]![Tên Form]![Tên trường/Tên đối tượng]

Trong Report khác: [Reports]![Tên Report]![Tên trường/Tên đối tượng]

Đối với các thuộc tính

[Forms/Reports]![Tên Form/Tên Report]![Tên trường].[Tên thuộc tính]

Add menu: Tạo thêm một Drop Down Menu vào một menu bar cho một form hoặc Report.

Menu name: Tên của Drop Down Menu muốn thêm vào menu bar.

Menu macro name: Tên macro chứa các lệnh về việc tạo menu.

Status bar: Thông báo ở thanh trạng thái khi chọn menu này.

Apply Filter: Lọc (Truy vấn) các dữ liệu khi xử lý Table, Form, Report.

Filter name: Tên của truy vấn lọc dữ liệu.

Where condition: Điều kiện lọc dữ liệu.

FindRecord: Tìm bản ghi đầu tiên nằm trong phạm vi và thoả mãn điều kiện.

Find What: Nội dung dữ liệu cần tìm là một giá trị hoặc một biểu thức, nếu là biểu thức sẽ bắt đầu dấu "=".

Where: Qui định cách so sánh giá trị cần tìm với giá trị của trường.

Any part of field: Một phần bất kỳ của trường.

Match Whole field: Giá trị cần tìm bằng giá trị của trường.

Start of field: Giá trị cần tìm là phần đầu của trường.

Match Case: Yes/No: Có/Không phân biệt chữ in hoa và chữ in thường.

Direction: Quy định hướng tìm

All: Tìm toàn bộ

Up: Tìm từ bản ghi hiện thời lên phía trên.

Down: Tìm từ bản ghi hiện thời lên phía dưới.

Search As Formatted: Qui định việc tìm có dựa trên dữ liệu sau khi đã định dạng trong các trường hay không?

Search in: Qui định việc tìm trên trường hiện thời hoặc trên tất cả các trường.

Current Field: Tìm trên trường hiện thời.

All Fields: Tìm tất cả các trường.

Find First: Yes/No: Qui định tìm từ bản ghi đầu tiên hay tìm từ bản ghi hiện thời.

Findnext: Tìm bản ghi kế tiếp thoả mãn điều kiện tìm kiếm của lệnh FindRecord.

CopyObject: Sao chép một đối tượng trong tập tin CSDL hiện thời thành một đối tượng khác của tập tin CSDL khác trong MS Access.

Destination Database: Tên tập tin CSDL đích

New name: Tên mới của đối tượng sau khi sao chép.

Source Object Type: Kiểu của đối tượng nguồn.

Source Object Name: Tên của đối tượng nguồn.

DeleteObject: Xoá một đối tượng trong tập tin CSDL hiện thời.

Object Type: Kiểu của đối tượng.

Object Name: Tên của đối tượng

4. NHÓM TẬP LỆNH VÀ TẬP LỆNH CÓ ĐIỀU KIỆN

4.1. Nhóm tập lệnh

Là Macro chứa các macro con, thay vì tạo ra nhiều macro với nhiều tên khác nhau thì các macro này được gom lại thành một tên chung nhằm giảm bớt số lượng và thuận lợi trong quá trình sử dụng. Tuy nhiên các Macro được nhóm khi chúng có liên quan với nhau.

Có thể có nhiều Macro trong nhóm có cùng hành động, tuy nhiên chúng được phân biệt bởi tên Macro.

Đặt tên cho Macro ta thực hiện:

Tại chế độ thiết kế Macro:

View/Macro name

Đặt tên cho Macro tại cột Macro name.

Cách thực hiện 1 macro trong macro name

<Tên Macro Group>.<Tên Macro cần thực hiện>

4.2. Macro có điều kiện

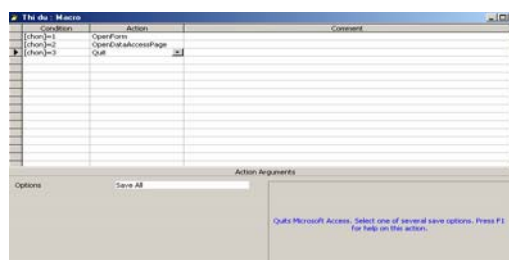
Là macro có chứa các điều kiện thi hành cho mỗi hành động.

Cách tạo Macro có điều kiện

Tại chế độ thiết kế Macro: Chọn View/Conditions

Tại cột Condition : Đặt điều kiện thi hành cho mỗi hành động.

Ví dụ



4.3. Áp dụng Macro cho form và Report

4.3.1. Quy tắc chung khi gọi một đối tượng

Đối với form : *Forms![Tên form]![Tên đối tượng]*

Đối với Report : *Reports![Tên Report]![Tên đối tượng]*

4.3.2. Các thuộc tính của một đối tượng

Muốn gắn một nút lệnh trên một biểu mẫu hoặc báo cáo với một Macro nào đó vào nút lệnh này ta thực hiện: Click chuột phải vào nút lệnh, chọn Properties và gắn Macro vào các hành động tương ứng.

On Enter: Macro thi hành khi nhấn Enter vào bên trong đối tượng

On Exit: Macro thi hành khi thoát khỏi đối tượng

On Got Focus: Thiết lập nhận biết khi có di chuyển con trỏ đến một form hoặc 1 trường trên form đang mở.

On Click: Macro thi hành khi click vào đối tượng

On Dbl Click: Macro thi hành khi Double click vào đối tượng.

On Mouse Down: Macro thi hành khi ấn và giữ chuột tại đối tượng

On Mouse Move: Macro thi hành khi di chuyển chuột ra khỏi đối tượng

On Mouse Up: Macro thi hành khi nhả chuột ra khỏi đối tượng.

On Key Down: Macro thi hành khi ấn và giữ một phím đối tượng.

On Key Press: Macro thi hành khi ấn một phím đối tượng.

On Key Up: Macro thi hành khi nhả một phím đối tượng.

4.3.3. Macro tự động thực hiện sau khi mở tập tin CSDL

Chúng ta có thể tạo ra một Macro mà mỗi khi mở một tập tin CSDL thì Macro này tự động thực hiện.

Để tạo Macro tự động thực hiện ta tiến hành các thao tác sau:

Tạo macro

Lưu trữ Macro với tên AutoExec.

5. THIẾT KẾ MENU TRONG ACCESS

Trong các ứng dụng, thường chúng ta phải tổ chức Menu để cho phép người sử dụng thực hiện các hành động thông qua các chức năng trên Menu này.

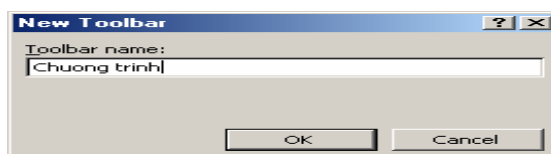
Có 2 cách hiển thị các Menu của người sử dụng (Custom Menu) trong các ứng dụng đó là:

Menu ứng với 1 form xác định nào đó: Menu này chỉ xuất hiện khi truy xuất đến Form này.

Menu toàn cục (Globally): Menu toàn cục này sẽ xuất hiện trong các ứng dụng và nó chỉ được thay thế khi có 1 menu ứng với form nào đó được mở.

5.1. Tạo menu của người sử dụng

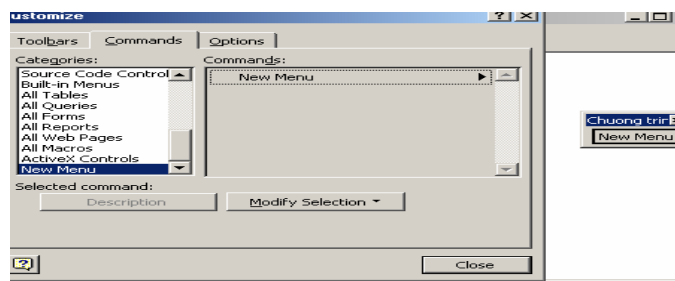
Chọn View/Toolbars/Customize/New/ Đặt tên cho thanh menu



Chọn Properties và chọn Type là Menu bar, close

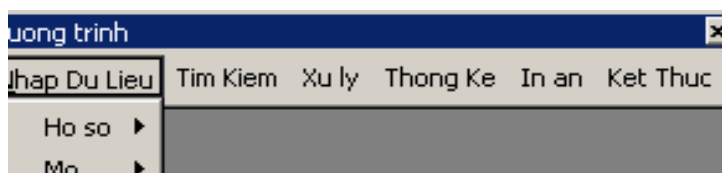


Chọn command, trong mục Categories chọn New menu và kéo sang thanh menu vừa tạo.



Click chuột phải để thay đổi các tiêu đề cho phù hợp trên thanh menu.

Tương tự cho các nhóm khác.



Có thể tạo các chức năng là các hành động như mở bảng, truy vấn, biểu mẫu.... bằng cách kéo các biểu tượng này trong mục command vào menu đang tạo.

5.2. Tạo menu toàn cục

Nếu muốn tạo một menu toàn cục sẽ thay thế menu có sẵn trong Access khi CSDL này được mở ta thực hiện các bước sau:

Mở hoặc tạo ra một Macro có tên là Autoexec.

Thêm vào hành động: SetValue

Trong mục các tham số (arguments):

Item: Application.menubar

Expression: Tên của Menubar người sử dụng tạo ra và đặt giữa cặp dấu ngoặc kép "".

BÀI SỐ 1

Khởi động MSACCESS, tạo một CSDL có tên NHANSU.MDB rồi lần lượt tạo các bảng dữ liệu sau

Bảng 1: Nhanvien

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|-------------|---------------|-----------|-----------------------|
| 1 | Manv | Text | 4 | Khoá chính |
| 2 | Holot | Text | 20 | |
| 3 | Ten | Text | 10 | |
| 4 | Ngaysinh | Date/time | 8 | |
| 5 | Gioitinh | Yes/No | 1 | Yes:Nam, No:Nu |
| 6 | Maphong | Text (Lookup) | 2 | Lấy từ bảng Dsphong |
| 7 | MaPXuong | Text (Lookup) | 2 | Lấy từ bảng dspxuong |
| 8 | Diachi | Text | 30 | |
| 9 | Ghichu | Memo | | |

Bảng 2: Dsphong

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|----------------|-----------|-----------|------------|
| 1 | Maphong | Text | 2 | Khoá chính |
| 2 | Tenphong | Text | 30 | |
| 3 | Sodthoai | Text | 11 | |

Bảng 3: Thunhap_NV

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|-------------|---------------|-----------|------------------|
| 1 | Manv | Text (Lookup) | 4 | Khoá chính |
| 2 | Luongchinh | Number | Double | |
| 3 | Heso | Number | Integer | |
| 4 | Phucap | Number | Double | |
| 5 | Thue | Number | Double | Thuế |
| 6 | Thamnien | Number | Byte | Số năm thâm niên |
| 7 | Tongluong | Number | Double | |

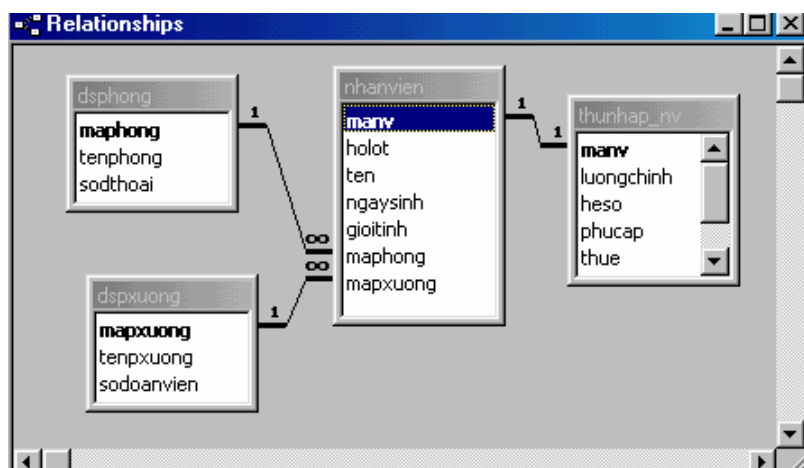
Bảng 4: DSpxuong

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|-----------------|-----------|-----------|--------------|
| 1 | Mapxuong | Text | 2 | Khoá chính |
| 2 | Tenpxuong | Text | 20 | |
| 3 | Sodoanvien | Number | Byte | Số đoàn viên |

1. Hãy nhập dữ liệu thích hợp cho các bảng trên.
2. Mở bảng Nhanvien sắp xếp tăng dần theo **manv**, sắp xếp giảm dần theo **ten**.
3. Sắp xếp tăng dần theo **ten**, nếu **ten** trùng nhau thì sắp xếp giảm dần theo **holot** nếu **holot** trùng nhau thì sắp xếp giảm dần theo **maphong**.
4. Lọc và hiển thị danh sách những nhân viên có tên **Thanh**.
5. Lọc và hiển thị những nhân viên có họ **Nguyen** hoặc **Cao** và có tên **anh**.
6. Lọc và hiển thị những nhân viên có **Manv** bắt đầu là **B** và **maphong** là **A1** hoặc **Manv** bắt đầu là **C** và **maphong** là **A2**.
7. Lọc và hiển thị những nhân viên **nam** và có năm sinh **1975**.
8. Lọc và hiển thị những nhân viên **nam** sinh trong thời gian từ **12/12/76** đến **12/12/79**.
9. Lọc và hiển thị những nhân viên **nữ** có **Mapxuong** là **P1** và sinh trong tháng **10** năm **1975** hoặc nhân viên **nam** có năm sinh **1972** đến **1976**.
10. Mở bảng **Thunhap_nv** để hiển thị những nhân viên có **heso>100** và có **luongchinh<500.000** hoặc có **phucap** từ **100.000** đến **500.000**.

BÀI SỐ 2

1. Mở CSDL NHANSU.MDB trong bài thực hành số 1 rồi thực hiện các nhiệm vụ sau:
 - ❖ Đặt khoá chính cho trường **Manv** (**nhanvien**), **Maphong**(**dsphong**), **manv**(**thunhap_nv**) và **mapxuong** (**dspxuong**).
 - ❖ Thiết lập các mối quan hệ giữa các bảng dữ liệu theo sơ đồ sau



❖ Hãy thiết lập các thuộc tính tham chiếu toàn vẹn và các thuộc tính như Format, caption, inputmask..... cho các trường một cách hợp lệ.

Đặt thuộc tính Format của trường **ngaysinh** : dd-mm-yy , Inputmask: 00/00/00

Đặt thuộc tính Inputmask của trường **sodthoai** : (000)000000

2. Tạo một truy vấn **query1** để hiển thị những thông tin sau:

Manv, holot, ten (trong bảng **nhanvien**), Luongchinh, phucap (Trong bảng **Thunhap_nv**), tenphong, maphong (Trong **dsphong**).

3. Tạo một truy vấn **query 2** để hiển thị những thông tin sau:

Manv, hoten (Nối holot và ten), maphong, tenphong, mapxuong, tenpxuong, tongthunhap trong đó tongthunhap được tính theo công thức:

Tongthunhap=luongchinh*heso+phucap-thue nếu là nhân viên **nam**.

Tongthunhap=luongchinh*heso+phucap nếu là nhân viên **nữ** .

4. Tạo một truy vấn **query3** để hiển thị những nhân viên có **maphong** là **p1** và **mapxuong** là **x1** sinh trong tháng 7 năm 1976 bao gồm những tin sau:

Hoten, gioitinh, maphong, tenphong, mapxuong, tenpxuong.

5. Tạo một truy vấn **query4** để hiển thị những nhân viên **nữ** và có **maphong** là **p2** hoặc **p3** sinh trong ngày 20 tháng 12 bao gồm những thông tin:

Hoten, maphong, ngaysinh, gioitinh (chú ý giá trị trường gioitinh phải hiển thị nam hoặc nữ).

6. Tạo một truy vấn **query5** để hiển thị những nhân viên **nam** có **tongthunhap** ≥ 60000 hoặc thuộc **maphong** là **p2** và không phải họ **Nguyễn** bao gồm những thông tin: Ten, maphong, tenphong, tongthunhap.
7. Tạo một truy vấn **query6** để hiển thị những nhân viên **nữ** có **maphong** không bắt đầu là **p** hoặc những nhân viên thuộc phân xưởng không có đoàn viên nào bao gồm những thông tin sau: Holot, ten, gioitinh, tenpxuong.
8. Tạo một truy vấn **query7** để hiển thị những nhân viên **nam** tên **Thanh** hoặc **Long** hoặc những nhân viên **nữ** không phải họ **Lê** hoặc **trần** bao gồm những thông tin: Hoten, gioitinh, maphong, tenphong.
9. Tạo một truy vấn **query8** để hiển thị những nhân viên **nam** sinh trong khoảng thời gian từ năm **1973** đến **1980** thuộc phân xưởng có **mapxuong** là **x2** hoặc những nhân viên không có **thuế** sinh trong tháng **4** đến tháng **8** năm **1975** bao gồm những thông tin: Holot, ten, thangsinh, namsinh, mapxuong, tenpxuong.
10. Tạo một truy vấn **query9** để hiển thị những thông tin: Hoten, namsinh, thamnien, luongchinh, trong đó nếu

Luongchinh ≥ 3000 và nhân viên **nữ** thì thamnien là **35**.

Luongchinh ≥ 4000 và nhân viên **nam** thì thamnien là **30**.

BÀI SỐ 3

Tạo một CSDL có tên QLTV.MDB, rồi lần lượt tạo các bảng dữ liệu sau:

Bảng 1: Loaisach

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|---------------|-----------|-----------|-----------------|
| 1 | Masach | Text | 4 | Khoá chính |
| 2 | Tensach | Text | 20 | |
| 3 | Tentacgia | Text | 20 | |
| 4 | Namxb | Date/time | 8 | |
| 5 | Soluongco | Number | 3 | Số lượng có |
| 6 | Sotrang | Number | 5 | Số trang sách |
| 7 | Manxb | Text | 4 | Mã nhà xuất bản |

Bảng 1: Docgiamuon

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|-----------------|---------------|-----------|-------------------------|
| 1 | Hoten | Text | 30 | |
| 2 | Quequan | Text | 30 | |
| 3 | Madocgia | Text | 4 | Khoá chính |
| 4 | Masach | Text (Lookup) | 4 | Lấy dữ liệu từ Loaisach |
| 5 | Sluong | Number | 1 | Số lượng mượn |
| 6 | Ngaymuon | Date/time | 8 | Ngày mượn |
| 7 | Ngayhen | Date/time | 8 | Ngày hẹn trả |

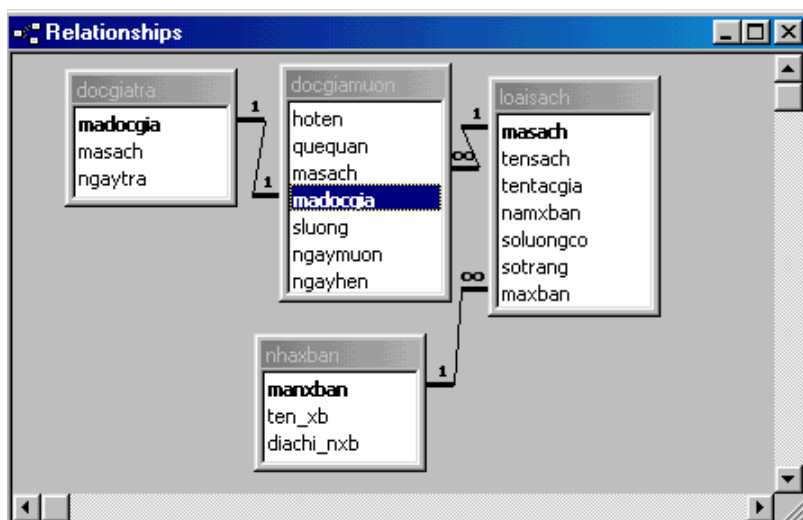
Bảng 3: Docgiatra

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|-----------------|-----------|-----------|------------|
| 1 | Madocgia | Text | 4 | Khoá chính |
| 2 | Masach | Text | 4 | |
| 3 | Ngaytra | Date/time | 8 | Ngày trả |

Bảng 4: : Nhaxban

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|----------------|-----------|-----------|------------------|
| 1 | Manxban | Text | 4 | Khoá chính |
| 2 | Ten_xb | Text | 30 | Tên nhà xuất bản |
| 3 | Diachi_nxb | Text | 30 | Địa chỉ Nhà XB |

1. Nhập dữ liệu thích hợp cho các bảng dữ liệu.
2. Đặt khoá chính cho các trường: masach(Loaisach), Madocgia(Docgiamuon), Madocgia(Docgiatra), manxban(Nhaxban).
3. Thiết lập các mối quan hệ giữa các bảng dữ liệu theo sơ đồ sau



4. Tạo một truy vấn **TV1** để hiển thị danh sách độc giả mượn sách trong tháng 10 năm 98 bao gồm những thông tin: Hoten, ngaymuon, tensach, tentacgia.
5. Tạo một truy vấn **TV2** để hiển thị số lượng còn của mỗi loại sách của nhà xuất bản **Kim đồng** bao gồm những thông tin: tensach, nxban, soluongcon.
6. Tạo một truy vấn **TV3** để hiển thị danh sách những độc giả mượn sách quá hạn bao gồm những thông tin: Hoten, Ngaymuon, Ngayhen, songayqua (Số ngày quá hạn).
7. Tạo một truy vấn **TV4** để hiển thị danh sách những độc giả mượn sách quá hạn bao gồm những thông tin Hoten, sluong, mucquahan
Trong đó: Mucquahan là Mức 1 nếu songayqua<5
 Mucquahan là Mức 2 nếu songayqua<10
 Mucquahan là Mức 3 nếu songayqua>=10
8. Tạo một truy vấn **TV5** để hiển thị danh sách những độc giả trả sách đúng hạn hoặc sớm hạn bao gồm những thông tin: Hoten, masach, tensach.
9. Tạo một truy vấn **TV6** để hiển thị những loại sách có số trang >100 và số lượng còn là 20 của **nhà xuất bản giáo dục** hoặc tên sách có chữ **Tin học** bao gồm Tensach, sotrang, ten_xb, tentacgia.

BÀI SỐ 4

1. Sử dụng CSDL QLTV.MDB, tạo một truy vấn **Truyvan1** để hiển thị tổng số lượng mượn của từng loại sách trong thư viện.
2. Tạo một truy vấn **Truyvan2** để hiển thị tổng số lượng mượn của từng loại sách trong tháng 12 năm 1998.
3. Tạo một truy vấn **Truyvan3** để hiển thị tổng số lượng mượn của từng loại sách theo từng tháng trong năm 1999.
4. Tạo một truy vấn **Truyvan4** để hiển thị tổng số lượng mượn của từng loại sách theo từng tháng của một năm nào đó(Tháng và năm được nhập từ bàn phím).
5. Tạo một truy vấn **Truyvan8** để hiển thị số lượng mượn của từng loại sách trong năm 1999 và có số lượng mượn của mỗi độc giả >2.
6. Tạo một truy vấn **Truyvan9** để hiển thị tổng số loại sách có trong thư viện.

7. Tạo một truy vấn **Truyvan10** để hiển thị tổng số loại sách trong thư viện cho mượn trong năm **1998**.
8. Tạo một truy vấn **Truyvan11** để hiển thị tên của độc giả nào đó mượn sách trong **tháng 1 năm 2001** (Tên được nhập từ bàn phím).
9. Tạo một truy vấn **Truyvan12** để hiển thị tên sách và tên tác giả xuất bản sách trong năm **2002** (Tên sách nhập từ bàn phím, Họ của tác giả nhập từ bàn phím).
10. Tạo một truy vấn tham khảo chéo **Truyvan13** để phản ánh tổng số mỗi loại sách mượn cho mượn trong **tháng 11 năm 2000**.
11. Tạo một truy vấn tham khảo chéo **Truyvan14** để hiển thị tổng số loại sách xuất bản trong năm **1995**.
12. Tạo một truy vấn **Truyvan14** để xoá những sinh viên đã mượn sách quá hạn **5** ngày.

BÀI SỐ 5

Tạo một CSDL có tên QLSV.MDB, rồi tạo các bảng dữ liệu sau:

Bảng 1: DSSV

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|-------------|-----------|-----------|------------|
| 1 | Masv | Text | 4 | Khoá chính |
| 2 | Malop | Text | 4 | |
| 3 | Hotensv | Text | 30 | |
| 4 | Ngaysinh | Date/time | 8 | |
| 5 | Quequan | Text | 30 | |
| 6 | Gioitinh | Yes/No | 1 | |
| 7 | Hocbong | Number | Double | |

Bảng 2: DSDIEM

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|-------------|-----------|-----------|----------------|
| 1 | Masv | Text | 4 | Khoá chính |
| 2 | Mamon | Text | 2 | Mã môn học |
| 3 | Diem_lan1 | Number | Double | Điểm thi lần 1 |
| 4 | Diem_lan2 | Number | Double | Điểm thi lần 2 |

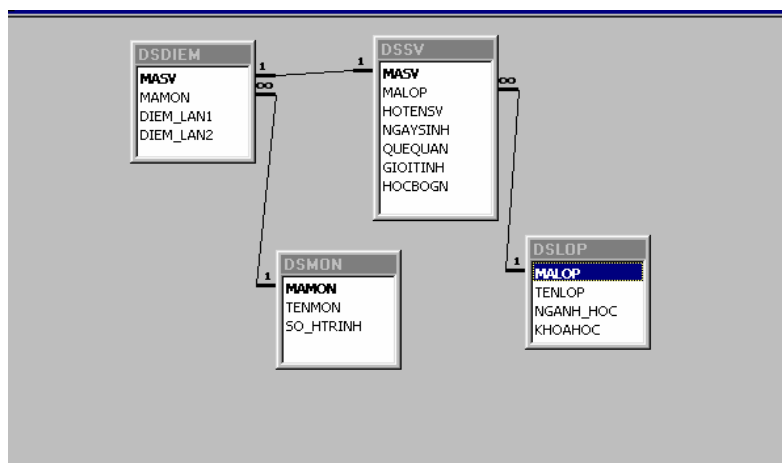
Bảng 3: DSLOP

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|--------------|-----------|-----------|------------|
| 1 | Malop | Text | 4 | Khoá chính |
| 2 | Tenlop | Text | 20 | |
| 3 | Nganh_hoc | Text | 30 | |
| 4 | Khoahoc | Text | 2 | |

Bảng 3: DSMON

| STT | Fieldname | Data Type | Fieldsize | Note |
|-----|--------------|-----------|-----------|--------------|
| 1 | Mamon | Text | 2 | Khoá chính |
| 2 | Tenmon | Text | 20 | |
| 3 | So_htrinh | Number | Byte | Số học trình |

1. Nhập dữ liệu thích hợp cho các bảng dữ liệu.
2. Đặt khoá chính cho các trường: MASV(DSSV), MASV(DSDIEM), MALOP(DSLOP), Mamon(DSMON).
3. Thiết lập các mối quan hệ giữa các bảng theo sơ đồ sau:



4. Tạo một truy vấn **BT1** để hiển thị tổng số sinh viên của mỗi lớp là bao nhiêu?
5. Tạo một truy vấn tham khảo chéo **BT2** để phản ánh tổng số sinh viên xếp loại *Xsắc*, *Giỏi*, *khá*, *Tb*, *Yếu* của mỗi lớp.
6. Từ bảng dữ liệu DSSV, tạo một truy vấn **BT3** để tạo ra bảng dữ liệu DSLUU lưu trữ những sinh viên có Mã lớp bắt đầu là **T** hoặc **H**.

7. Từ bảng dữ liệu DSSV, tạo một truy vấn **BT4** để tạo ra một bảng dữ liệu DSNAM để lưu trữ những sinh viên **nam** sinh trong năm **1976** bao gồm các thông tin: HOTENV, QUEQUAN, GIOTINH.
8. Từ bảng dữ liệu DSSV, tạo một truy vấn **BT5** để tạo ra một bảng dữ liệu DSNU để lưu trữ những sinh viên **nữ** sinh trong **quý 3** năm **1979** bao gồm các thông tin: HOTENV, QUEQUAN, GIOTINH.
9. Tạo một truy vấn **BT6** để tăng học bổng cho những sinh viên **nữ** thêm **30%**.
10. Tạo một truy vấn **BT7** để nối dữ liệu từ bảng DSNAM vào bảng DSNU.
11. Tạo truy vấn **BT8** để xóa những sinh viên có quê quán ở **Đà Nẵng** và có mã lớp bắt đầu là **B**.

BÀI SỐ 6

1. Sử dụng ngôn ngữ SQL để tạo cấu trúc của các bảng dữ liệu sau:

DSTRUONG (MATRUONG, TENTRUONG, DIACHI)

DSKHOA (MATRUONG, MAKHOA, TENKHOA, SODT)

DANHSACH (MASV, MAKHOA, HOTEN, NGAYSINH, LOP, HOCBONG)

BANGDIEM (MASV, DTBK1, DTBK2, DTBK3, DTBK4)

Trong đó: Các trường in đậm và gạch chân là khoá chính, kiểu dữ liệu và kích thước của các được mô tả như sau:

Bảng DSTRUONG

MATRUONG TEXT(2), TENTRUONG TEXT(20), DIACHI TEXT(30)

Bảng DSKHOA

MATRUONG TEXT(2), MAKHOA TEXT(4), TENKHOA TEXT(10), SODT TEXT(6).

Trường SODT được lập chỉ mục.

Bảng DANHSACH

MASV TEXT(4), MAKHOA TEXT(4), HOTEN TEXT(30), NGAYSINH(DATE/TIME), LOP TEXT(10), HOCBONG (DOUBLE). Trường MASV được lập chỉ mục.

Bảng BANGDIEM

MASV TEXT(4), DTBK1 (DOUBLE), DTBK2 (DOUBLE), DTBK3 (DOUBLE), DTBK4 (DOUBLE).

Chú ý

Sau khi tạo cấu trúc các bảng dữ liệu xong, hãy nhập dữ liệu và thiết lập mối quan hệ giữa các bảng phù hợp. Trường HOCBONG trong bảng DANHSACH chỉ nhập một trong ba giá trị 120000, 180000 hoặc 240000.

2. Sử dụng ngôn ngữ SQL để thay đổi cấu trúc của bảng dữ liệu

- a. Thêm trường GHICHU có kiểu MEMO vào trong bảng DANHSACH*
- b. Thêm trường TBCONG có kiểu DOUBLE vào trong bảng BANGDIEM*
- c. Thêm trường QUEQUAN có kiểu TEXT và GIOITINH có kiểu YES/NO vào bảng DANHSACH và lập chỉ mục trường QUEQUAN.*

3. Sử dụng ngôn ngữ SQL để tạo các truy vấn chọn sau

- a. Chọn MATRUONG, MAKHOA, TENKHOA trong bảng DSKHOA.*
- b. Chọn MATRUONG, MAKHOA, SODT trong bảng DSKHOA của những trường có MATRUONG bắt đầu là Q.*
- c. Chọn MASV, MAKHOA, HOTEN của những sinh viên sinh trong khoảng thời gian từ 20/10/74 đến 20/10/76 trong bảng DANHSACH.*
- d. Chọn MASV, HOTEN, LOP, HOCBONG của những sinh viên có MASV bắt đầu là T và thuộc lớp Tin học hoặc Hoá học hoặc kinh tế trong bảng DANHSACH (HOTEN đổi thành Họ và tên).*
- e. Chọn những sinh viên có tên THANH sinh trong tháng 10/76 hoặc có HOCBONG trong khoảng từ 150000 đến 200000.*

4. Sử dụng ngôn ngữ SQL để tạo các truy vấn tính tổng sau:

- a. Tạo một truy vấn để tính tổng HOCBONG của mỗi khoa.
- b. Tạo một truy vấn thống kê xem mỗi khoa số lượng sinh viên là bao nhiêu?
- c. Tạo một truy vấn để tính tổng HOCBONG của mỗi lớp trong mỗi khoa.
- d. Tạo một truy vấn thống kê xem mỗi trường có bao nhiêu khoa?

5. Sử dụng ngôn ngữ SQL để tạo các truy vấn tham số sau:

- a. Tạo một truy vấn tham số thống kê xem mỗi mức HOCBONG mỗi khoa có bao nhiêu sinh viên.
- b. Tạo một truy vấn tham số để thống kê xem số lượng sinh viên sinh trong mỗi tháng của năm 1979 của mỗi khoa là bao nhiêu?

6. Sử dụng ngôn ngữ SQL để tạo các truy vấn tạo bảng sau:

- a. Tạo một bảng DIEMLUU gồm tất cả các trường trong bảng DIEMTHI.
- b. Tạo một bảng DSLUU gồm các trường MASV, MAKHOA, HOTEN từ bảng DANHSACH của những sinh viên sinh trước ngày 20/11/76.
- c. Tạo một bảng DSLUU1 gồm các trường HOTEN, LOP của những sinh viên thuộc lớp **Tin K25A** và sinh năm 1985 hoặc trước năm 1978.

7. Sử dụng ngôn ngữ SQL để tạo các truy vấn nối dữ liệu sau:

- a. Tạo một truy vấn nối dữ liệu từ bảng DIEMLUU vào DIEMTHI nhưng chỉ gồm các trường DTBK1, DTBK2.
- b. Tạo một truy vấn nối dữ liệu từ bảng DIEMLUU vào DIEMTHI nhưng chỉ gồm các trường DTBK1, DTBK2, DTBK3 đối với những bản ghi ghi có $DTBK3 \geq 8$.

8. Sử dụng ngôn ngữ SQL để tạo các truy vấn cập nhật dữ liệu sau:

- a. Tính giá trị trường DTBCONG của bảng DIEMTHI theo công thức $(DTBK1 + DTBK2 + DTBK3 + DTBK4)/4$.
- b. Tăng HOCBONG thêm 100.000 cho những sinh viên có MASV bắt đầu là A trong bảng DANHSACH.

c. Giảm HOCBONG đi 50.000 cho những sinh viên có năm sinh từ 1975 đến 1978 hoặc những sinh viên có tên NHAN.

9. Sử dụng ngôn ngữ SQL để tạo các truy vấn xoá sau:

a. Tạo một truy vấn xoá những sinh viên thuộc khoa toán.

b. Tạo một truy vấn xoá những sinh viên sinh trong khoảng thời gian từ 20/10/74 đến 20/10/76 hoặc có HOCBONG=120.000.

c. Tạo một truy vấn xoá những sinh viên có họ NGUYEN hoặc tên THANH sinh trong tháng 7 năm 1978.

10. Sử dụng ngôn ngữ SQL để tạo các truy vấn dựa trên nhiều bảng sau:

a. Tạo truy vấn để hiển thị HOTEN, TENKHOA, LOP, HOCBONG từ 2 bảng dữ liệu DSKHOA và DANHSACH.

b. Tạo một truy vấn để hiển thị TENTRUONG, TENKHOA, HOTEN của những sinh viên sinh trong năm 1980 từ 3 bảng DSTRUONG, DSKHOA, DANHSACH.

c. Tạo một truy vấn để hiển thị TENTRUONG, TENKHOA, HOTEN, DTBK1, DTBK2 từ 4 bảng dữ liệu DSTRUONG, DSKHOA, DANHSACH, DIEMTHI.

11. Sử dụng ngôn ngữ SQL để tạo các truy vấn con sau:

a. Tạo một truy vấn để hiển thị HOTEN, NGAYSINH, LOP của những sinh viên có DTBK4 \geq 5.

b. Tạo một truy vấn để hiển thị ít nhất một sinh viên có DTBK2 \leq 4.

12. Sử dụng ngôn ngữ SQL để tạo các truy vấn hội sau:

a. Tạo truy vấn hội để hiển thị HOTEN, DTBK1, DTBK2 từ 2 bảng dữ liệu DIEMTHI và DIEMLUU.

b. Tạo truy vấn hội để hiển thị HOTEN, DTBK1, DTBK2, DTBK3 từ 2 bảng dữ liệu DIEMTHI và DIEMLUU nhưng chỉ hiển thị những sinh viên có DTBK3 \geq 7.

BÀI SỐ 7

Sử dụng CSDL **QLSV.MDB** trong **bài tập số 5** để thực hiện các yêu cầu sau:

1. Tạo một form có tên **Nhaplop** để nhập dữ liệu cho bảng **dslop** như sau:

2. Tạo một form chính phụ như sau:

3. Tạo một form có tên **Hienthi** để hiển thị danh sách sinh viên như sau :

4. Hãy thiết kế form theo yêu cầu sau:

Hãy gán các chức năng phù hợp với các mục chọn.

5. Hãy thiết kế một **Report** theo yêu cầu sau:

| Mã lớp | Mã sv | Họ và tên | Ngày sinh | Quê quán | Giới tính | Học bổng |
|--------|-------|------------------|-----------|------------|-------------------------------------|----------|
| K23A | a001 | Nguyễn Tèo | 10-10-78 | Huế | <input checked="" type="checkbox"/> | 1000000 |
| | a002 | Trần Bình Chi | 05-05-79 | Đà Nẵng | <input type="checkbox"/> | 2000000 |
| | a005 | Lê Mậu Ngọ | 08-08-79 | Huế | <input checked="" type="checkbox"/> | 1200000 |
| K24A | a003 | Nguyễn Thanh Thu | 12-10-79 | Huế | <input checked="" type="checkbox"/> | 2300000 |
| | a004 | Võ Tông | 02-02-79 | Quảng Bình | <input type="checkbox"/> | 2200000 |
| K24B | a010 | Trần Bạch đăng | 20-10-79 | Huế | <input checked="" type="checkbox"/> | 2000000 |
| K25A | a009 | Nguyễn Tiêu | 10-10-78 | Đà Nẵng | <input type="checkbox"/> | 1000000 |
| | a006 | Nguyễn Khánh | 10-05-75 | Đà Nẵng | <input checked="" type="checkbox"/> | 3000000 |
| | a008 | Trần Thanh | 25-07-77 | Huế | <input checked="" type="checkbox"/> | 3000000 |
| K25B | a007 | Lê Thuỷ Thu | 03-03-79 | Quảng Nam | <input type="checkbox"/> | 2000000 |

BÀI SỐ 8

Cho CSDL **QLSVIEN.MBD** gồm các bảng dữ liệu sau:

DSSVIEN(MASV, MALOP, HOTENSVIEN, NGAYSINH, GTINH, QQUAN)

DSMONHOC(MAMON, TENMON, SOH_TRINH)

DSDIEMTHI(MASV, MAMON, DIEM_LAN1, DIEM_LAN2)

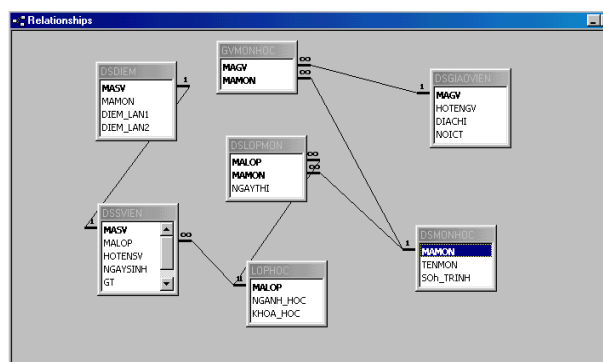
DSGIAOVIEN(MAGV, HOTENGVIEI, DIACHI, NOI_CT)

DSLOPHOC(MALOP, NGANH_HOC, KHOA_HOC)

GVMONHOC(MAGV, MAMON)

DSLOPMON(MALOP, MAMON, NGAYTHI)

Mối quan hệ giữa các bảng dữ liệu theo sơ đồ sau:



1. Tạo các biểu mẫu để cập nhật dữ liệu cho các bảng trên, chú ý sử dụng combo box hoặc List box đối với các trường cần thiết.
2. Tạo báo cáo để hiển thị danh sách sinh viên trong một lớp học nào đó, bao gồm cả ngành học tương ứng.
3. Tạo báo cáo để hiển thị danh sách sinh viên trong từng lớp học, bao gồm cả ngành học, trong đó các sinh viên cùng một lớp thì tên lớp được nhóm lại với nhau.
4. Tạo báo cáo để hiển thị danh sách giáo viên đã giảng dạy các môn học cho một lớp nào đó.
5. Tạo báo cáo để hiển thị danh sách giáo viên đã tham gia giảng dạy các môn học cho nhiều lớp học, trong đó các giáo viên giảng dạy các môn học cho một lớp học được nhóm lại với nhau.
6. Tạo báo cáo để hiển thị điểm thi các môn học của một sinh viên.
7. Tạo báo cáo để hiển thị điểm thi một môn học của một lớp nào đó.