



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

PHÁT TRIỀN ỨNG DỤNG TRONG QUẢN LÝ



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

THÔNG TIN LIÊN HỆ

- Họ tên giảng viên: TS.GVC Trịnh Hoài Sơn
- Đơn vị: Viện Công nghệ thông tin và Kinh tế số
- Địa chỉ: Phòng 1308 1312, Tầng 13, Tòa nhà A1
- Website: https://sitde.neu.edu.vn/
- Điện thoại:
- Email: sonth@neu.edu.vn



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MÔ TẢ HỌC PHẦN

- Học phần Phát triển các ứng dụng trong quản lý là học phần dành cho sinh viên chuyên ngành được bố trí vào học kỳ thứ 6 trong chương trình đào tạo.
- Học phần được tổ chức thành 8 chương, trong đó chương 1 trình bày các vấn đề lý thuyết liên quan đến xây dựng hệ thống thông tin ứng dụng trong quản lý tại các tổ chức, doanh nghiệp. Chương 2 đến chương 8 trình bày các nội dung liên quan đến lý thuyết đến kỹ năng xây dựng một phần mềm thương mại ứng dụng trong quản lý.





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỤC TIÊU CỦA HỌC PHẦN

Trang bị cho sinh viên lý thuyết và kỹ năng phân tích, thiết kế, xây dựng và đóng gói một phần mềm ứng dụng trong quản lý trên cơ sở ngôn ngữ lập trình bậc cao được sử dụng phổ biến trong thực tế.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

PHÂN BỔ THỜI GIAN

		Tổng số	Trong đó		Ghi chú
STT	Nội dung		Lý thuyết	Bài tập, thảo luận, kiểm tra	
1	Chương 1: Tổng quan về phát triển các ứng dụng trong quản lý		3	0	
2	Chương 2: Công cụ phát triển ứng dụng		2	2	Phòng máy tính có cài
3	Chương 3: Xây dựng giao diện cho ứng dụng trong quản lý	5	dặt công cụ phát triển ứng dụng MS Access, Visual Basic.NET,		
4	Chương 4: Lập trình liên kết và khai thác cơ sở dữ liệu		4	6	Crystal Report
5	Chương 5: Tạo báo cáo đầu ra cho ứng dụng trong quản lý	6	2	4	
6	Chương 6: Vấn đề an toàn và bảo mật của các ứng dụng trong quản lý	3	3	0	
7	Chương 7: Biên soạn tài liệu cho ứng dụng trong quản lý	4	2	2	
8	Chương 8: Tạo bộ cài đặt cho ứng dụng trong quản lý	3	1	2	
	Công	38	19	19	



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

PHƯƠNG PHÁP ĐÁNH GIÁ HỌC PHẦN

- Chuyên cần: 10%.
- Bài tập lớn: 30%.
 - Bài tập lớn được giao ngay từ khi bắt đầu môn học và được chấm điểm khi kết thúc môn học.
- Thi cuối học phần: 60%.
 - Sinh viên đủ tư cách dự thi học phần nếu có điểm bài tập lớn từ 5 điểm trở lên.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TÀI LIỆU THAM KHẢO

- 1. Trần Thị Song Minh (Chủ biên) (2011), Hệ thống thông tin quản lý, NXB Đại học Kinh tế Quốc dân.
- 2. Hàn Viết Thuận(2010), Giáo trình Kỹ nghệ phần mềm, NXB Thống kế
- 3. Trần Công Uẩn(2005), Giáo trình Cơ sở dữ liệu 1,2, NXB Thống Kê
- 4. Phạm Phương Hoa (2017), Lập trình với C# xây dựng ứng dụng, NXB Thanh niên
- 5. Dương Quang Thiện (2008), Lập trình báo cáo dung Crystal Report.NET, NXB Tổng hợp TP HCM
- 6. Sách điện tử do giảng viên cung cấp

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CHƯƠNG 1

TỔNG QUAN VỀ PHÁT TRIỂN CÁC ỨNG DỤNG TRONG QUẢN LÝ





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỤC ĐÍCH

Cung cấp những thông tin mang tính lý thuyết về phần mềm ứng dụng, lợi ích của việc trang bị phần mềm ứng dụng trong quản lý và các vấn đề liên quan đến phát triển phần mềm

CKINH TE QUOC ON

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

- 1.1 Lợi ích và thực trạng của việc tin học hoá công tác quản lý tại các doanh nghiệp Việt Nam
 - 1.1.1 Lợi ích của việc tin học hoá công tác quản lý
 - Lợi ích
 - Những thách thức
 - 1.1.2 Thực trạng của việc tin học hoá công tác quản lý tại Việt Nam
 - Giai đoạn 1: Những năm 90 của thế kỷ trước
 - Giai đoạn 2: Trước năm 2000
 - Giai đoạn 3: Từ 2000 đến nay



- 1.2 Các lý do xây dựng một HTTT mới
- Những vấn đề về quản lý
- Những yêu cầu mới của nhà quản lý
- Sự thay đối của công nghệ
- Sự thay đổi về sách lược chính trị



- 1.3 Khái niệm phần mềm và những vấn đề liên quan
 - 1.3.1 Khái niệm phần mềm
 - 1.3.2 Phân loại phần mềm
 - 1.3.3 Đặc điểm của phần mềm
 - 1.3.4 Các tiêu thức để đánh giá 1 phần mềm tốt



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Khái niệm phần mềm

 Phần mềm là một tập hợp bao gồm các chương trình, cấu trúc dữ liệu và tài liệu hướng dẫn sử dụng các chương trình ấy.

(Hàn Viết Thuận, 2010)



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Phân loại phần mềm

- Phần mềm hệ thống
 - Hệ điều hành
 - Phần mềm điều khiển thiết bị
 - Chương trình dịch
 - Phần mềm tiện ích
- Phần mềm ứng dụng
 - Phần mềm năng suất
 - Phần mềm kinh doanh
 - Phần mềm giáo dục tham khảo
 - Phần mềm giải trí



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Đặc điểm riêng có của phần mềm

- So với các sản phẩm thông thường khác, sản phẩm phần mềm có những đặc điểm riêng có:
 - Được tạo ra từ siêu nguyên liệu
 - Không bị khấu hao hữu hình theo thời gian
 - Có thể nhân bản mà không tốn thêm chi phí
 - Không có khối lượng, vận chuyển dễ dàng
 - · ...

CCKINH TÉ QUOCO DAN

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Các tiêu thức đánh giá phần mềm tốt

- Theo quan điểm của công ty phần mềm
 - Có đầy đủ chức năng theo yêu cầu của khách hàng
 - Hoạt động ổn định, ít lỗi ngẫu nhiên
 - Cho phép hiệu chỉnh, tùy biến dễ dàng
 - Giao diện thân thiện, dễ sử dụng
 - Tốc độ làm việc nhanh
 - Tài liệu hệ thống đầy đủ
- Theo quan điểm của người dùng cuối
 - Có đầy đủ chức năng theo yêu cầu
 - Giao diện thân thiện, dễ sử dụng
 - Tốc độ làm việc nhanh
 - Có đầy đủ tài liệu hướng dẫn sử dụng

ACKINH TE QUOC

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

- 1.4 Các bước tổng quát của quy trình phân tích, thiết kế và xây dựng phần mềm:
- Đánh giá yêu cầu
- Phân tích chi tiết
- Thiết kế logic
- Đề xuất các phương án của giải pháp
- Thiết kế vật lý ngoài
- Triển khai kỹ thuật hệ thống
- Cài đặt và khai thác hệ thống

THE QUOC OF A

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

1.5 Một số nguy cơ đối mặt trong quá trình sản xuất phần mềm

- Không có phương pháp mô tả rõ ràng
- Khó đáp ứng nhu cầu thay đổi người dùng một cách kịp thời
- Không có chuẩn trong việc thiết kế tài liệu
- Không thực hiện việc kiểm thử tính đúng đắn của phần mềm
- Coi trọng lập trình hơn thiết kế
- Nhiều thao tác do con người thực hiện
- Không chứng minh được tính đúng đắn của phần mềm
- Tiêu chuẩn đánh giá một phần mềm tốt không thể định lượng hóa
- Đầu tư nhân lực lớn vào việc bảo trì phần mềm
- Công việc bảo trì kéo dài
- Việc quản lý dự án lỏng lẻo
- Không có tiêu chuẩn ước lượng nhân lực của dự án

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CHƯƠNG 2 NGÔN NGỮ LẬP TRÌNH C#



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỤC ĐÍCH

- Giới thiệu tổng quan về ngôn ngữ lập trình C#;
- Hướng dẫn sử dụng cơ bản phần mềm Visual Studio .NET sử dụng để phát triển ứng dụng bằng ngôn ngữ C#.
- Cung cấp thông tin về các kiểu dữ liệu và các cấu trúc điều khiển thông dụng của C# sử dụng trong phát triển một phần mềm ứng dụng trong quản lý



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.1. Giới thiệu chung

- 2.1.1 Giới thiệu visual c#
 - Nguồn gốc
 - Ưu điểm
 - Các phiên bản
- 2.1.2. Giao diện làm việc của c#
 - Thanh Menu
 - Project Explorer
 - Properties window
 - Design window
 - Coding window





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.1.1 GIỚI THIỆU C#

- C# được phát triển bởi nhóm tác giả điều hành bởi Anders Hejlsberg
- C# được dẫn xuất từ C và C++ nhưng nó được tạo từ nền tảng tốt hơn
- C# là ngôn ngữ hướng đối tượng (Object Oriented Language)
 - Hỗ trợ định nghĩa và làm việc với lớp (class)
 - Hỗ trợ đầy đủ các đặc trưng của lập trình hướng đối tượng: đóng gói (encapsulation), kế thừa (inheritance) và đa hình (polymorphism)
- C# cung cấp những đặc tính hướng thành phần (component oriented): thuộc tính, sự kiện

CKINH TE QUOC

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

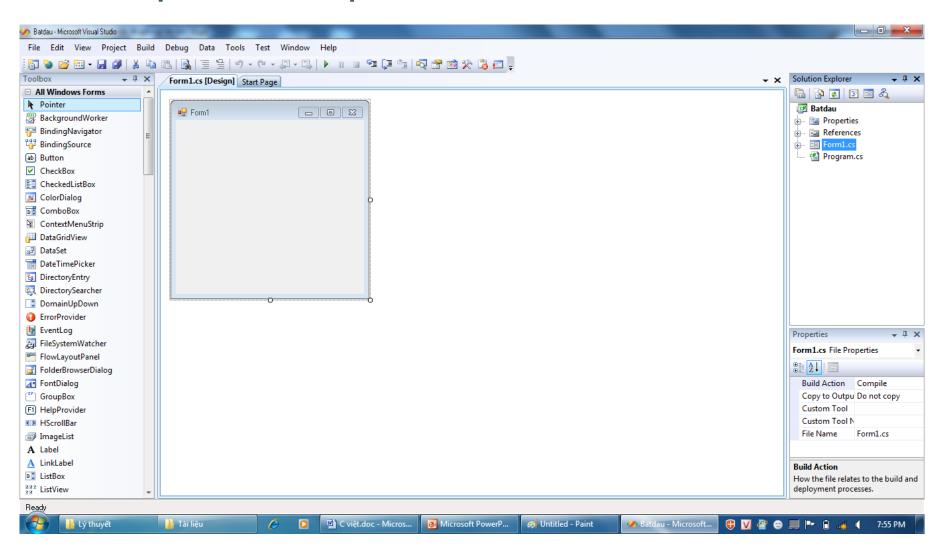
GIỚI THIỆU C#

- C# là ngôn ngữ đơn giản
- C# là ngôn ngữ hiện đại
- C# là ngôn ngữ hướng đối tượng
- C# là ngôn ngữ mạnh mẽ và mềm dẻo
- C# là ngôn ngữ có ít từ khoá
- C# là ngôn ngữ hướng module
- C# phổ biến



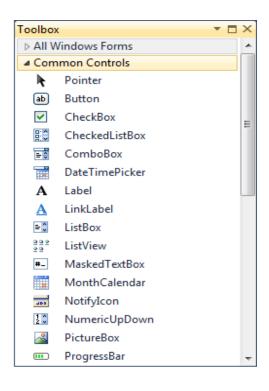
VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

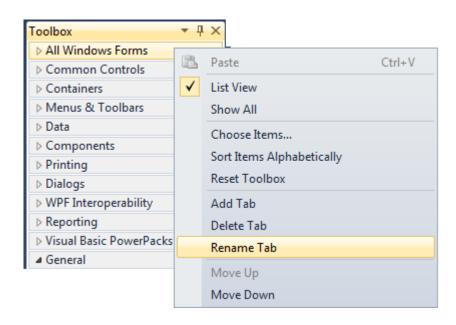
2.1.2 GIAO DIỆN LÀM VIỆC





- Toolbox: chứa các điều khiển khi thiết kế giao diện người dùng
 - Ån/hiện: View → Toolbox

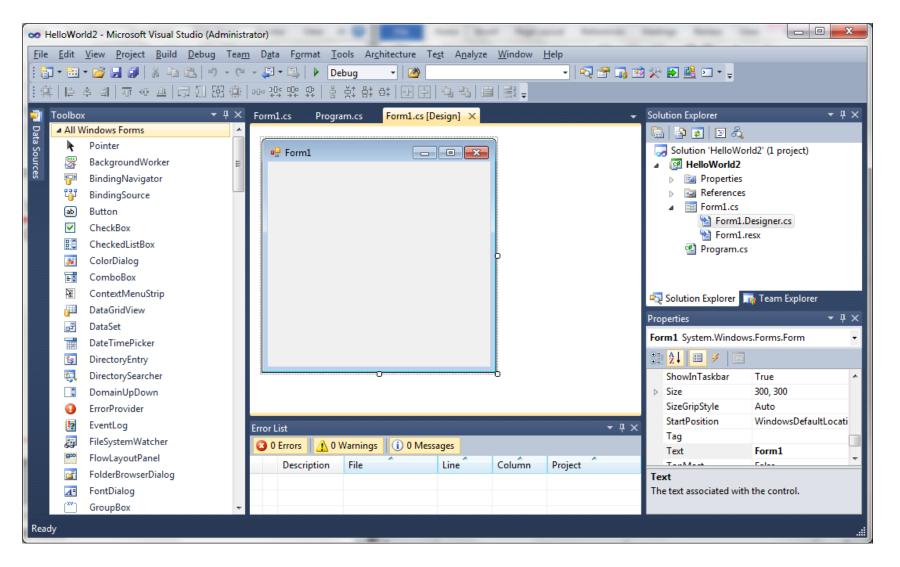






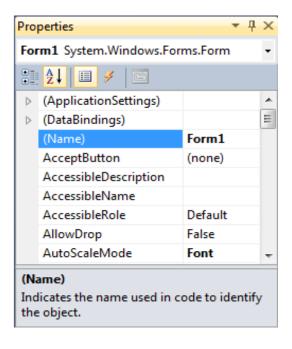
VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.1.2 GIAO DIỆN LÀM VIỆC





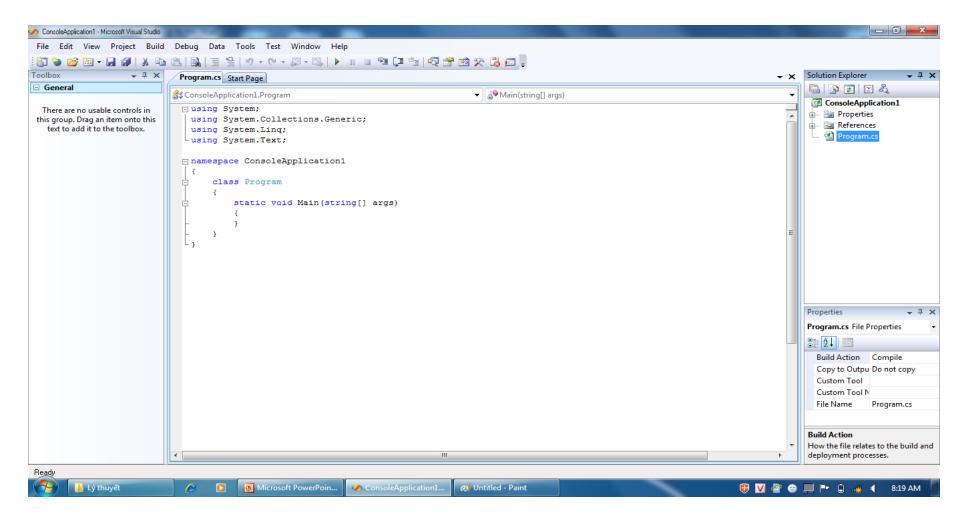
- Properties Window
 - Liệt kê tất cả các thuộc tính, sự kiện của các điều khiển
 - View → Properties Window





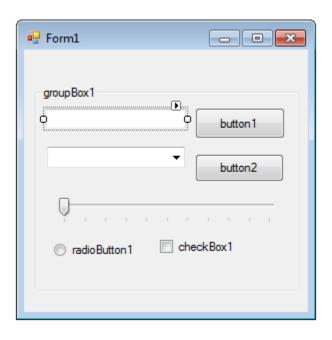
VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.1.2 GIAO DIỆN LÀM VIỆC

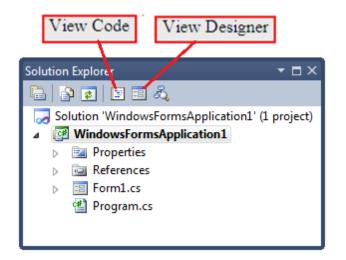




- Form Desinger
 - Cho phép thiết kế giao diện người dùng, sử dụng các điều khiển trong Toolbox

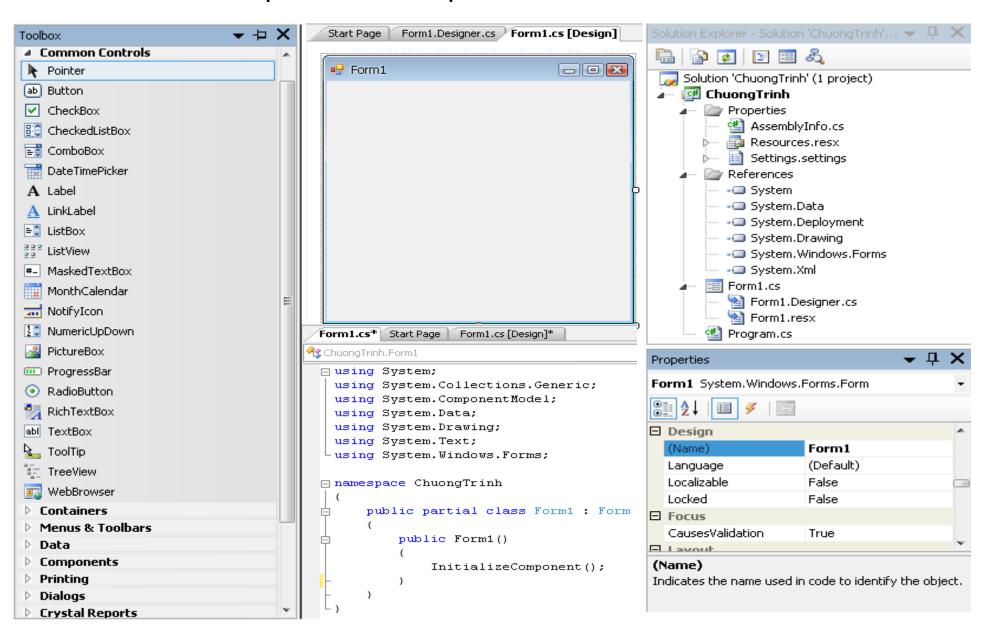


- Solution Explorer
 - View → Solution Explorer



- View Code: Xem mã nguồn
- View Designer: Xem thiết kế giao diện

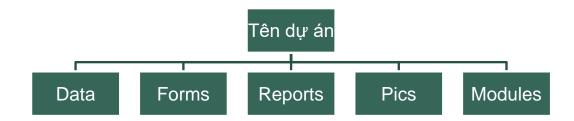




THE QUOC OF A

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

- 2.1.3. Dự án phần mềm(Project) và các thành phần của một dự án phần mềm
- Khái niệm
- Các thành phần
 - Form (giao diên)
 - Báo cáo (Report)
 - CSDL (Database)
 - Thư viện hàm và thủ tục(Module)
- Xây dựng cây thư mục cho dự án



THE QUOC OF THE QUOC

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.1.4. Một số thao tác cơ bản trong c#

- Cài đặt Visual Studio
- Cài đặt bản nâng cấp
- Tạo 1 dự án mới
- Thêm mới các thành phần của dự án
- Thêm các điều khiển vào dự án
- Thêm các đối tượng có sẵn vào dự án
- Mở cửa số truy suất các thành phần của dự án (Project Explorer)
- Thay đổi thuộc tính của một đối tượng trong cửa sổ thuộc tính
- Lập trình cho các chức năng của dự án
- Chạy thử dự án



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Một số thao tác cơ bản

- Mở chương trình
 - File → Open → Project/Solution...
- Lưu chương trình
 - File -→ Save All
 - Khi tạo đồ án, Visual Studio sinh ra các tập tin:
 - *.sln (solution): mỗi ứng dụng có 1 tập tin loại này, bao gồm một hoặc nhiều tập tin dự án
 - *.csproj (project): mỗi dự án có một hoặc nhiều tập tin nguồn (các tập tin trong cùng dự án có cùng ngôn ngữ)
 - *.cs (source): chứa mã nguồn chương trình
 - AssemblyInfor.cs: chứa các thuộc tính

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Một số thao tác cơ bản

- Biên dịch
 - Dịch Solution: Build → Build Solution (F6)
 - Dịch dự án: Build → Build tên dự án (Shift + F6)
- Chạy chương trình
 - Chạy chương trình kèm sửa lỗi:

Debug → Start Debugging (F5 hoặc nhấn



- Chạy chương trình không kèm sửa lỗi:
 - Debug → Start Without Debugging (Ctrl + F5)
- Dùng chương trình
 - Debug → Stop Debugging (Ctrl + Alt + Break





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.2 CÁC CẤU TRÚC ĐIỀU KHIỂN TRONG VISUAL C#

- 2.2.1. Một số kiểu dữ liệu thường dung
- Dữ liệu kiểu số nguyên

Name	CTS Type	Description	Range (min:max)
sbyte	System.SByte	8-bit signed integer	-128:127 (-27:27-1)
short	System.Int16	16-bit signed integer	-32,768:32,767 (-215:215-1)
int	System.Int32	32-bit signed integer	-2,147,483,648:2,147,483,647 (-231:231-1)
			-9,223,372,036,854,775,808: 9,223,372,036,854,775,807 (-
long	System.Int64	64-bit signed integer	263:263-1)
byte	System.Byte	8-bit signed integer	0:255 (0:28-1)
ushort	System.UInt16	16-bit signed integer	0:65,535 (0:216-1)
uint	System.UInt32	32-bit signed integer	0:4,294,967,295 (0:232-1)
ulong	System.UInt64	64-bit signed integer	0:18,446,744,073,709,551,615(0:264-1)



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Dữ liệu kiểu số thập phân

Name	CTS Type	Description	Range (approximate)		
decimal	System.Decimal	128-bit high precision decimal notation	$\pm 1.0 \times 10^{-28}$ to $\pm 7.9 \times 10^{28}$		

Kiểu dữ liệu số dấu chấm di động (Floating Point Types)

Name	CTS Type	Description	Range (approximate)
Float	System.Single	32-bit single-precision floating- point	±1.5 × 10-45 to ±3.4 × 1038
Double	System.Double	64-bit double-precision floating- point	±5.0 × 10-324 to ±1.7 × 10308

AC KINH TE QUOC DA

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

- Kiểu Boolean (Bool)
- Kiểu ngày tháng (DateTime)
- Kiểu Character (char)
- Kiểu String (string)
 - Chú ý:Khi in ra màn hình dữ liệu kiểu string, chúng ta cần quan tâm đến các ký tự "escape"- các ký tự này sẽ định nghĩa cách chuỗi được in ra. Các kỹ tự escape được bắt đầu bởi '\' và theo sau là một ký tự.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỘT SỐ KÝ TỰ ESCAPE THÔNG DỤNG

- \' Thêm vào một dấu nháy đơn
- \" Thêm vào một dấu nháy kép
- \\ Thêm vào ký tự xuyệt ('\')
- \a Kích hoạt một cảnh báo hệ thống
- \b Lùi con trỏ về bên trái 1 vị trí
- \n Thêm một dòng mới
- \r Chuyển qua đoạn văn mới
- \t Thêm một tab
- Ví dụ: string s = "I am \'Sơn Trịnh \' ";

Console.WriteLine(s);

Kết quả là: I am 'Sơn Trịnh '

C KINH TE QUOC

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Một số lưu ý khi sử dụng các kiểu dữ liệu

Kiểu dữ liệu có miền giá trị lớn hơn sẽ chứa được kiểu dữ liệu có miền giá trị nhỏ hơn.
 Như vậy biến kiểu dữ liệu nhỏ hơn có thể gán giá trị qua biến kiểu dữ liệu lớn hơn (sẽ được trình bày trong phần tiếp theo).

Giá trị của kiểu char sẽ nằm trong dấu ' ' (nháy đơn).

Giá trị của kiểu string sẽ nằm trong dấu "" (nháy kép).

Giá trị của biến kiểu float phải có chữ F hoặc f làm hậu tố.

int a; a=10;

long b=10L;

float c = 19.34F;

double d=19.34;

- Giá trị của biến kiếu decimal phải có chữ m hoặc M làm hậu tố.
 Một bằng cố nguyên cũng có các bậu tố (cuffix) Li bay Li thể biên kiểu uncique
- Một hằng số nguyên cũng có các hậu tố (suffix) U hay L thể hiện kiểu unsigned hay long.
- Trừ kiếu string, tất cả kiểu dữ liệu trên đều không được có giá trị null:
 - Null là giá trị rỗng, không tham chiếu đến vùng nhớ nào.
 - Để có thể gán giá trị null cho biến thì ta thêm ký tự? vào sau tên kiểu dữ liệu là được. Ví dụ: int? hay bool?



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC PHÉP TOÁN SỐ HỌC:

- *Cộng*: +,+=
- *Trù*: -, -=
- *Nhân*: *, *=
- Chia:/, /=

Chú ý: các phép toán x[+,-,*,/]=y tương đương với phép toán: x=x[+,-,*,/]y

- %: phép chia lấy phần dư, ví dụ 5 % 2=1,
 - **1**0.8 % 4 = 2.8



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC TOÁN TỬ SO SÁNH

- Kết quả trả về của các phép toán so sánh có dạng đúng hoặc sai.
- + Lớn hơn: >
- + Nhỏ hơn: <
- + Lớn hơn hoặc bằng: >=
- + Nhỏ hơn hoặc bằng: <=
- + Bằng: ==
- + Khác: !=



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC TOÁN TỬ LOGIC

- + Phép toán Và &&: xét biểu thức A && B chỉ nhận giá trị đúng khi và chỉ khi cả A và B cùng nhận giá trị đúng, còn nhận giá trị sai trong tất cả các trường hợp còn lại
- + Phép toán **Hoặc** //: xét biểu thức A | | B chỉ nhận giá trị sai khi và chỉ khi cả A và B cùng nhận giá trị sai, còn nhận giá trị đúng trong tất cả các trường hợp còn lại.
- + Phép toán Phủ định !: ta có !A nhận giá trị đúng khi A nhận giá trị sai và ngược lại.

C KINH TË QUOC O P

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC HÀM TOÁN HỌC

- + Math.Round(x,n): hàm làm tròn số thực, ví dụ: Math.Round(11.346,2) = 11.35
- + Math.Abs(x): hàm tính giá trị tuyệt đối của x
- + Math.Sqrt(x): hàm tính căn bậc hai của x
- + Math.Floor(x): hàm trả về số nguyên gần x nhất, ví dụ: Math.Floor(11.756) = 11
- + Math.Truncate(x): hàm trả về phần nguyên của x, ví dụ: Math.Truncate(11.756)
 = 11
- Vd:Math.Floor(-11.756) = -12 và Math.Truncate(-11.756) = -11

OCKINH TE QUOC

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

- +s.Length: trả về chiều dài của chuỗi s.
- CÁC HÀM CHUỖI KÝ TỰ
- ■Ví dụ: "Lâm Anh".Length có kết quả = 7
- +s.Replace(str1,str2): thay thế chuỗi str1 trong chuỗi s bằng chuỗi str2.
- ■Ví dụ: "Hà Nội".Replace("Nội", "Tây") cho kết quả "Hà Tây"
- +s.Substring(vt,n): trả về một chuỗi con gồm n ký tự trong chuỗi s, bắt đầu từ ký tự ở vị trí vt. (Chuỗi được tính từ vị trí 0).
- ■Ví dụ: "Hoa hồng ".Substring(1, 2) cho kết quả "oa"
- +s. Insert(vt, str): chèn thêm giá trị của chuỗi str vào chuỗi s tại vị trí vt.
- ■Ví dụ: "Trời xanh". Insert(4, " màu") cho kết quả "Trời màu xanh"
- +s. ToLower. biến đổi chuỗi s về chữ in thường.
- ■Ví dụ: "Hà Nội".ToLower cho kết quả "hà nội"
- +s.ToUpper. biến đổi chuỗi s về chữ in hoa.
- ■Ví dụ: "Hà Nội".ToUpper có kết quả "HÀ NỘI"

C KINH TE QUOC

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

- +s.Remove(vt,n): xóa n ký tự trong chuỗi s, bắt đầu từ ký tự ở vị trí vt.
- ■Ví dụ: "Hoa hồng".Remove(1, 2) cho kết quả "H hồng"

CÁC HÀM CHUỖI KÝ TỰ (TIẾP)

- +s.TrimStart. xóa các ký tự rỗng ở đầu chuỗi s.
- Ví dụ: " Hoa hồng ".TrimStart cho kết quả "Hoa hồng "
- +s.TrimEnd: xóa các ký tự rỗng ở cuối chuỗi s.
- ■Ví dụ: "Hoa hồng ".TrimEnd cho kết quả "Hoa hồng"
- +s. Trim: xóa các ký tự rỗng ở đầu và cuối chuỗi s.
- ■Ví dụ: "Hoa hồng ".Trim cho kết quả "Hoa hồng"
- +s.Split(ch): tách chuỗi s thành các chuỗi con ngăn cách nhau bởi ký tự ch.
- Ví dụ: string s = "Ha Noi";
- string[] tu=s.Split('');
- Kết quả: tu[0]= "Ha", tu[1]= "Noi"



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC HÀM NGÀY THÁNG

- + DateTime.Now: trả về ngày và giờ hiện hành, ví dụ: 09/02/2009 5:20:28PM
- + Date.Day;Date.Month; Date.Year. trả về giá trị ngày, tháng, năm của Date, ví dụ: d.Year cho kết quả 2009
- + Date.AddDays(n): trả về một ngày mới cách ngày Date n ngày.
- + Date.AddMonths(n): trả về một ngày mới cách ngày Date n tháng.
- + Date.AddYears(n): trả về một ngày mới cách ngày Date n năm.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

HÀM CHUYỂN ĐỔI KIỂU DỮ LIỆU

Hàm chuyển đổi	Đổi giá trị sang kiểu
Convert.ToBoolean(Giatri)	Boolean
Convert.ToByte(Giatri)	Byte
Convert.ToDateTime(Giatri)	Date
Convert.ToDouble(Giatri)	Double
Convert.ToInt16(Giatri)	Integer – 2 byte
Convert.ToInt32(Giatri)	Integer – 4 byte
Convert.ToInt64(Giatri)	Integer – 8 byte
Convert.ToSingle(Giatri)	Single
Convert.ToString(Giatri)	String

Chú ý: giá trị truyền cho hàm phải hợp lệ, nghĩa là phải thuộc miền giá trị của kiểu kết quả nếu không C# sẽ báo lỗi. Ví dụ:

- + Convert.ToDateTime("25/04/79") trả về giá trị kiểu ngày tháng 25/04/79
- + Convert.ToInt32("25") = 25
- + Convert.ToInt32("25a") hoặc Convert.ToInt32("a25") sẽ báo lỗi.

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.2.2. KHAI BÁO HẰNG, KHAI BÁO BIẾN TRONG C#

Biến có thể được khai báo tại 2 nơi gồm:

- Bên trong phần định nghĩa lớp của Form.
- Bên trong một phương thức.

```
Ví dụ:

public class frmWelcome : Form

{

// Nơi khai báo biến;

private void bntDisplay_Click(object sender, EventArgs e)

{

// Nơi khai báo biến;

}
```

Phạm vi của một biến: phụ thuộc vào vị trí khai báo của biến, nếu biến được khai báo trong phần định nghĩa lớp của Form thì là biến toàn cục có tác dụng trong toàn bộ các đoạn mã lệnh của form, nếu biến được khai báo bên trong một phương thức thì là biến cục bộ chỉ có tác dụng trong phương thức chứa nó.

Cú pháp khai báo biến như sau:

```
Kiểu dữ liêu Tênbiến [=Giá tri];
```

Tênbiến: là một chuỗi các ký tự do người lập trình tự đặt bao gồm các chữ cái, chữ số và dấu gạch dưới. Tên biến phải bắt đầu bằng một chữ cái, không được chứa dấu cách, C# phân biệt chữ hoa chữ thường.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Hằng

Hằng là đối tượng dùng để chứa một giá trị không đổi trong suốt quá trình thực hiện chương trình.

Hằng được khai báo theo cú pháp sau:

const Kiểu dữ liệu Tênhằng = Giá trị;

C CUNH TE QUOC OR

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

KHAI BÁO BIẾN MẢNG

- Khai báo mảng: mảng được khai báo theo cú pháp sau:
- Kiểu dữ liệu[] tên mảng;
- Để cấp phát bộ nhớ cho mảng ta dùng toán tử new theo sau là tên kiểu dữ liệu và kích thước của mảng được đặt trong cặp dấu ngoặc vuông.
- Ví dụ khai báo mảng một chiều nguyên a gồm 10 phần tử:
 - int[] a = new int[10];
- Khai báo mảng 2 chiều thực b gồm 10 hàng, 5 cột
- double[,] b = new double[10,5];



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

KHỞI TẠO GIÁ TRỊ CHO CÁC PHẦN TỬ CỦA MẢNG KHI KHAI BÁO

double[] a = new double[2] {34.56, -45};
 hoặc double[] a = {34.56, -45};
 string[] Tennuoc = {"Anh", "Pháp", "Đức", "Việt Nam"};
 int[,] a = {{4, 6, 9}, {5, 7, 9}, {12, 44, 23}};



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.2.3 CÂU LỆNH RỄ NHÁNH TRONG C#

- Câu lệnh If

```
Dang 1:
if (Điềukiện)
Khối lệnh;
Hoạt động: Nếu <Điều kiện> nhận giá trị đúng thì <Khối lệnh> được thực hiện.
Dang 2:
if (Điều kiện)
Khối lệnh 1;
Else
       Khối lệnh 2;
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.2.3 CÂU LỆNH LỰA CHỌN CASE

```
switch (Biểu thức kiểm tra)
case <Biểu thức 1>:
  Khối lệnh 1;
    [break;]
case <Biểu thức 2>:
   Khối lệnh 2;
   [break;]
default:
   Khối lệnh n+1;
   [break;]
```

Hoạt động: máy so sánh giá trị của <Biểu thức kiểm tra> với giá trị của các <Biểu thức i>. Nếu <Biểu thức kiểm tra> có giá trị thỏa mãn <Biểu thức i> thì <Khối lệnh i> được thực hiện, sau đó máy sẽ thoát ngay ra khỏi câu lệnh switch.

Nếu <Biểu thức kiểm tra> không thỏa mãn <Biểu thức i> nào thì <Khối lệnh n+1> được thực hiện.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÂU LỆNH LỰA CHỌN CASE

Chú ý: nếu các biểu thức khác nhau cùng thực hiện chung một khối lệnh thì ta có thể viết gộp như sau:

```
switch (Biểu thức kiểm tra)
 case <Biểu thức 1>:
 case <Biểu thức 2>:
   Khối lệnh;
   break;
 default:
   Khối lệnh n+1;
   break;
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.2.4. LỆNH CHU TRÌNH TRONG C#

- Câu lệnh For

Quy cách:

```
for (biểu thức khởi tạo; biểu thức điều kiện; biểu thức cập nhật) {
Khối lệnh;
[break;]
}
```

Hoạt động: Đầu tiên máy thực hiện biểu thức khởi tạo để khởi tạo giá trị của biến điều khiển, sau đó máy kiểm tra giá trị của biểu thức điều kiện, nếu biểu thức này đúng thì <Khối lệnh> được thực hiện và cập nhật giá trị của biến điều khiển thông qua biểu thức cập nhật. Sau đó quay lại kiểm tra giá trị của biểu thức điều kiện, cứ lặp lại như vậy cho đến khi biểu thức điều kiện nhận giá trị sai thì dừng lại.

Chú ý: để thoát ngay ra khỏi vòng lặp for ta có thể dùng lệnh break.

OC KINH TE QUOC

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

```
Ví dụ:
static void Main(string[] args)
       int i;
       for (i=0; i<=3; i++)
          Console.Write(i);
       Console.ReadLine();
Kết quả: 0123
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.2.4. LỆNH CHU TRÌNH WHILE TRONG C#

```
while (Biểu thức điều kiện)
{
    Khối lệnh;
    [break;]
}
```

Hoạt động: Đầu tiên máy kiểm tra giá trị của <Biểu thức điều kiện>, nếu biểu thức này nhận giá trị đúng thì <Khối lệnh> được thực hiện. Sau đó lại quay lại kiểm tra giá trị của <Biểu thức điều kiện>, cứ lặp lại như vậy cho đến khi <Biểu thức điều kiện> nhận giá trị sai thì dừng lại.

Chú ý: + Vì <Biểu thức điều kiện> được kiểm tra trước, nên <Khối lệnh> có thể không được thực hiện lần nào nếu ngay từ đầu <Biểu thức điều kiện> đã nhận giá trị sai và trước khi thực hiện khối lệnh phải khởi gán giá trị cho <Biểu thức điều kiện>.

+ Trong <Khối lệnh> phải có ít nhất một lệnh làm thay đổi giá trị của <Biểu thức điều kiện> để đến một lúc nào đó <Biểu thức điều kiện> nhận giá trị sai, nhằm dừng vòng lặp lại, nếu không nó sẽ lặp mãi không dừng.

THE QUOC

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.3.4. LỆNH CHU TRÌNH DO ...WHILE TRONG C#

Cho phép thực hiện lặp đi lặp lại một đoạn chương trình nhiều lần, với số lần lặp không được xác định trước.

```
Do
{
Khối lệnh;
[break;]
}
while (Biểu thức)
```

- while (Biểu thức điều kiện);
- Hoạt động: Đầu tiên máy thực hiện <Khối lệnh>, sau đó kiểm tra giá trị của <Biểu thức điều kiện>, nếu biểu thức này nhận giá trị đúng thì tiếp tục thực hiện <Khối lệnh>, cứ lặp lại như vậy cho đến khi <Biểu thức điều kiện> nhận giá trị sai thì dừng lại.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

- Chú ý: + Vì <Biểu thức điều kiện> được kiểm tra sau, nên <Khối lệnh> luôn được thực hiện
- ít nhất 1 lần.
- + Trong <Khối lệnh> phải có ít nhất một lệnh làm thay đổi giá trị của <Biểu thức điều kiện> nhằm dừng vòng lặp lại.
- Ví dụ: Tính tổng T = 1 + 2 + ... + 10



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.2.5 MỘT SỐ HÀM KIỂM TRA TRONG C#

- IsDate (<bieu thuc>) = true nếu biểu thức có kiểu ngày tháng
- IsEmpty (<bien>) = true nếu biến chưa được khởi tạo
- IsError (<bieu thuc>) = true n\u00e9u bi\u00e9u th\u00fcc c\u00f3 l\u00f3i
- IsNull (<bien/ten truong>) = true n\u00e9u gi\u00e1 tri l\u00e0 null
- IsNumeric (<bieu thuc>) = true n\u00e9u bi\u00e9u th\u00fcc c\u00f3 ki\u00e9u s\u00f3.

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Chương 3

Xây dựng giao diện ứng dụng





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỤC ĐÍCH

Trang bị kiến thức và hướng dẫn sinh viên thực hành khai báo, thiết kế và xây dựng hệ thống giao diện của phần mềm ứng dụng chuyên nghiệp đạt tiêu chuẩn thương mại hóa

C KINH TË QUOC O DE

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

3.1. ĐỐI TƯỢNG (CONTROL) VÀ THUỘC TÍNH

3.1.1. Khái niệm

Đối tượng(control)

- 3.1.2. Một số thuộc tính thường gặp của đối tượng
- Name _ Tên của đối tượng
 - Định nghĩa
 - Quy tắc đặt tên
 - Một số tiền tố sử dụng khi đặt tên

THE QUOC OF THE QUOC

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỘT SỐ TIỀN TỐ SỬ DỤNG KHI ĐẶT TÊN

- cbo _ ComboBox
- chk _ CheckBox
- cmd _ Command button
- frm _ form
- grd _ Grid
- img _ image
- Ibl _ label
- Ist _ list
- mnu _ menu
- opt _ nút tuỳ chọn Option button
- pic _ hộp chứa hình vẽ PictureBox
- tmr _ đối tượng thời gian Timer
- txt _ hộp văn bản TextBox

C KINH TÉ QUOC O A N

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

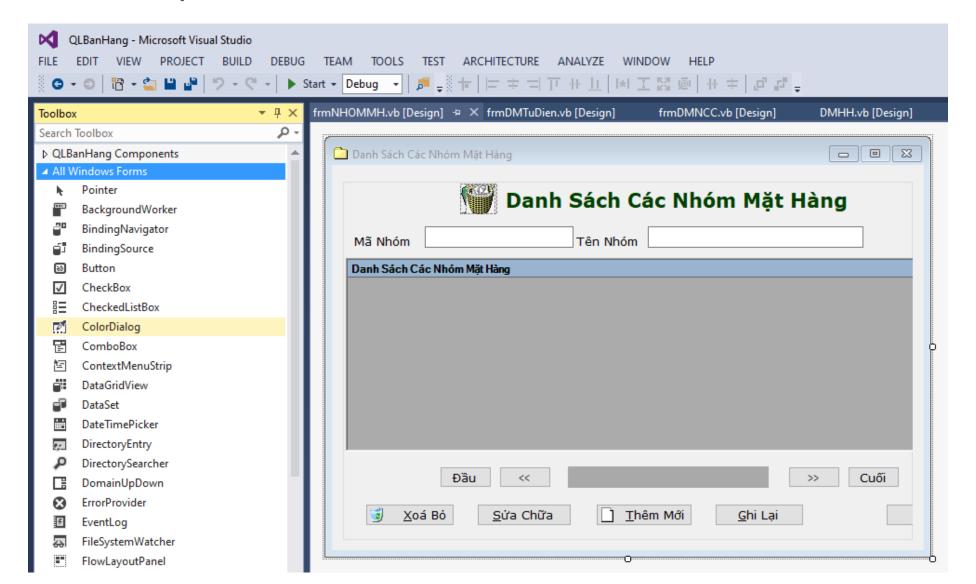
3.1.2. MỘT SỐ THUỘC TÍNH THƯỜNG GẶP CỦA ĐỐI TƯỢNG

- Thuộc tính tiêu đề _ Caption
- Thuộc tính khả ứng_Enable
- Thuộc tính Visible
- Thứ tự truy suất_Tab Index
- Giới thiệu ngắn_Tool Tip Text
- Định vị đối tương: top, left
- Kích thước đối tượng: Height, Width



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

3.2. MỘT SỐ ĐỐI TƯỢNG THƯỜNG DÙNG ĐỂ THIẾT KẾ FORM





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

FORM

Các thuộc tính

Thuộc tính	Mô tả
Name	Tên form, bắt đầu bằng frm
IsMdiContainer	Trạng thái SDI, MDI
BackColor	Màu nền
BackgroundImage	Hình nền trên form
AcceptButton	Nút xử lý mặc định – phím Enter
CancelButton	Nút xử lý mặc định – Phím Esc
FormBorderStyle	Đường viền cho form
Enabled	True/False: cho phép/không cho phép tác động
Font	Font chữ cho các điều khiển trên form
ForeColor	Màu cho các điều khiển trên form
Icon	Icon cho form



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

FORM

Các thuộc tính

Thuộc tính	Mô tả
MainMenuStrip	Menu chính
ContextMenu	Menu ngữ cảnh
Opacity	Độ trong suốt (0%-100%)
StartPosition	Vị trí xuất hiện
Text	Tiêu đề cho form
WindowState	Trạng thái của form khi chạy chương trình: Normal, Maximized, Minimized)
ShowInTaskbar	Hiển thị trên Taskbar
MaximizeBox	True/False: Có/không hiển thị nút phóng lớn
MinimizeBox	True/False: Có/không hiển thị nút thu nhỏ



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

FORM

Các sự kiện



- Nhấn để hiển thị danh sách các sự kiện
- Nháy đúp chuột vào tên sự kiện để vào cửa sổ viết mã lệnh

Sự kiện	Mô tả
Load	Khi form được nạp vào bộ nhớ, thường dùng khởi tạo giá trị và trạng thái cho các biến, điều khiển trên form
Click	Người dùng nhấn chuột
FormClosed	Khi người dùng nhấn nút Close x
FormClosing	Khi người dùng nhấn nút close x (trước sự kiện FormClosed)



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

FORM

Các phương thức

Phương thức	Mô tả
Close	Đóng form
Hide	Ẩn form
Show	Hiển thị form chế độ thông thường (modeless)
ShowDialog	Hiển thị form như hộp thoại (model)



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TEXTBOX

Thuộc tính

Thuộc tính	Mô tả
Name	Tên, bắt đầu bằng txt
BackColor	Màu nền
Enabled	True/False: bật/tắt textbox
Font	Font chữ
ForeColor	Màu chữ
Locked	True/false: khoá/không khoá
MaxLength	Chiều dài tối đa (0-32767)
Multiline	True: cho phép nhiều dòng
PasswordChar	Hiển thị kí tự này thay cho kí tự nhập vào
ReadOnly	Chỉ đọc



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TEXTBOX

Thuộc tính

Thuộc tính	Mô tả
ScrollBars	Thanh cuốn ngang, dọc (có hiệu lực khi Multiline = true, thanh cuốn ngang có hiệu lực khi WordWrap=False)
TabIndex	Thứ tự truy cập của hộp văn bản khi người dùng bấm phím Tab, thứ tự đầu tiên là 0.
Text	Chứa nội dung của hộp văn bản.
TextAlign	Căn lề
Visible	True/False: ẩn/hiện
WordWrap	WordWrap = True: dòng văn bản được tự động cuộn xuống dòng khi gặp lề bên phải của hộp TextBox, ngược lại thì nhận giá trị False. Chỉ có hiệu lực khi Multiline = True.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TEXTBOX

Sự kiện

Sự kiện	Mô tả
TextChanged	Có sự thay đổi trong hộp văn bản
Click	Nháy chuột
DoubleClick	Nháy đúp chuột
GotFocus	Khi chuyển tiêu điểm tới hộp văn bản.
KeyPress	Trả về ký tự (trừ các ký tự đặc biệt như phím Delete, Home, Ctrl, Fl) mà người sử dụng gõ vào hộp văn bản thông qua thuộc tính KeyChar.
KeyDown	Trả về mã ASCII của tất cả các ký tự mà người sử dụng gõ vào hộp văn bản thông qua thuộc tính KeyCode.
LostFocus	Được kích hoạt khi hộp văn bản mất tiêu điểm.
MouseMove	Di chuyển qua hộp văn bản
MouseLeave	Di chuyển ra khỏi hộp văn bản



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TEXTBOX

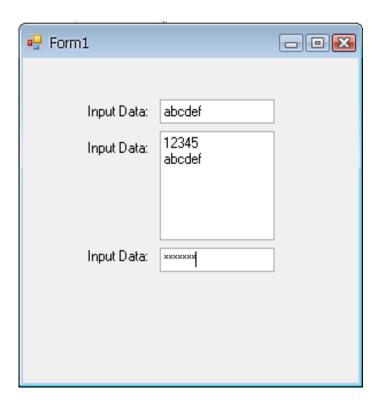
Phương thức

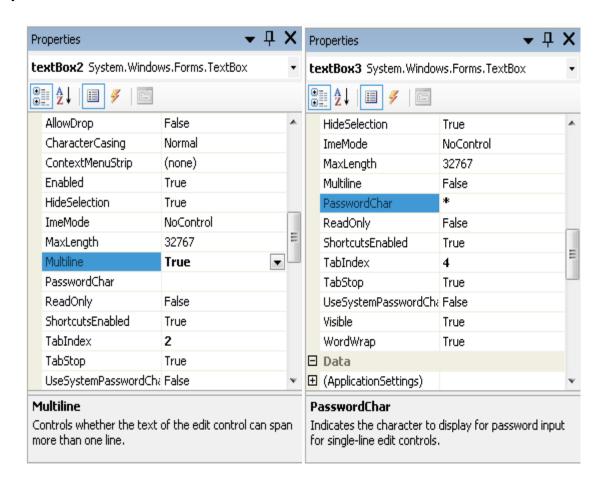
Phương thức	Mô tả
AppendText	Cộng dồn chuỗi
Clear	Xoá nội dung trên Textbox
Сору	Sao chép
Cut	Cắt
Paste	Dán
Undo	Quay về trạng thái cũ
GetCharIndexFromPosition	Lấy kí tự tại vị trí
Select	Chọn



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TEXTBOX





```
private void textBox1_TextChanged(object sender, EventArgs e)
{
    //Code Here
}
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TEXTBOX – VÍ DỤ

Hiển thị mã ASCII của kí tự bất kì nhập vào Textbox1

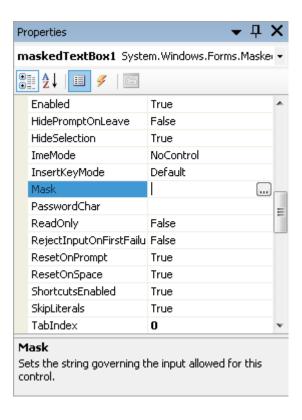
```
private void textBox1_KeyDown(object sender, KeyEventArgs e)
    {
        int a;
        a = Convert.ToInt32(e.KeyCode);
        MessageBox.Show(a.ToString());
}
```

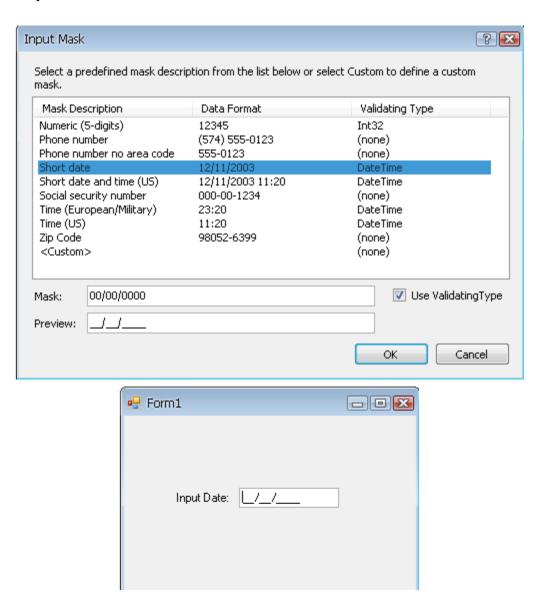
Chỉ cho phép nhập số 0-9, dấu -, chấm ., phím del (mã ASCII 13) và backspace (mã ASCII 8)



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MASKEDTEXTBOX



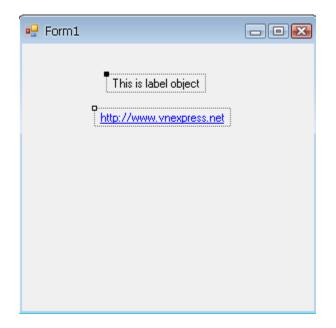




VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

LABEL - NHÃN

- Label:
 - Trình bày, chú giải tiêu đề
 - Có hai loại Label và LinkLabel
 - Name: bắt đầu bằng lbl



```
Label lb = new Label();
lb.Text = "This is Label Object";
this.Controls.Add(lb);
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

BUTTON - NÚT NHẤN

- Dùng chuột nhấn để thao tác
- Có thể show Icon trên button



```
private void btSearch_Click(object sender, EventArgs e)
{
    //Code Here
}
```

Θ	Action	
	Click	
	DoubleClick	
	MouseCaptureChang	
	MouseClick	
	MouseDoubleClick	
	ResizeBegin	
	ResizeEnd	
	Scroll	



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

BUTTON

Thuộc tính

Thuộc tính	Mô tả
Name	Tên, bắt đầu bằng btn
BackColor	Màu nền
BackgroundImage	Ảnh nền
Enabled	True/False: bật/tắt nút lệnh
Font	Font chữ
ForeColor	Màu chữ
Image	Ảnh trên nút
Locked	True/False: khoá/không khoá dịch chuyển vị trí nút
TabIndex	Thứ tự truy cập (khi nhấn Tab)
Text	Tiêu đề nút (Thêm dấu & trước kí tự đặt phím nóng)
Visible	True/False: Hiện/Ẩn nút lệnh



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

BUTTON

Sự kiện

Sự kiện	Mô tả
Click	Nhấn nút
GotFocus	Chuyển tiêu điểm tới nút
LostFocus	Mất tiêu điểm
MouseDown	Đặt chuột vào nút
MouseUp	Đưa chuột ra khỏi nút
MouseMove	Di chuyển chuột trên nút
MouseLeave	Dời chuyển ra khỏi nút



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

GROUPBOX - NHÓM

Chứa các điều khiển khác, tạo thành các vùng làm việc độc lập trên form

Thuộc tính	Mô tả
Name	Tên nhóm, bắt đầu bằng grb
BackColor	Thiết lập mầu nền cho nhóm, nếu BackColor = Transparent thì nhóm sẽ có mầu nền giống với mầu nền của Form.
TabIndex	Thứ tự truy cập của nhóm khi người dùng bấm phím Tab.
Text	Thiết lập tiêu đề của nhóm.
Visible	Visible = True: hiển thị nhóm, Visible = False: ẩn nhóm.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CHECKBOX – HỘP ĐÁNH DẦU

Cho phép không chọn/chọn một/chọn nhiều khả năng



Thuộc tính	Mô tả
Name	Tên, bắt đầu bằng chk
Checked	True/false: checkbox đã được chọn/không được chọn
CheckState	Checked: được chọn
	Unchecked: không được chọn
	Indeterminate: chưa xác định



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CHECKBOX – HỘP ĐÁNH DẦU

Sự kiện

Sư kiện	Mô tả
Click	Nhấn chuột vào checkbox
GotFocus	Chuyển tiêu điểm vào checkbox
LostFocus	Mất tiêu điểm
CheckedChanged	Checkbox thay đổi trạng thái



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

RADIOBUTTON - NÚT TUÝ CHỌN

- Cho phép người dùng chọn một trong nhiều lựa chọn
- Thuộc tính

Thuộc tính	Mô tả
Name	Tên, bắt đầu bằng rdo
Checked	True/false: radiobutton đã được chọn/ không được chọn
Enable	True/False: bật/tắt nút tuỳ chọn
Visible	True/False: Hiện/Ẩn nút



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

RADIOBUTTON - NÚT TUÝ CHỌN

Sự kiện

Sư kiện	Mô tả
Click	Nhấn chuột vào nút tuỳ chọn
GotFocus	Chuyển tiêu điểm vào nút tuỳ chọn
LostFocus	Mất tiêu điểm
CheckedChanged	Nút tuỳ chọn thay đổi trạng thái

- Chú ý
 - RadioButton kết hợp với GroupBox



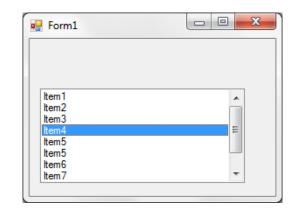




VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

LISTBOX – HỘP DANH SÁCH

- Cung cấp danh sách cho phép lựa chọn
- Hiển thị thanh cuộn nếu vượt quá vùng thể hiện
- Sự kiện:



Sự kiện	M ô tả
Click	Nháy chuột vào danh sách
DoubleClick	Nháy đúp chuột
GotFocus	Chuyển tiêu điểm tới
LostFocus	Mất tiêu điểm
SelectedIndexChanged	Thay đổi trạng thái lựa chọn các dòng dữ liệu trong listbox



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

LISTBOX – HỘP DANH SÁCH

Thuộc tính	Mô tả
Name	Tên hộp ListBox, bắt đầu bằng Ist
DataSource	Nguồn dữ liệu cho ListBox
MultiColumn	True/False: Nhiều cột
ColumnWidth	Độ rộng cho mỗi cột trong ListBox.
Items	Danh sách khởi tạo các phần tử
SelectedIndex	Trả về số thứ tự của phần tử đang được chọn trong danh sách, phần tử đầu tiên có SelectedIndex = 0, nếu không có phần tử nào được chọn thì SelectedIndex = -
SelectionMode	Chế độ lựa chọn các phần tử trong hộp danh sách khi thực thi chương trình. None,One,MultiSimple (lựa chọn nhiều phần tử riêng biệt), MultiExtended (chọn một khối các phần từ liền nhau)
SelectedItems	Trả về tập các phần tử đang được chọn.
Sorted	True/False:sắp xếp hay không

CC KINH TE QUOC D AN

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

LISTBOX – HỘP DANH SÁCH

- Phương thức
 - ListName.Items.Add(Item); //thêm phần tử
 - ListName.Items.Remove(Item); //xoá phần tử
 - ListName.Items.RemoveAt(Index); //xoá tại chỉ số index
 - ListName.Items.Clear(); //xoá danh sách Ví du:

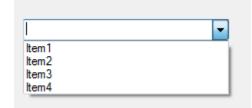
```
lstQue.Items.Add("Ha Noi");
```

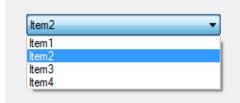


VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

COMBOBOX – HỘP LỰA CHỌN

Cho phép lựa chọn một mục dữ liệu trong hộp danh sách thả xuống





Thuộc tính	Mô tả
Name	Tên hộp ComboBox, bắt đầu bằng cbo
DataSource	Nguồn dữ liệu cho ComboBox
DropDownStyle	Các kiểu. DropDown: cho phép chọn và nhập dữ liệu, Simple: Hiển thị toàn bộ danh sách các mục, cho phép nhập. DropDownList: Chỉ cho phép lựa chọn các phần tử trong hộp thả xuống
Items	Khởi tạo các giá trị khi thiết kế
SelectedIndex	Số thứ tự các phần tử đang được chọn



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

COMBOBOX – HỘP LỰA CHỌN

Sự kiện

Sự kiện	Mô tả
Click	Nháy chuột vào danh sách
DoubleClick	Nháy đúp chuột
GotFocus	Chuyển tiêu điểm tới
LostFocus	Mất tiêu điểm
SelectedIndexChanged	Thay đổi trạng thái lựa chọn các dòng dữ liệu trong ComboBox
TextChanged	Có sự thay đổi văn bản của ComboBox
DropDown	Chỉ xảy ra đối với hộp Combo DropDown và DropDownList, sự kiện này được gọi ngay sau khi người dùng nhấp mũi tên để thả hộp danh sách xuống (phím tắt Alt+↓), sự kiện này chủ yếu được sử dụng để nhập dữ liệu cho các phần tử của hộp Combo.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

COMBOBOX – HỘP LỰA CHỌN

- Phương thức
 - ComboName.Items.Add(Item); //Thêm
 - ComboName.Items.Remove(Item); //xoá
 - ComboName.Items.RemoveAt(Index); //xoá tại vị trí index
 - ComboName.Items.Clear(); //xoá tất cả

```
Ví dụ: nhập dữ liệu cho hộp cboQue
```

```
private void cboQue_DropDown(object sender, EventArgs e)
{
    cboQue.Items.Clear();
    cboQue.Items.Add("Hà Nội");
    cboQue.Items.Add("Nam Định");
    cboQue.Items.Add("Đà Lạt");
}
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CHECKEDLISTBOX

- Gần giống như ListBox, có thêm checkbox ở đầu dòng
- Đặt tên bắt đầu bằng clb
- Có thuộc tính, sự kiện tương tự như ListBox
- Bổ sung thêm
 - Thuộc tính: CheckedItems: tập các phần tử được check
 - Sự kiện: ItemCheck: người dùng nháy đúp chuột



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

NUMERICUPDOWN

Lựa chọn một giá trị số trong một khoảng giá trị với bước nhảy xác định

Thuộc tính	Mô tả
Name	Tên, bắt đầu bởi tiếp đầu ngữ nud
Increment	Bước nhảy
Maximum	Cận trên
Minimum	Cận dưới
Value	Giá trị hiện tại

Sự kiện	Mô tả
ValueChanged	Thay đổi giá trị của điều khiển



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

THANH CUỘN HSCROLLBAR VÀ VSCROLLBAR

Thanh cuộn ngang: HScrollBar và thanh cuộn dọc VScrollBar

Thuộc tính	Mô tả
Name	Tên thanh cuộn, bắt đầu bởi hsb và vsb.
Minimum	Giá trị nhỏ nhất
Maximum	Giá trị lớn nhất
Value	Giá trị
LargeChange	Mức độ thay đổi giá trị khi cuộn chuột
SmallChange	Mức độ thay đổi khi nhấn nút mũi tên trên thanh cuộn (mặc định I)

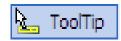
Sự kiện	Mô tả
ValueChanged	Thay đổi giá trị của điều khiển
Scroll	Cuộn thanh



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TOOLTIP - DÒNG CHÚ THÍCH

- Tooltip
 - Điều khiển Tooltip



hiển thị thông tin chú thịch khi đưa chuột qua





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TIMER – BỘ ĐẾM THỜI GIAN

Cho thực thi hành động sau khoảng thời gian

Thuộc tính	Mô tả
Name	Tên điều khiển Timer, bắt đầu bởi tiếp đầu ngữ tmr
Interval	= n là chu kỳ thực hiện sự kiện Tick của điều khiển Timer.
	n là số nguyên, được tính bằng mili giây và có giá trị >0
Enabled	True/False: cho phép/không cho phép hoạt động

Sự kiện	Mô tả
Tick	Kích hoạt sau mỗi chu kỳ Interval

Phương thức

- Start: kích hoạt (tương tự Enable = True)
- Stop: dwng (twong tw Enable = False)

C KINH TE QUOC OAN

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

RICHTEXTBOX

- Cho phép tạo/hiển thị các tập văn bản Rich Text (*.rtf)
 - Name: bắt đầu bằng rtb
- Phương thức:
 - LoadFile: nap nôi dung
 - SaveFile: Iuu file



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

DATETIMEPICKER

Cho phép chọn thời gian dưới dạng lịch

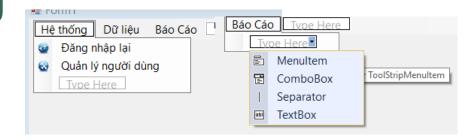
Thuộc tính	Mô tả
Name	Tên điều khiển DateTimePicker, bắt đầu bởi dtp
Format	Định dạng kiểu hiển thị của thời gian, Ngày tháng thường chọn giá trị Short
Values	Gá trị hiện thời





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

3.3. MENU VÀ THANH CÔNG CỤ



Tạo menu đổ xuống (Drop down menu)

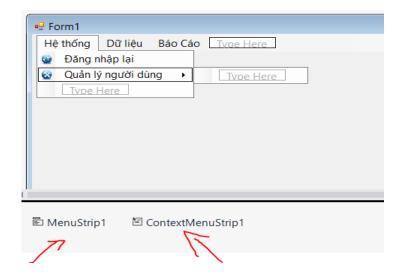
- B1: Thêm một form mới vào dự án
- B2: Chọn MenuStrip trong hộp Toolbox và định vị vào phần trên cùng của form
- B3: Soạn thảo nội dung menu bằng cách: tại mỗi ô "Type Here" trên menustrip, trong bảng combo box tương ứng ta chọn hình thức của mục menu là: menu item; combobox; textbox và separator.
- B4: Gán biểu tượng (icon) cho mỗi mục menu bằng cách nháy phải chuột vào mục tương ứng và chọn "Set image..." và thực hiện lựa chọn icon trong thư viện (resource) đã được import
- B5:Lập trình cho mục menu cấp thấp nhất bằng cách nháy kép chuột vào mục tương ứng để hiện ra cửa sổ coding



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

3.3.1 TẠO MENU ĐỔ XUỐNG

- Thuộc tính của menu
- Với mỗi mục menu thì đều có các thuộc tính thường dung sau
 - Name: trùng với nội dung của mục menu (caption)
 - Check: (true/false)
 - Enable: (true/false)
 - Visible: (true/false)
- Ta bấm truy cập menu đổ xuống hay menu ngữ cảnh bằng cách bấm vào tên của nó ở lề bên dưới màn hình.



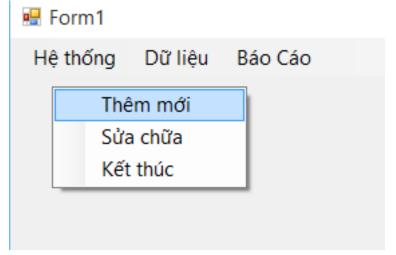


VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

3.3.2 TẠO MENU NGỮ CẢNH (POPUP MENU)

- B1: ContextMenuStrip trong hộp Toolbox và định vị vào Form
- B2: Soạn thảo nội dung menu giống như với Dropdown menu
- B3: Trong cửa sổ thuộc tính của Form hiện thời, tại mục ContextmenuStrip, ta chọn tên của popupmenu vừa thiết kế
- B4: Lập trình cho từng mục của menu bằng cách nháy kép chuột vào mục menu tương ứng.

 Khi chạy chương trình, mỗi khi ta nháy phải chuột tại form tương ứng thì sẽ làm hiện ra menu ngữ cảnh.

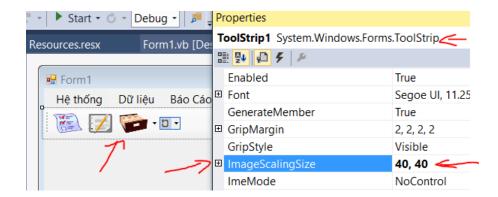


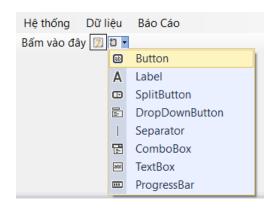


VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

3.3.3 TẠO THANH CÔNG CỤ

- B1: Chọn ToolStrip trong hộp Toolbox và định vị lên form
- B2: Với mỗi ô trên thanh công cụ ta có thể lựa chọn nó có thể là 1 trong 8 khả năng như hình bên
- B4: với mỗi loại đỗi tượng ta sẽ có những thuộc tính và phương thức riêng. Gán icon cho mỗi nút giống như làm với menu. Đặt kích thước cho icon bằng cách thay đổi giá trị của thuộc tính ImageScalingSize của thanh công cụ (nên đặt khoảng 40,40 là vừa)
- B5: Lập trình cho mỗi thành phần trên thanh công cụ bằng cách nháy kép chuột tại vùng muốn lập trình



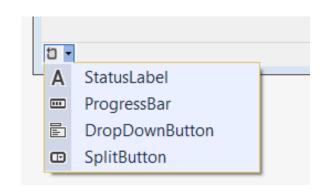




VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

3.3.4 TẠO THANH TRẠNG THÁI (STATUS BAR)

- B1: Chọn Status Strip trong hộp Toolbox và them vào lề dưới cùng của form
- B2: Khai báo mỗi ô trên thanh trạng thái bằng cách chọn 1 trong nhiều lựa chọn (hình bên), thường ta hay chọn là StatusLabel
- B3: Ta gán nội dung cho mỗi label trên thanh trạng thái khi chương trình thi hành bằng cách dung câu lệnh
- <tên ô text trên stt bar>.text= "Nội dung text"
- Với i=0..n



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Chương 4

Lập trình cơ sở dữ liệu



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỤC ĐÍCH

- Ôn tập lại hệ thống kiến thức về ngôn ngữ truy vấn SQL sử dụng để lập trình kết nối, tương tác với cơ sở dữ liệu của ứng dụng.
- Trang bị kiến thức và hướng dẫn sinh viên thực hành các bước để kết nối, tương tác với cơ sở dữ liệu của ứng dụng.

CC KINH TË QUOC DAN

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.1. ÔN TẬP NGÔN NGỮ TRUY VẤN CÓ CẦU TRÚC SQL

- Một số khái niệm liên quan đến CSDL
 - Fields
 - Record
 - Table
 - Key
 - Database
- Ngôn ngữ SQL



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.2 Khái quát về SQL sử dụng trong phát triển ứng dụng

- Giới thiệu chung
 - SQL, viết tắt của Structured Query Language (ngôn ngữ hỏi có cấu trúc), do Viện tiêu chuẩn Quốc gia Mỹ (ANSI) tạo ra. Là công cụ sử dụng để tổ chức, quản lý và truy xuất dữ liệu được lưu trữ trong các cơ sở dữ liệu. SQL là một hệ thống ngôn ngữ bao gồm tập các câu lệnh sử dụng để tương tác với cơ sở dữ liệu quan hệ.
- Các chức năng chính của SQL



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.2 Khái quát về SQL sử dụng trong phát triển ứng dụng

Vai trò của SQL trong các lĩnh vực tin học

- SQL là ngôn ngữ hỏi có tính tương tác: Người sử dụng có thể dễ dàng thông qua các trình tiện ích để gởi các yêu cầu dưới dạng các câu lệnh SQL đến cơ sở dữ liệu và nhận kết quả trả về từ cơ sở dữ liệu
- SQL là ngôn ngữ lập trình cơ sở dữ liệu: Các lập trình viên có thể nhúng các câu lệnh SQL vào trong các ngôn ngữ lập trình để xây dựng nên các chương trình ứng dụng giao tiếp với cơ sở dữ liệu
- SQL là ngôn ngữ quản trị cơ sở dữ liệu: Thông qua SQL, người quản trị cơ sở dữ liệu có thể quản lý được cơ sở dữ liệu, định nghĩa các cấu trúc lưu trữ dữ liệu, điều khiển truy cập cơ sở dữ liệu,...
- SQL là ngôn ngữ cho các hệ thống khách/chủ (client/server): Trong các hệ thống cơ sở dữ liệu khách/chủ, SQL được sử dụng như là công cụ để giao tiếp giữa các trình ứng dụng phía máy khách với máy chủ cơ sở dữ liệu.
- SQL là ngôn ngữ truy cập dữ liệu trên Internet: Cho đến nay, hầu hết các máy chủ Web cũng như các máy chủ trên Internet sử dụng SQL với vai trò là ngôn ngữ để tương tác với dữ liệu trong các cơ sở dữ liệu.
- SQL là ngôn ngữ cơ sở dữ liệu phân tán: Đối với các hệ quản trị cơ sở dữ liệu phân tán, mỗi một hệ thống sử dụng SQL để giao tiếp với các hệ thống khác trên mạng, gởi và nhận các yêu cầu truy xuất dữ liệu với nhau.
- SQL là ngôn ngữ sử dụng cho các cổng giao tiếp cơ sở dữ liệu: Trong một hệ thống mạng máy tính với nhiều hệ quản trị cơ sở dữ liệu khác nhau, SQL thường được sử dụng như là một chuẩn ngôn ngữ để giao tiếp giữa các hệ quản trị cơ sở dữ liệu.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.2 Khái quát về SQL sử dụng trong phát triển ứng dụng

- Các chức năng chính của SQL trong phát triển ứng dụng
 - Định nghĩa dữ liệu: SQL cung cấp khả năng định nghĩa các cơ sở dữ liệu, các cấu trúc lưu trữ và tổ chức dữ liệu cũng như mối quan hệ giữa các thành phần dữ liêu.
 - Truy xuất và thao tác dữ liệu: Với SQL, người dùng có thể dễ dàng thực hiện các thao tác truy xuất, bổ sung, cập nhật và loại bỏ dữ liệu trong các cơ sở dữ liệu.
 - Điều khiển truy cập: SQL có thể được sử dụng để cấp phát và kiểm soát các thao tác của người sử dụng trên dữ liệu, đảm bảo sự an toàn cho cơ sở dữ liệu
 - Đảm bảo toàn vẹn dữ liệu: SQL định nghĩa các ràng buộc toàn vẹn trong cơ sở dữ liệu nhờ đó đảm bảo tính hợp lệ và chính xác của dữ liệu trước các thao tác cập nhật cũng như các lỗi của hệ thống.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thưởng dùng trong lập trình CSDL

Câu lệnh Select

SELECT [ALL | DISTINCT][TOP n] danh_sách_chon

[INTO tên_bảng_mới]

FROM danh_sách_bảng/khung_nhìn

[WHERE điều_kiện]

[GROUP BY danh_sách_cột]

[HAVING điều_kiện]

[ORDER BY cột_sắp_xếp]



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thưởng dùng trong lập trình CSDL

Sử dụng cấu trúc CASE trong danh sách chọn

```
CASE biểu thức
WHEN biểu thức kiểm tra THEN kết quả
[ ... ]
[ELSE kết quả của else]
END
hoặc:
CASE WHEN điều kiện THEN kết quả
[ ... ]
[ELSE kết_quả_của else]
END
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

Sử dụng cấu trúc CASE trong danh sách chọn

Ví dụ : Để hiển thị mã, họ tên và giới tính (nam hoặc nữ) của các sinh viên, ta sử dụng câu lệnh

SELECT masv, hodem, ten,

CASE gioitinh

WHEN 1 THEN 'Nam'

ELSE 'Nữ'

END AS gioitinh

FROM sinhvien



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

Sử dụng cấu trúc CASE trong danh sách chọn

Ví dụ : Để hiển thị mã, họ tên và giới tính (nam hoặc nữ) của các sinh viên, ta sử dụng câu lệnh

SELECT masv, hodem, ten,

CASE

WHEN gioitinh=1 THEN 'Nam'

ELSE 'Nữ'

END AS gioitinh

FROM sinhvien

O DAN

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

Một số kỹ thuật khác trong câu lện Select

- Trong danh sách chọn của câu lệnh SELECT còn có thể sử dụng các biểu thức. Mỗi một biểu thức trong danh sách chọn trở thành một cột trong kết quả truy vấn.

Ví dụ : câu lệnh dưới đây cho biết tên và số tiết của các môn học

SELECT tenmonhoc, sodvht*15 AS sotiet

FROM monhoc

- Trong kết quả của truy vấn có thể xuất hiện các dòng dữ liệu trùng nhau. Để loại bỏ bớt các dòng này, ta chỉ định thêm từ khóa DISTINCT ngay sau từ khoá SELECT
- Cần hạn chế số lượng các dòng xuất hiện trong kết quả truy vấn, ta chỉ định thêm mệnh đề TOP <n> ngay trước danh sách chọn của câu lệnh SELECT.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

Một số kỹ thuật khác trong câu lện Select

- Để kiểm tra xem giá trị dữ liệu nằm trong (ngoài) một khoảng nào đó, ta sử dụng toán tử BETWEEN (NOT BETWEEN)
- Từ khoá IN được sử dụng khi ta cần chỉ định điều kiện tìm kiếm dữ liệu cho câu lệnh SELECT là một danh sách các giá trị. Sau IN (hoặc NOT IN) có thể là một danh sách các giá trị hoặc là một câu lệnh SELECT khác.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

Một số kỹ thuật khác trong câu lện Select

- Từ khoá LIKE (NOT LIKE) sử dụng trong câu lệnh SELECT nhằm mô tả khuôn dạng của dữ liệu cần tìm kiếm. Chúng thường được kết hợp với các ký tự đại diện sau đây:

Ký tự đại diện	ý nghĩa
%	Chuỗi ký tự bất kỳ gồm không hoặc nhiều ký tự
_	Ký tự đơn bất kỳ
	Ký tự đơn bất kỳ trong giới hạn được chi định (ví dụ [a-f]) hay một tập (ví dự [abcdef])
[^]	Ký tự đơn bất kỳ không nằm trong giới hạn được chi định (ví dụ [^a-f] hay một tập (ví dụ [^abcdef]).

C KINH TE QUOC OR

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

Một số kỹ thuật khác trong câu lện Select

Giá trị NULL

Dữ liệu trong một cột cho phép NULL sẽ nhận giá trị NULL trong các trường hợp sau:

- Nếu không có dữ liệu được nhập cho cột và không có mặc định cho cột hay kiểu dữ liệu trên cột đó.
 - Người sử dụng trực tiếp đưa giá trị NULL vào cho cột đó.
 - Một cột có kiểu dữ liệu là kiểu số sẽ chứa giá trị NULL nếu giá trị được chỉ định gây tràn số.

Trong mệnh đề WHERE, để kiểm tra giá trị của một cột có giá trị NULL hay không, ta sử dụng cách viết:

WHERE tên_cột IS NULL

hoặc:

WHERE tên_cột IS NOT NULL

CCKINH TE QUOCO

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

- Một số kỹ thuật khác trong câu lện Select
 - Câu lệnh SELECT ... INTO có tác dụng tạo một bảng mới có cấu trúc và dữ liệu được xác định từ kết quả của truy vấn. Bảng mới được tạo ra sẽ có số cột bằng số cột được chỉ định trong danh sách chọn và số dòng sẽ là số dòng kết quả của truy vấn
 - Ví dụ: Câu lệnh dưới đây truy vấn dữ liệu từ bảng SINHVIEN và tạo một bảng TUOISV bao gồm các trường HODEM, TEN và TUOI

SELECT hodem, ten, YEAR (GETDATE())-YEAR (ngaysinh) AS tuoi

INTO tuoisv

FROM sinhvien

CCKINH TË QUOCO P.N.

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

- Thống kê dữ liệu với GROUP BY
- Mệnh đề GROUP BY sử dụng trong câu lệnh SELECT nhằm phân hoạch các dòng dữ liệu trong bảng thành các nhóm dữ liệu, và trên mỗi nhóm dữ liệu thực hiện tính toán các giá trị thống kê như tính tổng, tính giá trị trung bình,...
- Các hàm gộp được sử dụng để tính giá trị thống kê cho toàn bảng hoặc trên mỗi nhóm dữ liệu. Chúng có thể được sử dụng như là các cột trong danh sách chọn của câu lệnh SELECT hoặc xuất hiện trong mệnh đề HAVING, nhưng không được phép xuất hiện trong mệnh đề WHERE

SELECT lop.malop,tenlop,COUNT(masv) AS siso

FROM lop, sinhvien

WHERE lop.malop=sinhvien.malop

GROUP BY lop.malop,tenlop



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

■ Thống kê dữ liệu với GROUP BY

Hàm gộp	Chức năng
SUM([ALL DISTINCT] biểu_thức)	Tính tổng các giá trị.
AVG([ALL DISTINCT] biểu_thức)	Tính trung bình của các giá trị
COUNT([ALL DISTINCT]	Đếm số các giá trị trong biểu thức.
biểu_thức)	
COUNT(*)	Đếm số các dòng được chọn.
MAX(biểu_thức)	Tính giá trị lớn nhất
MIN(biểu_thức)	Tính giá trị nhỏ nhất

Trong đó:

- Hàm SUM và AVG chỉ làm việc với các biểu thức số.
- Hàm SUM, AVG, COUNT, MIN và MAX bỏ qua các giá trị NULL khi tính toán.
- Hàm COUNT(*) không bỏ qua các giá trị NULL.

CCKINH TË QUOCO DE

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

- Chỉ định điều kiện đối với hàm gộp
 - Mệnh đề HAVING được sử dụng nhằm chỉ định điều kiện đối với các giá trị thống kê được sản sinh từ các hàm gộp tương tự như cách thức mệnh đề WHERE thiết lập các điều kiện cho câu lệnh SELECT.
 - Mệnh đề HAVING thường không thực sự có nghĩa nếu như không sử dụng kết hợp với mệnh đề GROUP BY.
 - Một điểm khác biệt giữa HAVING và WHERE là trong điều kiện của WHERE không được có các hàm gộp trong khi HAVING lại cho phép sử dụng các hàm gộp trong điều kiện của mình.

SELECT sinhvien.masv,hodem,ten,

SUM(diemlan1*sodvht)/sum(sodvht)

FROM sinhvien, diemthi, monhoc

WHERE sinhvien.masv=diemthi.masv AND

diemthi.mamonhoc=monhoc.mamonhoc

GROUP BY sinhvien.masv,hodem,ten

HAVING sum(diemlan1*sodvht)/sum(sodvht)>=5

CCKINH TË QUOCO PE

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

- Truy vấn con (Subquery)
 - Truy vấn con là một câu lệnh SELECT được lồng vào bên trong một câu lệnh SELECT, INSERT, UPDATE, DELETE hoặc bên trong một truy vấn con khác. Loại truy vấn này được sử dụng để biểu diễn cho những truy vấn trong đó điều kiện truy vấn dữ liệu cần phải sử dụng đến kết quả của một truy vấn khác.
- Khi sử dụng truy vấn con cần lưu ý một số quy tắc sau:
- Một truy vấn con phải được viết trong cặp dấu ngoặc. Trong hầu hết các trường hợp, một truy vấn con thường phải có kết quả là một cột (tức là chỉ có duy nhất một cột trong danh sách chọn).
 - Mệnh đề COMPUTE và ORDER BY không được phép sử dụng trong truy vấn con.
 - Các tên cột xuất hiện trong truy vấn con có thể là các cột của các bảng trong truy vấn ngoài.
- Một truy vấn con thường được sử dụng làm điều kiện trong mệnh đề WHERE hoặc HAVING của một truy vấn khác.
- Nếu truy vấn con trả về đúng một giá trị, nó có thể sử dụng như là một thành phần bên trong một biểu thức (chẳng hạn xuất hiện trong một phép so sánh bằng)

CC KINH TË QUOC D AN

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

- Truy vấn con (Subquery)
- Nếu truy vấn con trả về nhiều hơn một giá trị, việc sử dụng phép so sánh như trên sẽ không hợp lệ. Trong trường hợp này, sau phép toán so sánh phải sử dụng thêm lượng từ ALL hoặc ANY. Lượng từ ALL được sử dụng khi cần so sánh giá trị của biểu thức với tất cả các giá trị trả về trong kết quả của truy vấn con; ngược lại, phép so sánh với lượng từ ANY có kết quả đúng khi chỉ cần một giá trị bất kỳ nào đó trong kết quả của truy vấn con thoả mãn điều kiện.
- Ví dụ: Câu lệnh dưới đây cho biết họ tên của những sinh viên lớp Tin K25 sinh trước tất cả các sinh viên của lớp Toán K25

SELECT hodem,ten

FROM sinhvien JOIN lop ON sinhvien.malop=lop.malop

WHERE tenlop='Tin K25' AND

ngaysinh<ALL(SELECT ngaysinh

FROM sinhvien JOIN lop

ON sinhvien.malop=lop.malop

WHERE lop.tenlop='Toán K25')



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

Bổ sung từng dòng dữ liệu với lệnh INSERT

 Để bổ sung một dòng dữ liệu mới vào bảng, ta sử dụng câu lệnh INSERT với cú pháp như sau:

INSERT INTO tên_bảng[(danh_sách_cột)]

VALUES(danh_sách_tri)

- Trong câu lệnh INSERT, danh sách cột ngay sau tên bảng không cần thiết phải chỉ định nếu giá trị các trường của bản ghi mới được chỉ định đầy đủ trong danh sách trị. Trong trường hợp này, thứ tự các giá trị trong danh sách trị phải bằng với số lượng các trường của bảng cần bổ sung dữ liệu cũng như phải tuân theo đúng thứ tự của các trường như khi bảng được định nghĩa.
- Trong trường hợp chỉ nhập giá trị cho một số cột trong bảng, ta phải chỉ định danh sách các cột cần nhập dữ liệu ngay sau tên bảng. Khi đó, các cột không được nhập dữ liệu sẽ nhận giá trị mặc định (nếu có) hoặc nhận giá trị NULL (nếu cột cho phép chấp nhận giá trị NULL). Nếu một cột không có giá trị mặc định và không chấp nhận giá trị NULL mà không được nhập dữ liệu, câu lệnh sẽ bị lỗi.

CKINH TE QUOCO DE

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

Bổ sung nhiều dòng dữ liệu từ bảng khác

- Một cách sử dụng khác của câu lệnh INSERT được sử dụng để bổ sung nhiều dòng dữ liệu vào một bảng, các dòng dữ liệu này được lấy từ một bảng khác thông qua câu lệnh SELECT. Ở cách này, các giá trị dữ liệu được bổ sung vào bảng không được chỉ định tường minh mà thay vào đó là một câu lệnh SELECT truy vấn dữ liệu từ bảng khác.
- Cú pháp câu lệnh INSERT có dạng như sau:

INSERT INTO tên_bảng[(danh_sách_cột)] câu_lệnh_SELECT

Ví dụ: Giả sử ta có bảng LUUSINHVIEN bao gồm các trường HODEM, TEN, NGAYSINH. Câu lệnh dưới đây bổ sung vào bảng LUUSINHVIEN các dòng dữ liệu có được từ câu truy vấn SELECT:

INSERT INTO luusinhvien

SELECT hodem,ten,ngaysinh

FROM sinhvien

WHERE noisinh like '%Huế%'



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

<u>Câu lệnh UPDATE</u> trong SQL được sử dụng để cập nhật dữ liệu trong các bảng. Câu lệnh này có cú pháp như sau:

```
UPDATE tên_bảng

SET tên_cột = biểu_thức

[, ..., tên_cột_k = biểu_thức_k]

[FROM danh_sách_bảng]
```

[WHERE điều_kiện]

Sau UPDATE là tên của bảng cần cập nhật dữ liệu. Một câu lệnh UPDATE có thể cập nhật dữ liệu cho nhiều cột bằng cách chỉ định các danh sách tên cột và biểu thức tương ứng sau từ khoá SET. Mệnh đề WHERE trong câu lệnh UPDATE thường được sử dụng để chỉ định các dòng dữ liệu chịu tác động của câu lệnh (nếu không chỉ định, phạm vi tác động của câu lệnh được hiểu là toàn bộ các dòng trong bảng)

C KINH TË QUOC O P

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

Xoá dữ liệu

 Để xoá dữ liệu trong một bảng, ta sử dụng câu lệnh DELETE. Cú pháp của câu lệnh này như sau:

DELETE FROM tên_bảng

[FROM danh_sách_bảng]

[WHERE điều_kiện]

- Trong câu lệnh này, tên của bảng cần xoá dữ liệu được chỉ định sau DELETE FROM. Mệnh đề WHERE trong câu lệnh được sử dụng để chỉ định điều kiện đối với các dòng dữ liệu cần xoá. Nếu câu lệnh DELETE không có mệnh đề WHERE thì toàn bộ các dòng dữ liệu trong bảng đều bị xoá.
- Ví dụ : Câu lệnh dưới đây xoá khỏi bảng SINHVIEN những sinh viên sinh tại Huế

DELETE FROM sinhvien

WHERE noisinh LIKE '%Huế%'

CKINH TE QUOCO DE

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

Xoá dữ liệu

- Xoá toàn bộ dữ liệu trong bảng
- Câu lệnh DELETE không chỉ định điều kiện đối với các dòng dữ liệu cần xoá trong mệnh đề WHERE sẽ xoá toàn bộ dữ liệu trong bảng. Thay vì sử dụng câu lệnh DELETE trong trường hợp này, ta có thể sử dụng câu lệnh TRUNCATE có cú pháp như sau:
- TRUNCATE TABLE tên_bảng
- Ví dụ : Câu lệnh sau xoá toàn bộ dữ liệu trong bảng diemthi:

DELETE FROM diemthi

có tác dụng tương tự với câu lệnh

TRUNCATE TABLE diemthi



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.3 Các câu lệnh SQL thường dùng trong lập trình CSDL

Thống kê trên toàn bộ dữ liệu

- Khi cần tính toán giá trị thống kê trên toàn bộ dữ liệu, ta sử dụng các hàm gộp trong danh sách chọn của câu lệnh SELECT. Trong trường hợp này, trong danh sách chọn không được sử dụng bất kỳ một tên cột hay biểu thức nào ngoài các hàm gộp.
- Ví dụ : Để thống kê trung bình điểm lần 1 của tất cả các môn học, ta sử dụng câu lệnh như sau:

SELECT AVG(diemlan1)

FROM diemthi

CKINH TË QUOC

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.4 BÀI TẬP TỔNG HỢP

- Hãy thiết kế mô hình CSDL cho 1 trung tâm thương mại với các thông tin ràng buộc như sau:
- Trung tâm kinh doanh nhiều loại mặt hàng, mỗi mặt hàng đều được quản lý với các thông tin: tên hàng, nước sản xuất, đơn giá, tỷ lệ thuế.
 Đồng thời trung tâm củng có danh sách các khách hàng của mình, mỗi khác hàng được quản lý thông tin gồm tên khách hàng, địa chỉ, mã số thuế. Mỗi hoá đơn bán hàng tại trung tâm thì sẽ ghi nhân thông tin về 1 mặt hàng được bán.
- Viết lệnh SQL để trả lời những câu hỏi sau:
- 1, Liệt kệ những mặt hàng ngoại nhập hiện được bán tai trung tâm'
- 2, Liệt kê danh sách những mặt hàng do nhật bản sản xuất và có đơn giá cao nhất
- 3, Liệt kê danh sách của 10 mặt hàng có đơn giá thấp nhất
- 4, Tính tổng doanh thu bán hàng của trung tâm trong quý 2 năm 2006
- 5, Tính tống doanh thu của việc kinh doanh mặt hàng tivi tại trung tâm năm 2006
- 6, Liệt kê danh sách những mặt hàng có doanh thu cao nhất trong năm 2006
- 7, Đếm tổng số lượng mặt hàng là tủ lạnh đã được bán trong quý I năm 2007
- 8, Liệt kê danh sách những mặt hàng do VN sản xuất mà không bán được cái nào trong tháng 9 năm 2006
- 9, Tạo bảng tổng hợp doanh thu bán hàng theo từng mặt hàng trong năm 2006
- 10, Trung tâm muốn tặng quà cho 3 khách hàng tiêu nhiều tiền nhất trong việc mua hàng tại trung tâm trong nữa đầu năm 2007. Hãy tìm tên của 3 người đó



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.5 TỔNG QUAN VỀ ADO.NET

- ADO.NET là một bộ các thư viện hướng đối tượng (OOP) cho phép bạn tương tác với nhiều nguồn dữ liệu khác nhau: database; text file; XML; Excel,...
- Mô hình truy xuất csdl trong ADO.NET có thể nói như sau: trước hết là thiết lập kết nối đến csdl. Tiếp theo đối tượng điều phối (data adapter) được tạo ra để truy vấn dl từ các bảng. Sau đó tạo các đối tượng DataSet chứa bản sao của các bảng dl bạn muốn làm việc. Trong một Dataset thì có thể có 1 hoặc nhiều DataTable. Mỗi DataTable là 1 bảng chứa dữ liệu mà ta có thể hiển thị và thực hiện việc them, sửa, xóa.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỘT SỐ KHÁI NIỆM LIÊN QUAN

- Data Provider: Là một thư viện chứa các cách thức cho phép kết nỗi với mỗi loại cơ sở dữ liệu khác nhau.
 - OleDb Data Provider: Để kết nối với CSDL Access hoặc Excel
 - SQL Data Provider: Để kết nối với SQL Server
- SqlConnection(OleDbConnection): đối tượng để thiết lập kết nối đến CSDL SQLserver
- SQLcommand(OleDbCommand): Để gửi một câu lệnh SQL đến CSDL. Mỗi đối tượng command dùng một đối tượng Connection để xác định CSDL nào sẽ dc truy xuất.
- SQIDataReader(OleDbDataReader): Cho phép lấy kết quả của 1 câu lện SELECT từ một đối tượng command



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

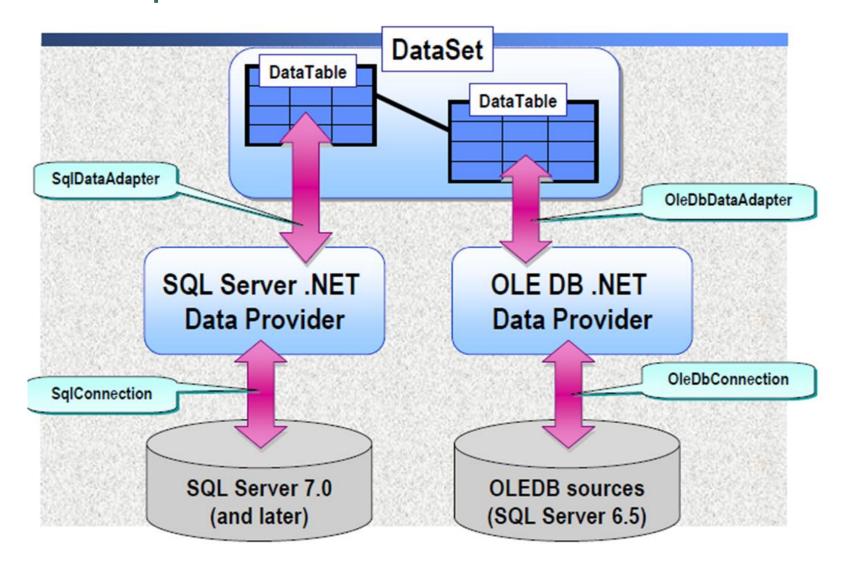
MỘT SỐ KHÁI NIỆM LIÊN QUAN

- DataSet: Là tập hợp bản sao của các bảng dữ liệu được lấy từ CSDL. Chứa nhiều DataTable. Nó có thể sử dụng với mọi DataProvider.
- DataTable: Là một bảng chứa dữ liệu được lấy ra từ DataSet
- SQLDataAdapter(OleDbDataAdapter): Là bộ điều phối dữ liệu, để đổ dữ liệu vào DataSet và thực hiện việc cập nhật những thay đổi dữ liệu đã được thực hiện ngược trở lại vào CSDL thông qua thủ tục Update.
- DataBindings: tương tự như datatable



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MÔ HÌNH ĐỐI TƯỢNG ADO.NET





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.6. KÉT NÓI CSDL BẰNG ADO.NET

- Connection
- Command
- DataReader
- DataAdaper
- Dataset

C KINH TÉ QUOC O A N

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.6.1. CONNECTION

- Thiết lập và quản lý kết nối với CSDL
- Có 2 loại:
 - SqlConnection
 - OleDbConnection
- Thuộc tính quan trọng:
 - ConnectionString
- Phương thức quan trọng:
 - Open()
 - Close()
- Chú ý:
 - Luôn đóng Connection sau khi sử dụng



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.6.2.CONNECTION (TIÉP)

Ví dụ:

```
SqlConnection con; //Đối tượng để kết nối
con = new SqlConnection(); //Khởi tạo
con.ConnectionString = @"Data
   Source=113.190.234.205,1433;Initial
   Catalog=ktqd2019;Persist Security
   Info=True;User ID=ktqd;Password=Ktqd-2019";
con.Open(); //Mở kết nối
```



Tạo ConnectString chính xác và nhanh nhất?

C KINH TE QUOC

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.6.3 COMMAND

- Thực hiện các thao tác với CSDL
 - DDL, DML, gọi thủ tục,
- Có 2 loại:
 - SqlCommand
 - OleDbCommand
- Thuộc tính quan trọng:
 - Connection
 - CommandText
- Phương thức quan trọng:
 - ExecuteNonQuery()

(thực hiện lệnh: INSERT, UPDATE, DELETE)

ExecuteScalar()

(thực hiện lệnh SELECT trả về 1 giá trị)

ExecuteReader()

(thực hiện lệnh SELECT trả về một hay nhiều bản ghi)



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

COMMAND (TIÉP)

Ví dụ: SqlCommand cmd = new SqlCommand(); cmd.Connection = con; cmd.CommandText = "UPDATE NhanVien set Luong = Luong + 100000 WHERE Manv = 01''; cmd.ExecuteNonQuery(); con.Close();

C KINH TÉ QUOC DA

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.6.4. DATAREADER

- Có dạng con trỏ, dùng để handle dữ liệu trả về từ CSDL
- Đặc điểm:
 - Con trỏ không thể lùi
 - Thường handle dữ liệu trả về từ phương thức ExecuteReader() của Command
- Có 2 loại:
 - SqlDataReader
 - OleDbReader
- Khuyến cáo:
 - Đối với các form chỉ SELECT dữ liệu, **NÊN DÙNG** DataReader để có tốc độ xử lý nhanh hơn



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

DATAREADER (TIÉP)

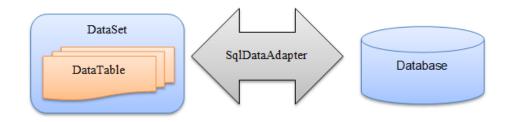
```
Ví du:
SqlCommand cmd = new SqlCommand();
cmd.Connection = con;
cmd.CommandText = "SELECT MaSV, Hoten FROM tblSinhvien";
SqlDataReader rd = cmd.ExecuteReader();
while (rd.Read())
  txtMaSV.Text = rd[0].ToString();
  txtHoten.Text= rd[1].ToString();
con.Close();
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.6.5. DATAADAPTER

Là cầu nối giữa CSDL và Dataset



- Các thuộc tính quan trọng:
 - SelectCommand
 - InsertCommand
- Các phương thức quan trọng:
 - Fill()
 - Update()
- Có cơ chế tự động đóng Connection

- UpdateCommand
- DeleteCommand



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

DATAADAPTER (TIÉP)

```
Ví dụ - Hiển thị dữ liệu:
string sql;
sql = "SELECT * from tblSinhVien";
//Đối tượng DataAdapter
SqlDataAdapter MyData = new SqlDataAdapter(sql,con);
tblSinhvien = new DataTable(); //Khởi tạo bảng
//Đổ dữ liệu từ DataAdapter vào bảng
MyData.Fill(tblSinhvien);
dataGridView.DataSource = tblSinhvien;
```

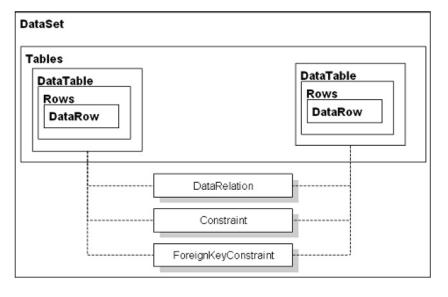


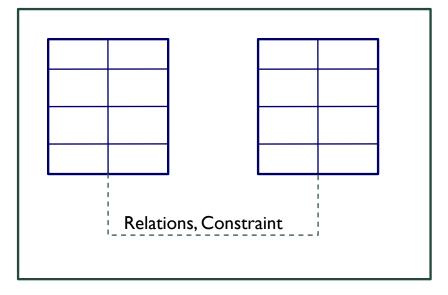
VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

4.6.6. DATASET

Là đối tượng lưu dữ liệu trả về từ CSDL



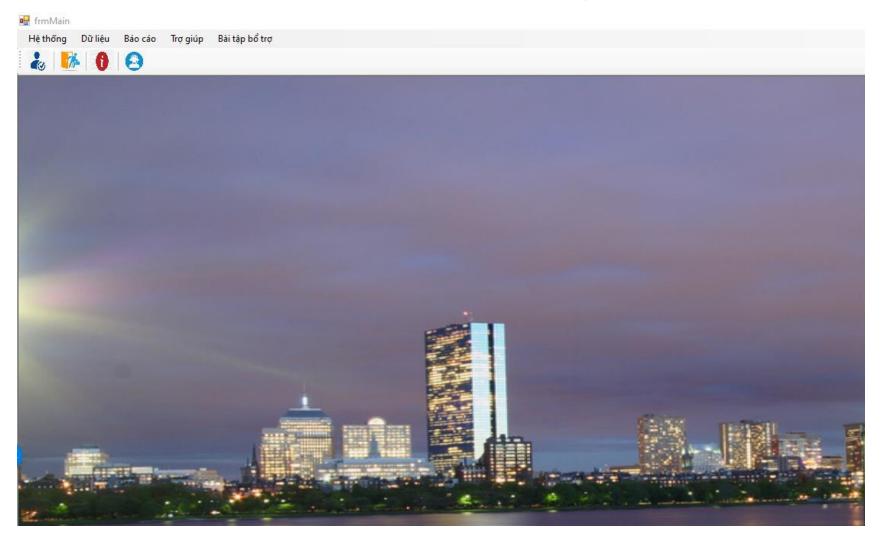




Dataset

Database

HƯỚNG DẪN THỰC HÀNH LẬP TRÌNH KẾT NỐI VÀ KHAI THÁC CSDL SQL SERVER





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

KHAI BÁO THƯ VIỆN SỬ DỤNG

- using System.Data;
- using System.Data.SqlClient;
- using System.Windows.Forms;
- using CrystalDecisions.CrystalReports.Engine;



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

KHAI BÁO BIẾN

```
    public partial class frmDMHH: Form
    {
    SqlConnection conn = new SqlConnection();
    SqlDataAdapter da = new SqlDataAdapter();
    SqlCommand cmd = new SqlCommand();
    DataTable dt = new DataTable();
    string sql, constr;
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

KÉT NÓI VỚI CSDL

Thường được làm khi FormLoad

constr = "Data Source=113.190.234.205,1433;Initial Catalog=ktqd2019;Persist Security Info=True;User ID=ktqd;Password=Ktqd-2019";

conn.ConnectionString = constr;

conn.Open();



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

ĐOẠN LÊNH SAU LÀM VIỆC GÌ?

```
SqlConnection conn = new SqlConnection("Data Source=113.190.234.205,1433;Initial Catalog=ktqd2019;Persist Security Info=True;User ID=ktqd;Password=Ktqd-2019");
string sql = "Select * from tblPass where UserName='" + txtusername.Text +
           "' and Password='" + txtpass.Text + "";
 conn.Open();
 SqlCommand cmd = new SqlCommand(sql, conn);
 SqlDataAdapter adapt = new SqlDataAdapter(cmd); DataSet ds = new DataSet();
 adapt.Fill(ds); conn.Close();
int count = ds.Tables[0].Rows.Count;
      if (count == 1)
           MessageBox.Show("..... Successful!"); this.Hide();
         else
           MessageBox.Show("..... Failed!");
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

THỦ TỤC THỰC HIỆN MỘT CẦU SQL TƯƠNG TÁC

```
public static void DoSQL(string sql)
string strcn = "Data Source=113.190.234.205,1433;Initial Catalog=ktqd2019;Persist Security
  Info=True;User ID=ktqd;Password=Ktqd-2019";
         SqlConnection conn = new SqlConnection(strcn);
conn.Open();
         SqlCommand cmd = new SqlCommand(sql, conn);
cmd.ExecuteNonQuery();
conn.Close();
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

NẠP DỮ LIỆU VÀO GRID

```
sql = "Select * From tbIDMHH";
da = new SqlDataAdapter(sql, conn);
da.Fill(dt);
grdData.DataSource = dt;
grdData.Refresh();
```





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

NÚT ĐIỀU HƯỚNG DỮ LIỆU TRÊN Ô LƯỚI

```
private void btnPrv_Click(object sender, EventArgs e)
      int i = Convert.ToInt16(grdData.CurrentRow.Index.ToString());
      if (i>0)
      grdData.CurrentCell = grdData[0,i - 1];
      NapCT();
private void btnFirst_Click(object sender, EventArgs e)
      grdData.ClearSelection();
      grdData.CurrentCell = grdData[0,0];
    NapCT();
```





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

NÚT ĐIỀU HƯỚNG DỮ LIỆU TRÊN Ô LƯỚI

```
private void btnNext_Click(object sender, EventArgs e)
       int i = Convert.ToInt16(grdData.CurrentRow.Index.ToString());
       if (i < grdData.RowCount-1)
         grdData.CurrentCell = grdData[0, i + 1];
         NapCT();
private void btnLast_Click(object sender, EventArgs e)
      grdData.CurrentCell = grdData[0, grdData.RowCount-1];
       NapCT();
```

C KINH TE QUOC OR

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

LẤY DỮ LIỆU TỪ DÒNG, CỘT CHỈ ĐỊNH TRÊN Ô LƯỚI

```
public void NapCT()
int i = grdData.CurrentRow.Index;//lấy số thứ tự dòng hiện thời
           txtMaNhom.Text = grdData.Rows[i].Cells["MaNhom"].Value.ToString();
txtTenHang.Text = grdData.Rows[i].Cells["TenHH"].Value.ToString();
txtDonVi.Text = grdData.Rows[i].Cells["Dvt"].Value.ToString();
           txtDonGia.Text = grdData.Rows[i].Cells["Dgvnd"].Value.ToString();
           txtNuocSX.Text = grdData.Rows[i].Cells["SanXuat"].Value.ToString();
txtMaHang.Text = grdData.Rows[i].Cells["MaHH"].Value.ToString();
```

j = Convert.ToInt16(grdData.RowCount.ToString()); //đém số dòng



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

NẠP DỮ LIỆU CHO DATA COMBO BOX

```
Hiện tên lấy mã
   private void comTruong_SelectedIndexChanged(object sender, EventArgs e)
sql = "Select Distinct" + comTruong.Text + " From tblDmHH ";
DataTable comtb = new DataTable();
da = new SqlDataAdapter(sql, conn);
da.Fill(comtb);
comGT.DataSource = comtb;
comGT.DisplayMember = comTruong.Text;
comGT.ValueMember = comGT.Text;
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TÌM KIẾM HAY LỌC DỮ LIỆU

```
private void btnLoc_Click(object sender, EventArgs e)
dt.Clear();
         sql = "Select * From tbIDMHH where " + comTruong.Text + "='" + comGT.Text + "'";
da = new SqlDataAdapter(sql, conn);
da.Fill(dt);
grdData.DataSource = dt;
         grdData.Refresh();
NapCT();
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

NÚT THÊM MỚI

```
private void btnnew_Click(object sender, EventArgs e)
      txtMaNhom.Text = "";
      txtTenHang.Text = "";
      txtDonVi.Text = "";
      txtDonGia.Text = "";
      txtNuocSX.Text = "";
      txtMaHang.Text = "";
      txtMaNhom.Focus();
      addnewFlag = true;
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

NÚT XÓA

```
private void btndel Click(object sender, EventArgs e)
       string mahang;
       if (MessageBox.Show("Ban có chắc chắn muốn xóa bản ghi hiện thời?", "Xác nhận yêu cầu",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)
         int i = Convert.ToInt16(grdData.CurrentRow.Index.ToString());
         mahang = grdData.Rows[i].Cells[1].Value.ToString();
         sql = "Delete from tbIDMHH where maHH='" + mahang + "'";
         clsMain.DoSQL(sql); dt.Clear();
         sql = "Select * From tbIDMHH";
         da = new SqlDataAdapter(sql, conn);
         da.Fill(dt); grdData.DataSource = dt;
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CẬP NHẬT SỬA CHỮA

```
if (addnewFlag == false)
         n = Convert.ToInt16(grdData.RowCount.ToString());
        for (i = 0; i < n - 1; i++)
           tManhom = grdData.Rows[i].Cells["MaNhom"].Value.ToString();
           tTenHH = grdData.Rows[i].Cells["TenHH"].Value.ToString();
           TDonGia = grdData.Rows[i].Cells["Dgvnd"].Value.ToString();
           tSanXuat = grdData.Rows[i].Cells["SanXuat"].Value.ToString();
           tMaHang = grdData.Rows[i].Cells["MaHH"].Value.ToString();
           sql = "Update tblDMHH set MaNhom='" + tManhom + "',TenHH=N'" + tTenHH + "',Dgvnd='" + TDonGia
               + "',SanXuat=N'" + tSanXuat + "' where Mahh='" + tMaHang + "'";
           clsMain.DoSQL(sql);
         MessageBox.Show("Đã cập nhật thành công!");
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CẬP NHẬT THÊM MỚI

```
if (addnewFlag == true)
           tManhom = txtMaNhom.Text;
           tTenHH = txtTenHang.Text;
           TDonGia = txtDonGia.Text;
           tSanXuat = txtNuocSX.Text;
           tMaHang = txtMaHang.Text;
           tdvt = txtDonVi.Text;
sql = "Insert into tbIDMHH (MaNhom, MaHH, TenHH, Dvt, Dgvnd, SanXuat) Values ('" + tManhom + "',N'" + tMaHang + "',N'" + tTenHH + "','" + tdvt + "'," + TDonGia + ",N'" + tSanXuat + "')";
           clsMain.DoSQL(sql);
           NapLai();
           addnewFlag = false;
```

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

Chương 5 Tạo báo cáo



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỤC ĐÍCH

 Trang bị kiến thức và hướng dẫn sinh viên thực hành tạo các loại báo cáo đầu ra của phần mềm ứng dụng trong quản lý



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

2.8 TẠO BÁO CÁO VỚI CRYSTAL REPORT

- 1. Thiết kế báo cáo
- 2. Gọi hiển thị báo cáo từ visual c#



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC THÀNH PHẦN CỦA 1 BÁO CÁO

Report Header

Page Header

Details

Report Footer

Page Footer

Report Header

Page Header

Details

Page Footer

Page Header

Details

Report Footer

Page Footer

Report 1 trang

Report nhiều trang



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC LOẠI BÁO CÁO

- Báo cáo chi tiết đầy đủ
 - Số dòng trong báo cáo tương ứng với số bản ghi trong bảng dữ liệu. Vd in danh sách sinh viên
- Báo cáo chi tiết có chọn lọc
 - Nội dung báo cáo chỉ chứa những bản ghi thỏa mãn điều kiện lọc. Ví dụ in danh sách sinh viên được điểm giỏi
- Báo cáo tổng hợp
 - Mỗi dòng trong báo cáo đại diện cho một nhóm bản ghi trong bảng dữ liệu. Ví dụ in bảng tổng hợp doanh thu bán hàng theo tên hàng.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

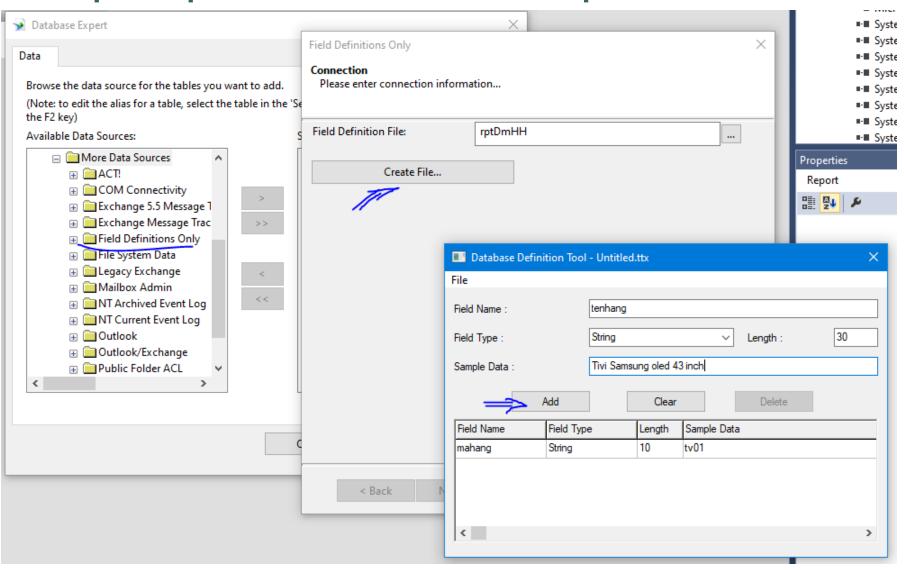
2.8.1 THIẾT KẾ BẢN MẪU DỮ LIỆU CHO BÁO CÁO

- Khởi đầu chọn Crystal Report \ Menu File \ New. Chọn Blank Report
- Trong cửa sổ Data Explorer\ More Data Sources \ Active Data \ Active Data (Field Definition Only)\ Select Data Source. Nhấn Browse để chọn bản mẫu dữ liệu được thiết kế trước hoặc nút New để tạo mới một bản mẫu dữ liệu
- Trong cửa sổ thiết kế Data Definiton Tools lần lượt khai báo thông tin các trường về Field Name, Field Type, Sample Data
- Sau khi thiết kế xong bản mẫu dữ liệu, ghi nó dưới dạng 1 tệp(có phần mở rộng là .ttx)
 trong thư mục Report của dự án
- Bấm Add để thêm bản mẫu dữ liệu vào báo cáo mới. Bấm Close để đóng cửa sổ dữ liệu này lại.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MINH HỌA TẠO BẢN MẪU DỮ LIỆU





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

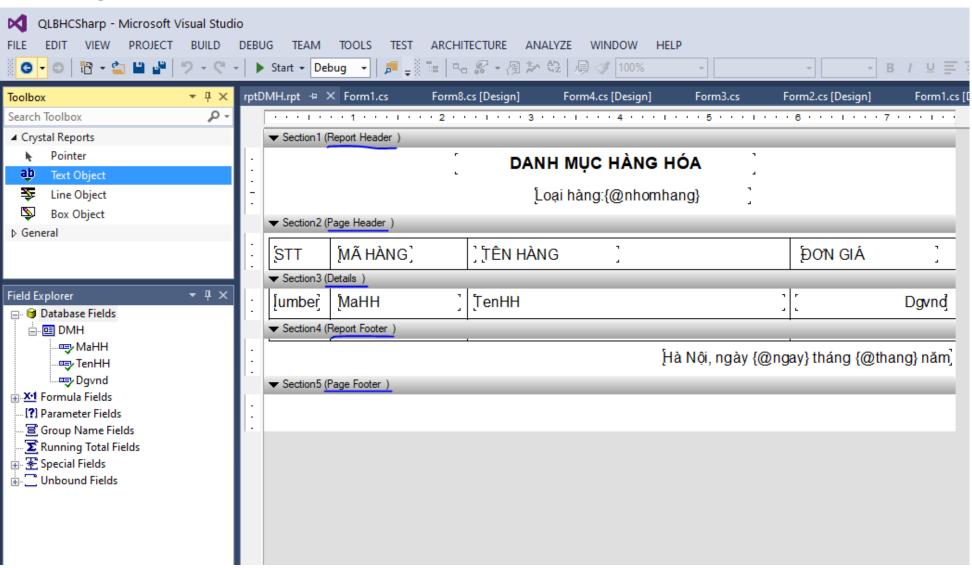
2.8.2 THIẾT KẾ BÁO CÁO

- Vẽ khung báo cáo:
 - Insert line để thêm đường kẻ vào báo cáo
 - Insert box để kẻ khung trong báo cáo
 - Insert Picture để thêm hình ảnh vào báo cáo
 - Insert Text Object để thêm các dòng văn bản vào báo cáo
- Để thêm các trường vào mục chi tiết của báo cáo chọn Insert\ Field Object hoặc Insert Field
- Thêm trường công thức Formula Field
- Thêm các trường đặc biệt: STT, số trang, số tổng hợp...



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MINH HỌA THIẾT KẾ BÁO CÁO





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

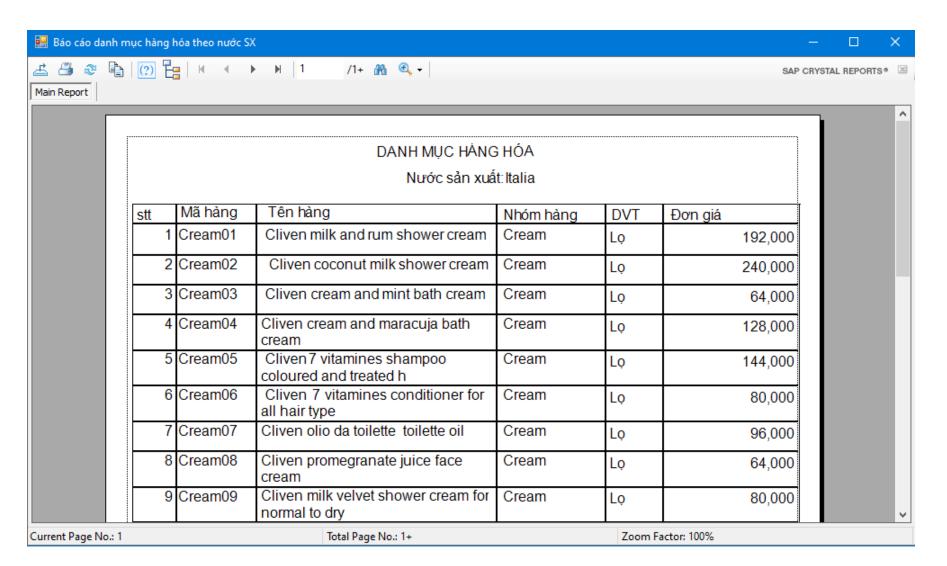
2.8.3 TRUYỀN DỮ LIỆU VÀ HIỂN THỊ BÁO CÁO

```
private void btnInSX_Click(object sender, EventArgs e)
      rptDMNuocSX rpt = new rptDMNuocSX ();
   sql = "Select MaHH, TenHH, MaNhom, Dvt, Dgvnd From tblDMHH where SanXuat='" + comGT.Text + "'";
      DataTable rpdata = new DataTable();
      rpdata.Clear();
      da = new SqlDataAdapter(sql, conn);
      da.Fill(rpdata);
      rpt.DataDefinition.FormulaFields["NuocSX"].Text = """ + comGT.Text + """;
      rpt.SetDataSource(rpdata);
      frmNuocSXrp rp = new frmNuocSXrp(rpt);
      rp.Show();
```



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

BÁO CÁO MINH HỌA





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CHƯƠNG 6

VẤN ĐỀ AN TOÀN VÀ BẢO MẬT CỦA CÁC ỨNG DỤNG TRONG QUẢN LÝ



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỤC ĐÍCH

- Giới thiệu về tầm quan trọng và ý nghĩa của việc bảo mật ứng dụng trong quản lý
- Cung cấp ý tưởng giúp sinh viên xây dựng hệ thống bảo mật cho ứng dụng bao gồm:
 - Phương án giữ bản quyền phần mềm
 - Phương án bảo mật dữ liệu của ứng dụng



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TẦM QUAN TRỌNG CỦA BẢO MẬT

- Sản phẩm phần mềm chỉ có thể bán được nếu nhà sản xuất có phương án đăng ký và giữ bản quyền phù hợp
- Thông tin, dữ liệu ngày càng có vai trò quan trọng đối với mỗi tổ chức doanh nghiệp vì thế nó cần phải được bảo mật
- Số lượng dữ liệu phát sinh ngày càng nhiều, nguy cơ rò rỉ, trùng lặp dữ liệu ngày càng cao vì thế nó cần được bảo mật để tránh thất thoát
- Việc cạnh tranh không lành mạnh thường dẫn đến hành động xâm nhập, ăn cắp, phá hoại dữ liệu, phần mềm=> cần phải có phương án bảo mật phù hợp.
- Ngày càng nhiều công cụ, phương pháp phá khóa phần mềm được tạo ra=> không ngừng nâng cấp, cải tiến các giải pháp bảo mật. Không bao giờ có giải pháp bảo mật tuyệt đối và bền vững trong thời gian dài.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

HAI NHIỆM VỤ BẢO MẬT CẦN THỰC HIỆN

- 1. Giữ bản quyền đối với sản phẩm phần mềm
 - Nhằm khiến cho người dùng buộc phải trả tiền nếu muốn sử dụng phần mềm
 - Theo dõi và kiểm soát được danh sách người dùng
- 2. Bảo mật thông tin, dữ liệu chứa trong cơ sở dữ liệu của phần mềm
 - Ngăn chặn được sự truy cập không được phép của người dùng
 - Xác định được trách nhiệm của cá nhân người dùng khi dữ liệu bị xâm phạm, rò rỉ



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC PHƯƠNG ÁN GIỮ BẢN QUYỀN PHẦN MỀM

- Đặt mật khẩu cho bộ cài đặt
- Tạo ra một bộ các mã kích hoạt phần mềm (active key) và người dùng sẽ trả tiền để mua các mã kích hoạt này.
- Tạo bộ mã kích hoạt dựa trên tên người dùng và email đăng ký
- Tạo mã kích hoạt dựa trên số serial của thiết bị phần cứng của máy tính. Ví dụ ổ cứng, bộ vi xử lý...
- Bản quyền phần mềm được cấp cho từng máy tính dựa trên số serial của ổ cứng nơi cài đặt phần mềm. Khi phần mềm hoạt động, có 1 modul sẽ kết nối với máy chủ trung tâm để xác thực bản quyền của máy tính tương ứng, nếu phát hiện gian lận thì sẽ không cho phép phần mềm hoạt động tiếp tục.
- Phát hành Khóa cứng (hash key) chứa mã số xác thực được mã hóa bằng phần cứng mà chỉ khi nào người dùng cắm khóa cứng(giống cái usb) vào máy thì phần mềm tương ứng mới có thể chạy được. Bản quyền phần mềm được bán thông qua việc bán các khóa cứng này.

OC KINH TË QUOCO P.N.

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC PHƯƠNG ÁN BẢO MẬT DỮ LIỆU

- Đặt mật khẩu đăng nhập cho phần mềm; đăng nhập quá 3 lần sẽ tự kết thúc.
- Tạo danh mục người dùng, mỗi người dùng có 1 tài khoản đăng nhập riêng
- Tạo danh mục người dùng, mỗi người sẽ thuộc vào 1 trong 3 nhóm đối tượng: Cán bộ thao tác vận hành; Cán bộ lãnh đạo và Cán bộ quản trị (admin). Mỗi nhóm người dùng sẽ được phân những quyền hạn cụ thể tương ứng với nhu cầu và đặc điểm công việc của mình.
- Thực hiện phân quyền cho từng người dùng chi tiết tới từng mục menu chức năng. Mỗi người dùng khác nhau sẽ có những quyền hạn khác nhau đối với phần mềm. Phương án này không bị giới hạn số nhóm đối tượng, tùy theo chính sách bảo mật dữ liệu của công ty để phân chia thành các nhóm người dùng khác nhau.
- Khi phần mềm hoạt động, luôn có 1 modul chức năng được vận hành với chức năng kết nối máy chủ và xác thực quyền hạn và danh tính của người dùng, việc xác thực này được thực hiện lặp lại sau mỗi khoản thời gian(vd 1 phút 1 lần hoặc ngắn hơn).
- Mỗi người dùng khác nhau sẽ được phát 1 thiết bị gọi là Khóa cứng. Đây là thiết bị được mã hóa duy nhất, không thể sao chép, không thể làm giả. Chỉ khi nào khóa cứng được lắp vào máy thì phần mềm tương ứng mới có thể vận hành.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CHƯƠNG 7

BIÊN SOẠN TÀI LIỆU HƯỚNG DẪN SỬ DỤNG



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỤC ĐÍCH

- Giới thiệu về vai trò, ý nghĩa của bộ tài liệu hướng dẫn sử dụng đối với phần mềm thương mại hóa
- Trang bị kiến thức và hướng dẫn sinh viên thực hành xây dựng bộ tài liệu hướng dẫn sử dụng và tích hợp nó vào phần mềm ứng dụng





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

VAI TRÒ CỦA MODUL HƯỚNG DẪN SỬ DỤNG

- Là điều kiện đủ để một phần mềm ứng dụng có thể thương mại hóa bán trên thị trường
- Là yếu tố tạo nên đẳng cấp của phần mềm
- Để rút ngắn thời gian trong đào tạo và truyển giao phần mềm
- Giúp giảm các lỗi logic trong quá trình sử dụng phần mềm từ đó giảm nhân lực hỗ trợ sau bán hàng
- Làm tăng doanh số, thị phần của sản phẩm phần mềm



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC LOẠI TRỢ GIÚP CÓ THỂ TÍCH HỢP VÀO PHẦN MỀM

- Tool tip text- trợ giúp nhanh
 - Hiện ra một đoạn văn bản giải thích hay giới thiệu chức năng của nút bấm, menu, ô tùy chọn,... trên giao diện chức năng mỗi khi người dùng di chuyển con trỏ lên phía trên đối tượng được trợ giúp
- Help contex- trợ giúp theo ngữ cảnh
 - Mỗi khi người dung bấm phím chức năng F1 hoặc chọn nút chức năng Help trong một giao diện chức năng nào đó thì cửa sổ trợ giúp sẽ hiện lên với nội dung hiển thị tương ứng với phần trợ giúp dành cho giao diện chức năng đó.
- Help book- sách tợ giúp
 - Như một cuốn sách điện tử chứa đầy đủ các nội dung trợ giúp có liên quan đến phầm mềm, có hệ thống mục lục và index thuận tiện để người dùng dễ dàng tìm được thông tin trợ giúp mong muốn.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC LOẠI HELP FILE

- Help file có đuôi "hlp" là chuẩn file trợ giúp cũ, phiên bản window hỗ trợ mới nhất là win7, sử dụng module winhlp32.exe để hiển thị
- Help file có đuôi "chm" được tạo theo chuẩn chung là HTLM Help và hiện đang dùng phổ biến. Để soạn thảo và biên dịch tệp CHM có thể dùng nhiều phần mềm:
 - HTML Help Workshop
 - Help and manual
 - PowerCHM
 - WinCHM
 - **-**



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

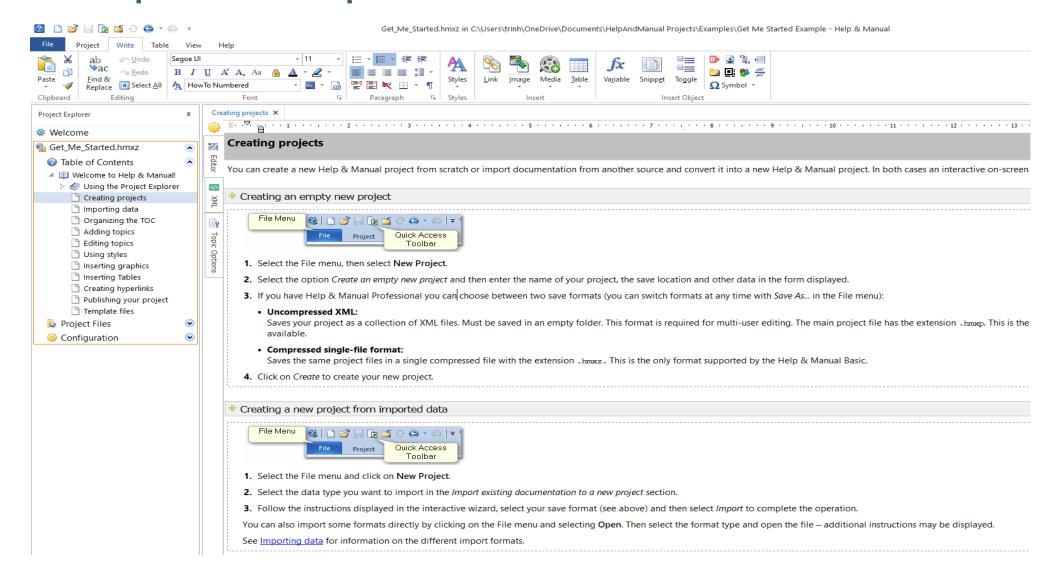
CÁC BƯỚC TẠO HTML HELP

- Bước 1: Chuẩn bị nội dung
 - Nội dung có thể hiển thị: văn bản; hình ảnh; biểu đồ; âm thanh; video; link;...
 - Tạo tệp word với hệ thống mục lục tự động tương ứng
- Bước 2: Tạo Help project sử dụng một phần mềm phù hợp
 - Html help table of contents: chứa hệ thống mục lục tương ứng với các chương mục của các nội dung help
 - Html help index: chứa danh sách các từ khóa giúp người dùng tìm nhanh được nội dung trợ giúp
 - Cắt dán các nội dung đã chuẩn bị trước vào những đề mục phù hợp
 - Biên dịch và tạo Help file CHM.
 - Chú ý: Để có thể biên dịch Helpfile CHM trước hết cần cài đặt Html help workshop



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

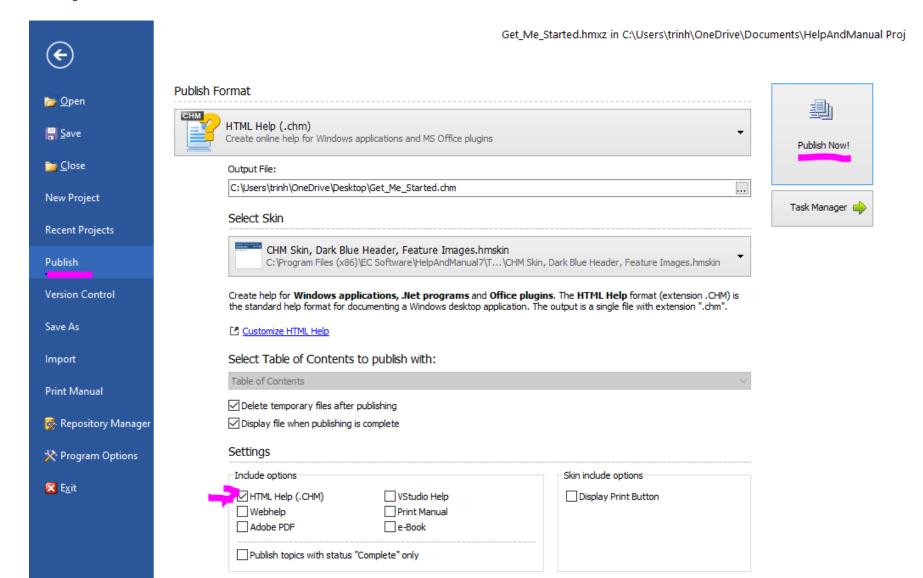
GIAO DIỆN LÀM VIỆC CỦA HELP AND MANUAL





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

BIÊN DỊCH FILE HELP





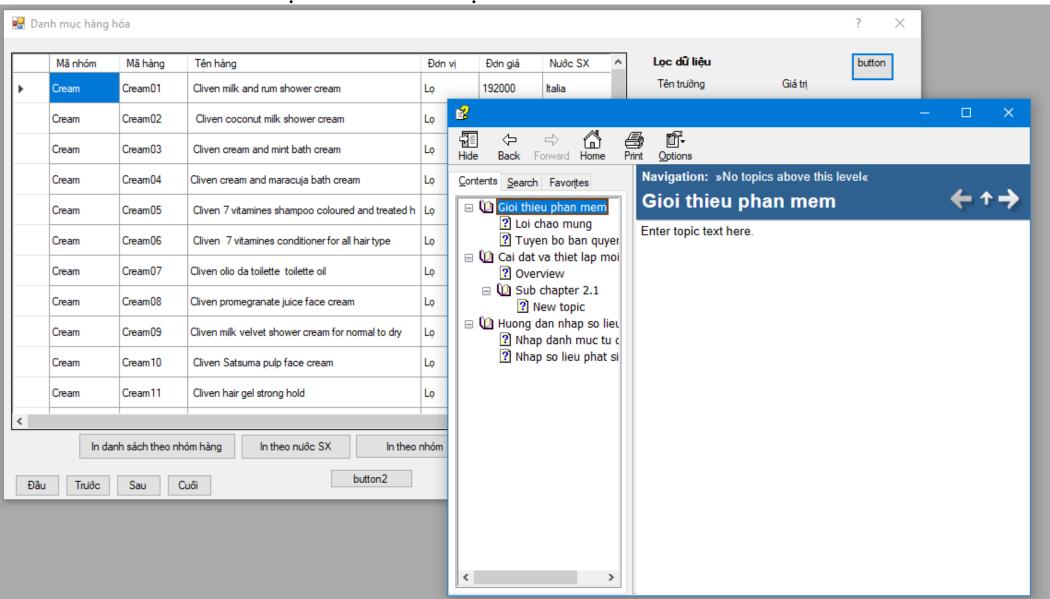
VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

GOI HIỂN THỊ FILE HELP

- Bước 1: Nên copy tệp help vào cùng thư mục với tệp thi hành của ứng dụng
- Thủ tục gọi file help sử dụng câu lệnh sau:
- Help.ShowHelp(this,
 "file://D:\\Dropbox\\ForMySelf\\BaiGiang\\PhatTrienUngDung\\QLBHCSharp\\bin\\Debug\\HelpQLBH.chm"); N\u00e9u mu\u00f3n d\u00ean d\u00ean t\u00eap help \u00e3 m\u00e9t thu mu\u00e5 cu th\u00e9 th\u00e1 làm turong t\u00c4 nhu tr\u00ean
- Help.ShowHelp(this, "HelpQLBH.chm"); N\u00e9u b\u00ean d\u00e4 copy t\u00eap help v\u00eao c\u00fcng thu muc v\u00f3i t\u00eap thi h\u00eanh hi chi c\u00ean vi\u00eat ng\u00ean th\u00ea n\u00eay l\u00ea dung thi chi c\u00ean vi\u00eat ng\u00ean th\u00ea n\u00eay l\u00ea dung th\u00ean chi c\u00ean vi\u00eat ng\u00ean th\u00ea n\u00eay l\u00ea dung th\u00ean chi c\u00ean vi\u00eat ng\u00ean th\u00ea n\u00eay l\u00ea dung th\u00ean c\u00ean vi\u00eat ng\u00ean th\u00ea n\u00eay l\u00ea dung th\u00ean vi\u00eat ng\u00ean th\u00ea n\u00eay l\u00ea dung th\u00ean th\u00ean vi\u00eat ng\u00ean th\u00ean n\u00eay l\u00ea dung th\u00ean th\u00ean vi\u00eat ng\u00ean th\u00ean n\u00ean th\u00ean n\u00ean th\u00ean n\u00ean th\u00ean n\u00ean th\u00ean n\u00ean t\u00ean n\u00ean th\u00ean n\u00ean t\u00ean n\u00ean n\



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ





VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CHƯƠNG 8 TẠO BỘ CÀI ĐẶT



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

MỤC ĐÍCH

- Giới thiệu về vai trò và ý nghĩa của việc tạo bộ cài đặt cho phần mềm ứng dụng thương mại hóa
- Khái quát các bước để tạo bộ cài đặt sử dụng công cụ có sẵn hoặc phần mềm của hãng thứ 3

CCKINH TË QUOCO DE

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

TỔNG QUAN

- Một phần mềm ứng dụng, được tạo bởi ngôn ngữ lập trình bậc cao, khi hoạt động thường sử dụng, truy xuất đến rất nhiều tệp liên quan:
 - Các tệp cơ sở dữ liệu
 - Các tệp chứa mẫu báo cáo
 - Các tệp hình ảnh(icons, pictures, wall papers...)
 - Tệp thư viện (dll, ocx...)
 - Tệp cấu hình
- Bộ cài đặt phần mềm là một tập hợp bao gồm tệp thị hành của phần mềm cùng với các tệp thành phần khác có liên quan như tệp cơ sở dữ liệu, tệp báo cáo, tệp thư viện... được bố trì để có thể cài đặt vào những thư mục phù hợp trên máy của người sử dụng sao cho phần mềm ứng dụng có thể hoạt động độc lập mà không cần đến môi trường phát triển của nó.



VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

LÝ DO CẦN TẠO BỘ CÀI ĐẶT

- Để có thể thiết lập môi trường hoạt động cho các modul của phần mềm ở trên máy của người dung nhanh và hoàn chỉnh
- Để tích hợp các tệp: cơ sở dữ liệu, thư viện, báo cáo, ...của phần mềm vào hệ thống thư mục trên máy người dung đáp ứng yêu cầu hoạt động của phần mềm
- Khi muốn cho, tặng, bán phần mềm ra thị trường
- Để bảo về bộ mã nguồn của phần mềm

CCUINH TE QUOCO PA

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN

VIỆN CÔNG NGHỆ THÔNG TIN VÀ KINH TẾ SỐ

CÁC BƯỚC TẠO BỘ CÀI ĐẶT

- Hoàn thiện bộ mã nguồn và các tệp thư viện có liên quan. Đảm bảo có thể biên dịch toàn bộ sang mã mãy mà không có lỗi
- 2. Chạy chương trình Deployment Wizard tích hợp sẵn trong bộ visual studio hoặc một phần mềm của bên thứ 3 ví dụ như InstallShield
- 3. Làm tuần tự theo các bước của trình wizard, khai báo các tham số có liên quan. Tạo các thư mục con trong thư mục cài đặt của dự án (ví dụ: thư mục Data, Report, Dll, Images) và đưa vào đó các tệp tương ứng của phần mềm.
- 4. Tạo shortcut cho ứng dụng cài đặt
- 5. Thiết lập tên công ty và thông số phiên bản của phần mềm
- 6. Đặt các thuộc tính cho gói cài đặt
- 7. Biên dịch và đóng gói dự án
- 8. Sửa các lỗi phát hiện trong quá trình biên dịch; biên dịch lại. Lặp lại như vậy cho đến khi bộ cài đặt được tạo thành công
- 9. Cài đặt thử trên một máy tính khác còn "trắng tinh" (mới chỉ cài đặt hệ điều hành). Chạy thử phần mềm trên máy mới. Nếu chưa đạt yêu cầu thì quay lại từ bước 3 để rà soát và làm lại.