



Microsoft®  
**Visual C#®**

---

CHUYÊN ĐỀ NGÔN NGỮ LẬP TRÌNH 1

***Người biên soạn:*** Hồ Quang Thái (MSCB: 2299)

BM. Công Nghệ Phần Mềm, Khoa CNTT&TT

***Email:*** [hqthai@cit.ctu.edu.vn](mailto:hqthai@cit.ctu.edu.vn)

***Số tín chỉ:*** 2 (20 LT + 20TH)



# CHƯƠNG 1

# **.NET FRAMEWORK**

# NỘI DUNG

- *Tổng quan*
- *Các thành phần chính **.NET***
- *Đặc điểm **.NET***
- *Lịch sử các phiên bản **.NET***
- *Các loại chương trình **.NET***
- *Chương trình **C#** đầu tiên*

# TỔNG QUAN

- Cuối những năm 1990, ngôn ngữ lập trình bị chia ra làm nhiều nhánh nhỏ.
- Hầu hết người lập trình sử dụng **Visual Basic**, **C** và **C++**.
- Một số lập trình viên C và C++ sử dụng thuần các hàm **API** của **Win32**, những hầu hết thì sử dụng **Microsoft Foundation Classes** (MFC). Một số khác thì chuyển đến **Component Object Model** (COM).

# TỔNG QUAN

- Tất cả các công nghệ kể trên đều có những vấn đề riêng:
  - Việc sử dụng thuần các hàm **API** của **Win32** thì không hỗ trợ lập trình hướng đối tượng và phải viết mã lệnh nhiều hơn **MFC**.
  - **MFC** thì hướng đối tượng nhưng sử dụng các thành phần **COM** đã quá cũ.

# TỔNG QUAN

Cần phải có một sự khởi đầu mới, tích hợp, hướng đối tượng và mang lại sự phù hợp với những ngôn ngữ lập trình đã tồn tại trước đó:

## MỤC TIÊU THỰC HIỆN

Bảo mật

Đa nền tảng

Hiệu năng

## MỤC TIÊU PHÁT TRIỂN

Tạo môi trường phát triển hướng đối tượng

Phù hợp với kinh nghiệp lập trình

Giao tiếp dùng chung một chuẩn công nghiệp

Triển khai đơn giản

Không phụ thuộc ngôn ngữ

Có khả năng tương tác

# TỔNG QUAN

- Năm 2002, **Microsoft** tung ra phiên bản .NET Framework, hứa hẹn giải quyết được nhiều vấn đề, chạm tới những hệ thống thế hệ mới.
- ***Phiên bản 3.0*** của **.NET Framework** được cài đặt sẵn trong *Windows Server 2008* và *Windows Vista*.
- ***Phiên bản 3.5*** được cài sẵn trong *Windows 7* và *Windows Server 2008 R2*, cũng có thể cài đặt cho *Windows XP* và *Windows Server 2003*.

# TỔNG QUAN

- Vào 12/04/2010, **.NET Framework 4** phát hành cùng *Visual Studio 2010*.
- Gia đình **.NET Framework** cũng bao gồm 2 phiên bản dành cho Mobile và thiết bị nhúng.
- Một phiên bản rút gọn, **.NET Compact Framework**, sử dụng cho nền tảng **Windows CE**, **.NET Mirco Framework** là một phiên bản dành cho các thiết bị có tài nguyên bị hạn chế.
- Phiên bản **.NET Framework 4.5.x** không còn hỗ trợ *Windows XP*.



# ĐẶC ĐIỂM .NET

- **Interoperability** (*khả năng tương tác*):
  - Có thể tương tác giữa hệ thống cũ và mới, truy cập đến các chức năng của các chương trình cũ và mới để thực thi bên ngoài môi trường **.NET Framework**.
  - Truy cập các thành phần **COM** sử dụng namespace **System.Runtime.InteropServices** và **System.EnterpriseServices**.
  - Truy cập đến các chức năng đã được đóng gói lại thông qua tính năng **Platform Invocation Services** (*Dịch vụ truy cập các nền tảng*).

# ĐẶC ĐIỂM .NET

- **Common Language Runtime engine** (*Bộ máy thực thi bằng ngôn ngữ chung*):
  - **Common Language Runtime (CLR)** được xem như là bộ máy thực thi của **.NET**.
  - Tất cả các chương trình **.NET** đều nằm dưới sự giám sát của **CLR** để bảo đảm các tính chất và hành vi nhất định trong việc *quản lý bộ nhớ, bảo mật, cơ chế bắt lỗi và sửa lỗi* trong khi thực thi chương trình viết bằng **.NET**.

# ĐẶC ĐIỂM .NET

- **Language independence** (*không phụ thuộc ngôn ngữ*):
  - .NET Framework giới thiệu **Common Type System** (CTS – *Kiểu hệ thống chung*) và **Common Language Infrastructure** (CLI – *Ngôn ngữ cơ sở chung*) để khi tất cả các ứng dụng được thực thi, chúng sẽ được chuyển đổi và tương tác qua lại giữa các kiểu dữ liệu và ngôn ngữ được hỗ trợ.
  - Với tính năng này, các chương trình sẽ có thể sử dụng, thừa kế các thư viện và ứng dụng được viết bằng các ngôn ngữ khác nhau của **.NET**.

# ĐẶC ĐIỂM .NET

- **Base Class Library** (*Thư viện lớp cơ sở*):
  - **Base Class Library (BCL)** là một thư viện chức năng được viết sẵn tồn tại dưới tất cả các ngôn ngữ mà **.NET** hỗ trợ.
  - Các lớp, giao tiếp này có thể được tái sử dụng và đã được đóng gói một số lượng lớn các hàm phổ biến, bao gồm việc đọc và ghi tập tin, kết xuất đồ họa, tương tác cơ sở dữ liệu, thao tác với tài liệu XML,...

# ĐẶC ĐIỂM .NET

- **Simplified deployment** (*Triển khai đơn giản*):
  - **.NET Framework** bao gồm tính năng thiết kế giúp quản lý và cài đặt ứng dụng và chắc rằng nó sẽ ko can thiệp với các phần mềm đã được cài đặt trước đó, điều này phù hợp với yêu cầu bảo mật.

# ĐẶC ĐIỂM .NET

- **Security** (*Bảo mật*):
  - Mục tiêu thiết kế hướng đến giải quyết một số vấn đề về bảo mật, như là lỗi **buffer overflows** (*tràn bộ đệm*), có thể bị khai thác bởi các phần mềm độc hại.
  - Thêm vào đó, **.NET** cũng đề xuất một mô hình chung cho tất cả ứng dụng.

# ĐẶC ĐIỂM .NET

- **Portability** (*Tính khả chuyển*):
  - Microsoft không cho phép cài **.NET** vào bất kỳ hệ thống nào ngoài **Windows**. Tuy nhiên, một số thành phần của .NET có thể được sử dụng đa nền như **Silverlight**.
  - Microsoft đã nộp đặc tả của **CLI** (bao gồm thư viện lớp chính, CTS, CIL đến ECMA và ISO, biến chúng thành những chuẩn chính thức. Để chắc chắn rằng một bên thứ 3 có thể triển khai các ứng dụng đến các nền tảng khác.

# .NET FRAMEWORK

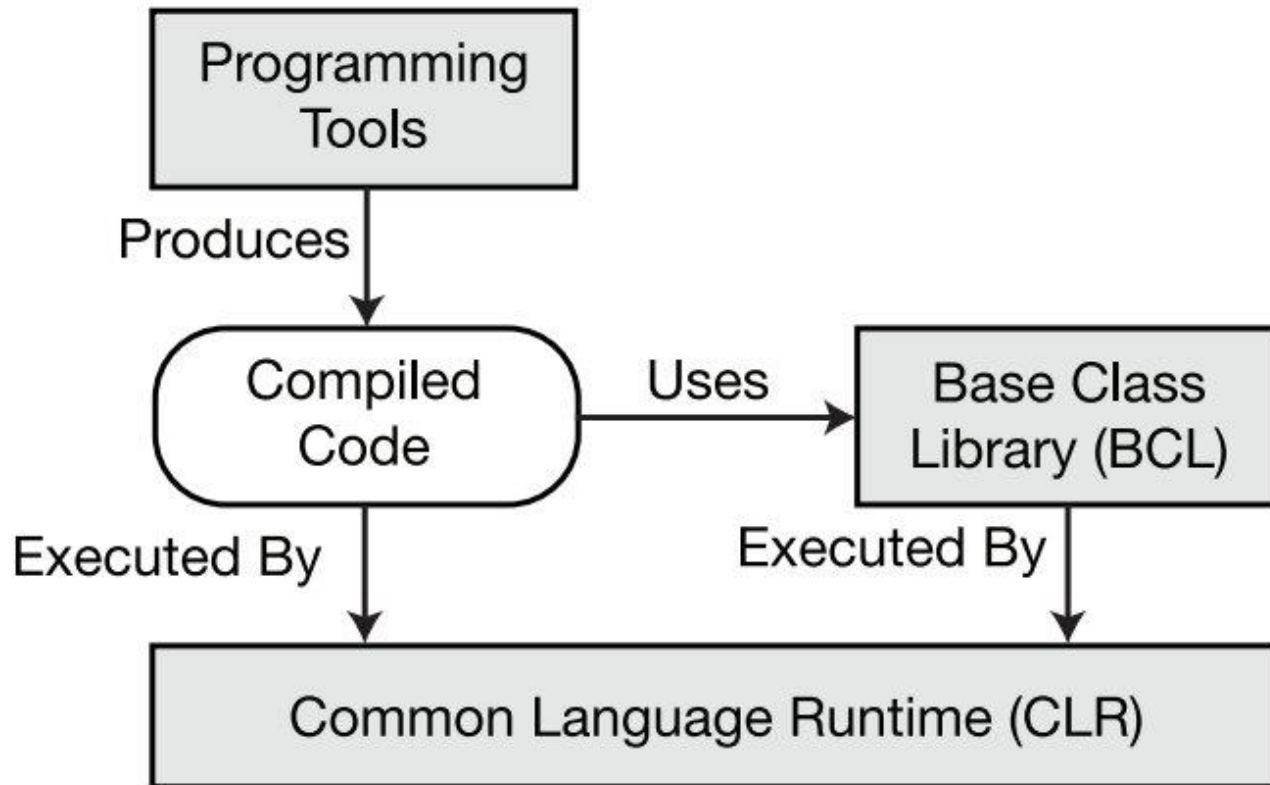
- Là một thành phần phần mềm được thêm vào hệ điều hành (Windows).
  - Chứa đựng những thư viện có sẵn.
  - Quản lý việc thực thi chương trình viết dưới nền tảng .NET
- **.NET Framework** quản lý việc thực thi các chương trình được viết dựa trên **.NET Framework** do đó người dùng cần phải cài **.NET Framework** để có thể chạy các chương trình được viết trên nền *.NET*.



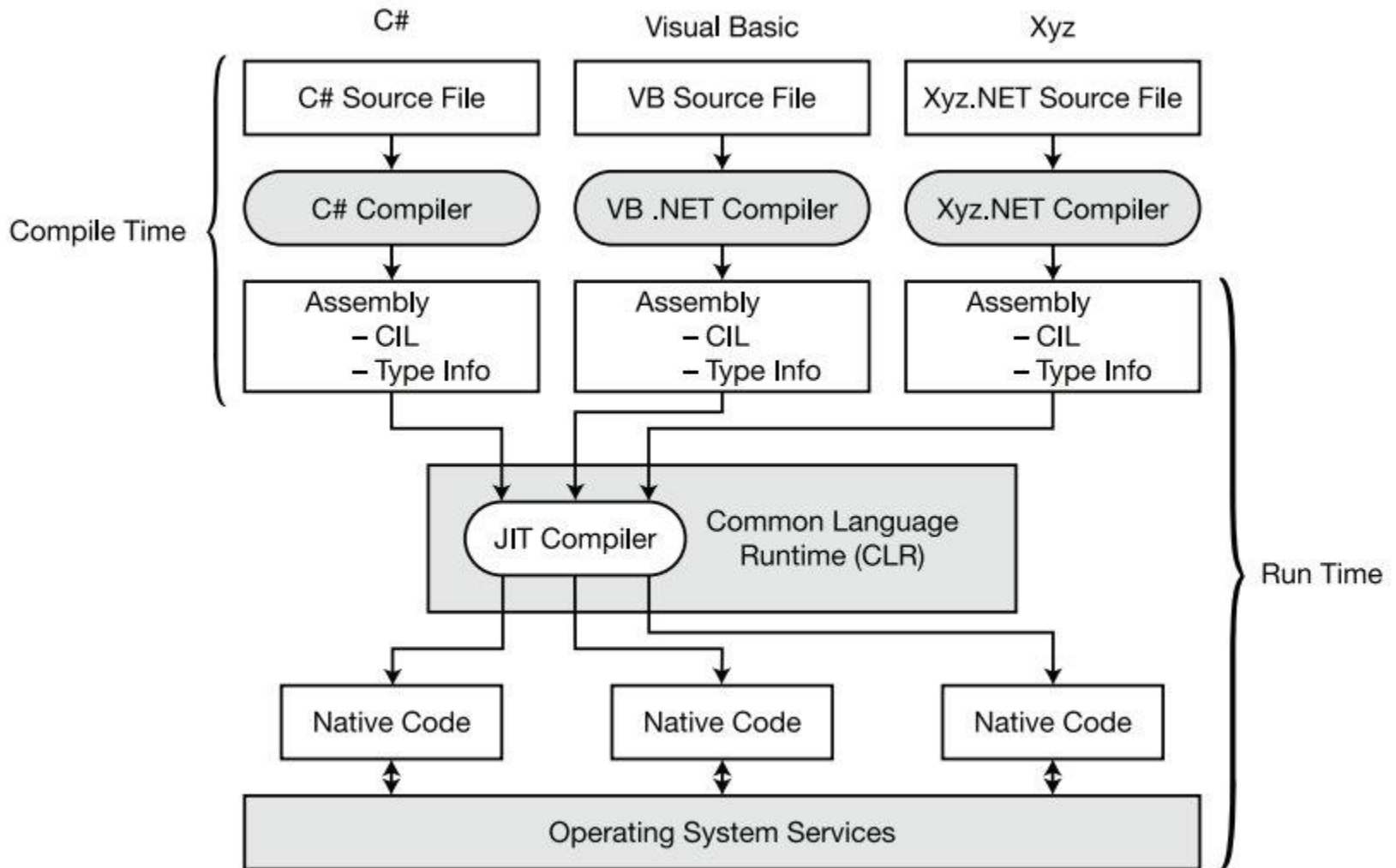
# CÁC THÀNH PHẦN .NET FRAMEWORK

- **.NET Framework** gồm 3 thành phần chính:
  - *Base Class Library (BCL)* : tập hợp các lớp thư viện được định nghĩa sẵn.
  - *Common Language Runtime (CLR)*: quản lý việc thực thi chương trình: quản lý bộ nhớ, thực thi mã lệnh, cấp phát và thu hồi vùng nhớ,...
  - *Programming Tools*: Bao gồm mọi thứ để viết mã lệnh và bắt lỗi: Visual Studio IDE, trình biên dịch các ngôn ngữ .NET (C#, VB.NET, F#, IronRuby và C++)

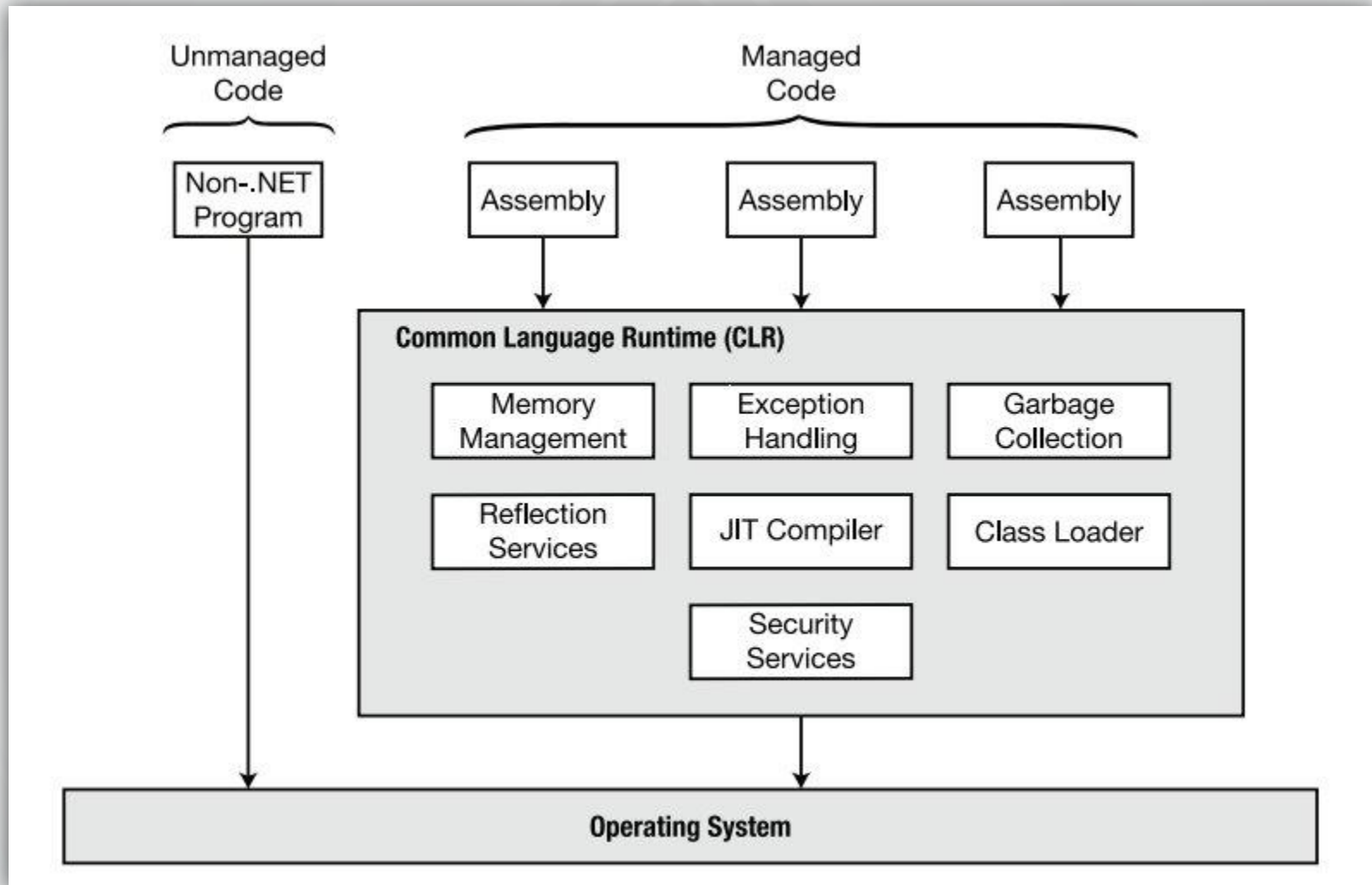
# CÁC THÀNH PHẦN .NET FRAMEWORK



# CÁC THÀNH PHẦN .NET FRAMEWORK



# CÁC THÀNH PHẦN .NET FRAMEWORK



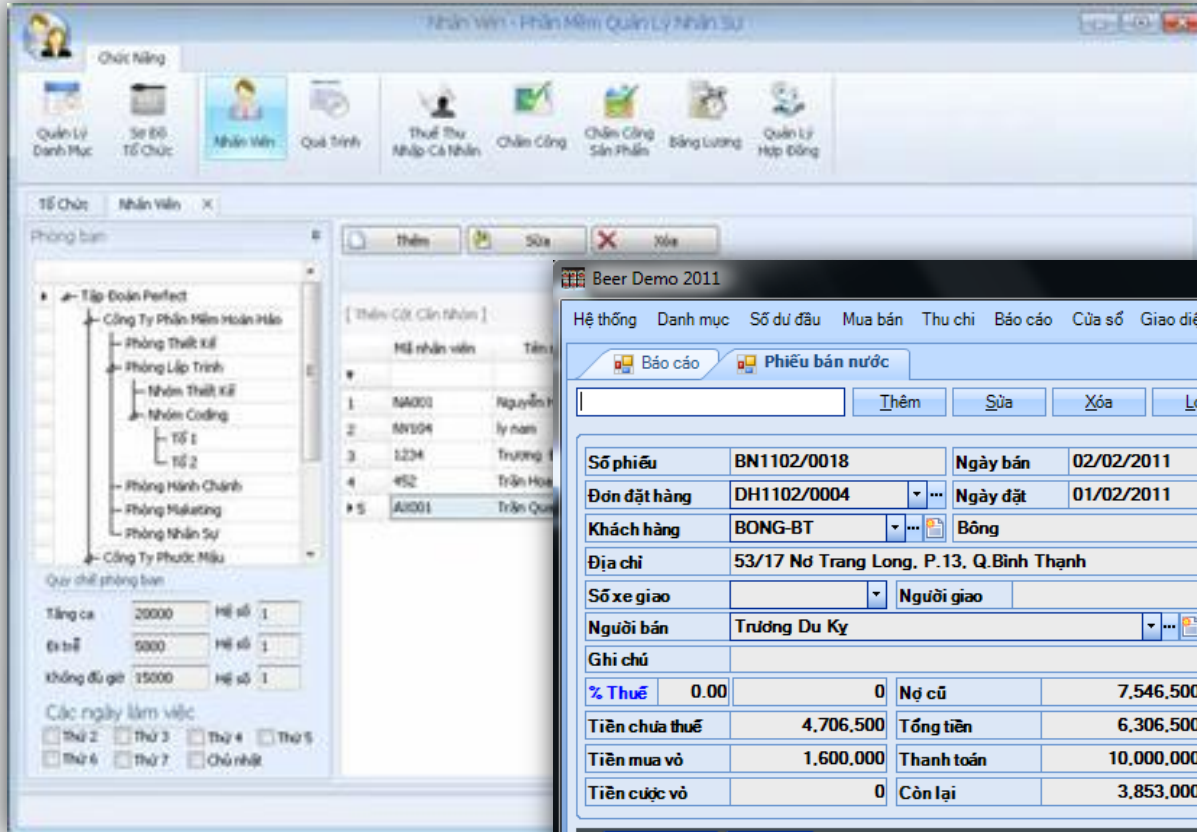
# CÁC PHIÊN BẢN .NET FRAMEWORK

Phiên bản	Số phiên bản	CLR	Xuất bản	Công cụ phát triển	Thay thế
1.0	1.0.3705.0	1.0	2002-02-13	Visual Studio .NET	N/A
1.1	1.1.4322.573	1.1	2003-04-24	Visual Studio .NET 2003	1.0
2.0	2.0.50727.42	2.0	2005-11-07	Visual Studio 2005	N/A
3.0	3.0.4506.30	2.0	2006-11-06	Microsoft Blend	2.0
3.5	3.5.21022.8	2.0	2007-11-19	Visual Studio 2008	2.0, 3.0
4.0	4.0.30319.1	4	2010-04-12	Visual Studio 2010	N/A
4.5	4.5.50709.17929	4	2012-08-15	Visual Studio 2012	4.0
4.5.1	4.5.50938.18408	4	2013-10-17	Visual Studio 2013	4.0, 4.5
4.5.2	4.5.51641	4	2014-05-05	Visual Studio 2013	4.0, 4.5, 4.5.1

# CÁC LOẠI ỨNG DỤNG PHỔ BIẾN

- Console Application
- Class Library
- Windows Forms Application
- Windows Presentation Foundation (WPF)
- ASP.NET
- Silverlight
- Windows và Universal Apps (*dành cho thiết bị cảm ứng*)
- Windows Phone

# WINDOWS FORMS



Beer Demo 2011

Hệ thống | Danh mục | Số dư đầu | Mua bán | Thu chi | Báo cáo | Cửa sổ | Giao diện

Báo cáo | **Phiếu bán nước**

Thêm | Sửa | Xóa | Lọc | In | Đóng | Cập nhật vò

Số phiếu	BN1102/0018	Ngày bán	02/02/2011
Đơn đặt hàng	DH1102/0004	Ngày đặt	01/02/2011
Khách hàng	BONG-BT	Bông	
Địa chỉ	53/17 Nơ Trang Long, P.13, Q.Bình Thạnh		
Số xe giao		Người giao	
Người bán	Trương Du Kỳ		
Ghi chú			
% Thuế	0.00	Nợ cũ	7.546,500
Tiền chưa thuế	4,706,500	Tổng tiền	6,306,500
Tiền mua vò	1,600,000	Thanh toán	10,000,000
Tiền cược vò	0	Còn lại	3,853,000

Dòng: 1 của 1

1 Hàng hóa | 2 Bao bì

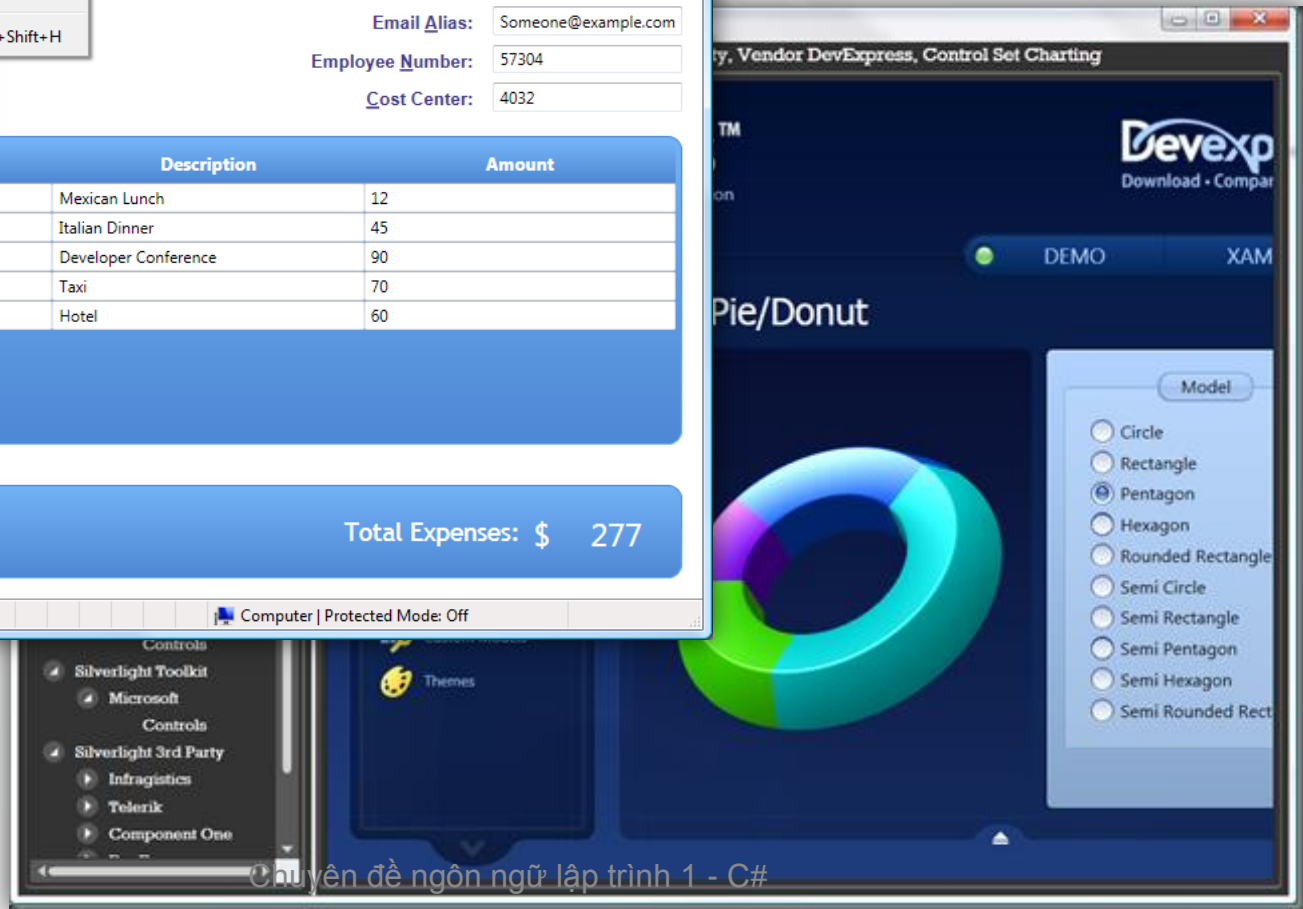
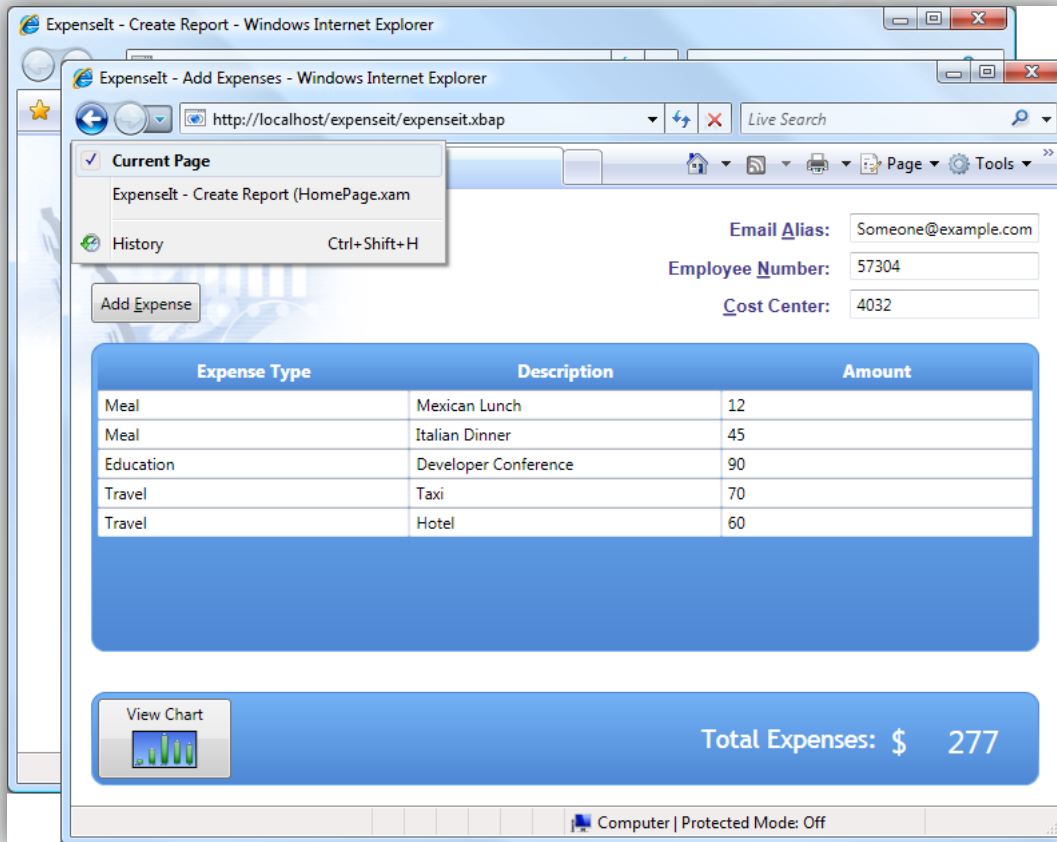
Mã kho	Mã hàng	Tên hàng	ĐVT	K.mãi	Chẩn đi	Đơn giá
K001	SG-BIA-EXP-1-355	Sài gòn++Export-chai-355	K20		30	111,5
K001	BT-N1Z-CAM EP-1-240	BT++N1-Cam Ép-chai-240	K24		20	68,0
Σ						

Dòng: 1 của 2

Người tạo: . Ngày tạo: . Người sửa: Admin . Ngày sửa: 17/ . Administrator - BEERDEM . Kỳ: 201102 NUM

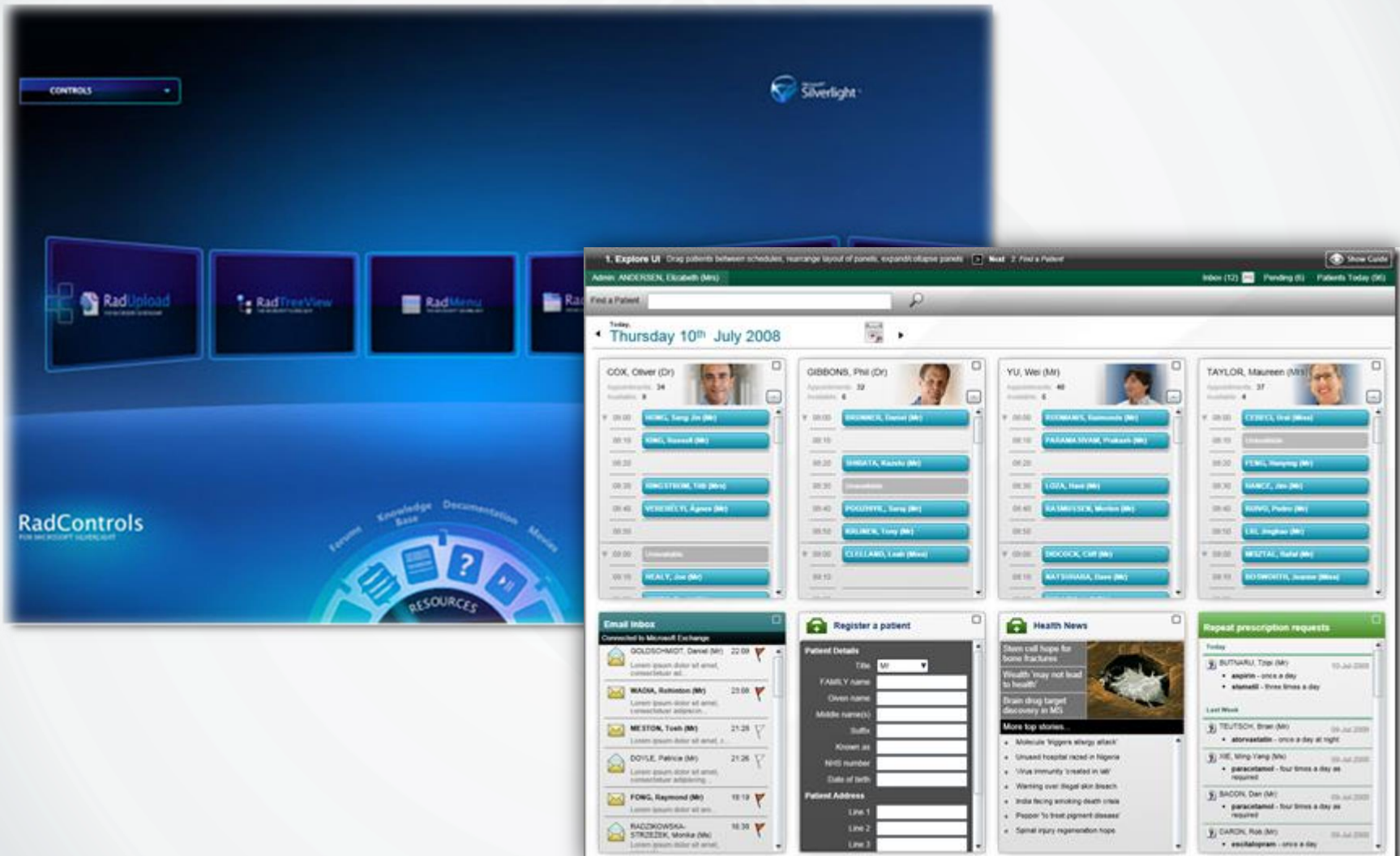


# WPF



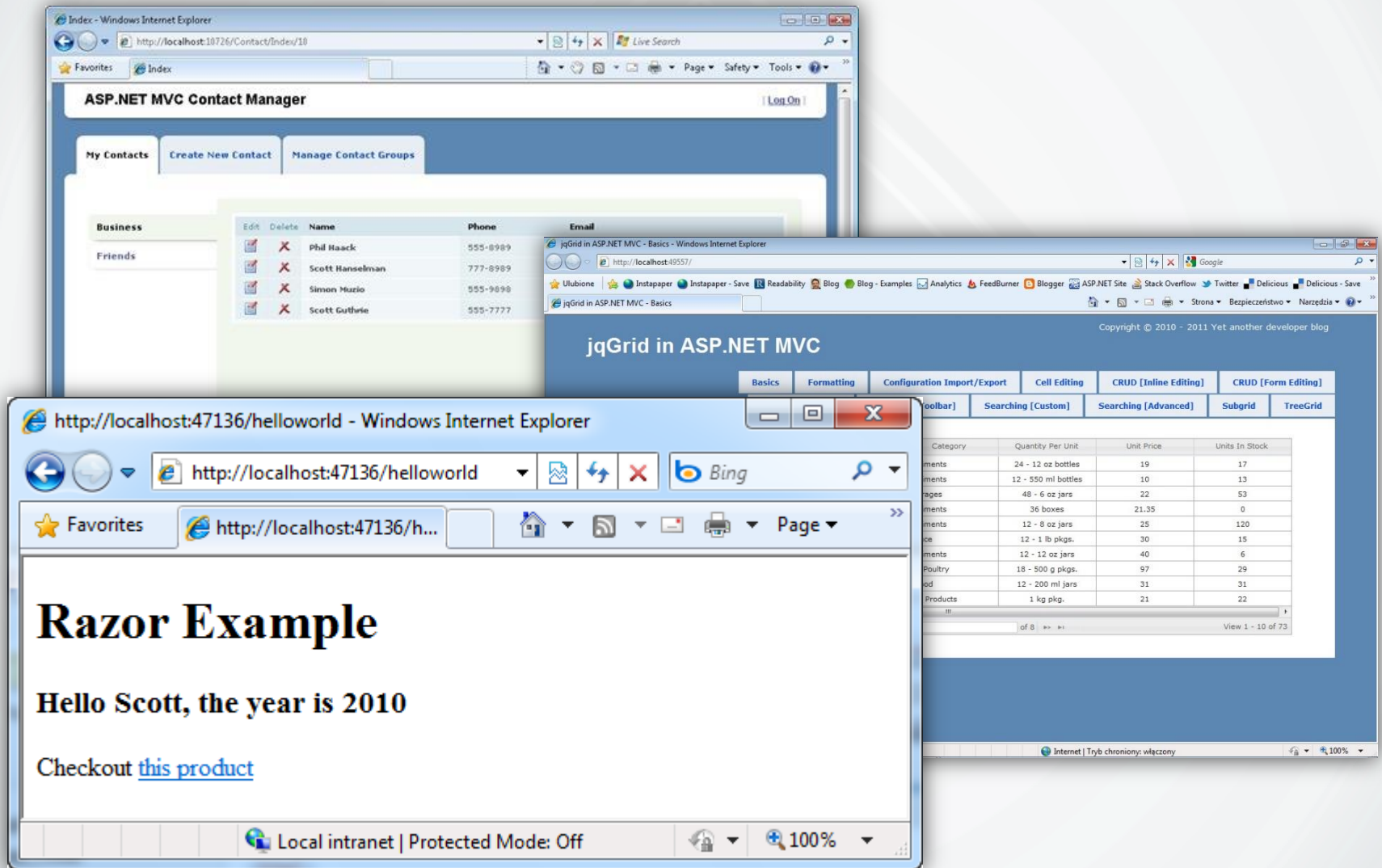


# SILVERLIGHT



Chuyên đề ngôn ngữ lập trình 1 - C#

# ASP.NET



# WINDOWS PHONE



**SPEEDTEST.NET**

TYPE	TIME	DOWNLOAD	UPLOAD	PING
Wi-Fi	01/07/13 11:34 AM	4.28	0.18	39
Wi-Fi	01/08/13 1:42 PM	6.87	6.53	46
Wi-Fi	12/20/12 2:00 PM	10.01	4.78	37
Wi-Fi	12/20/12 2:13 PM	11.44	5.01	48
Wi-Fi	12/20/12 2:12 PM	11.53	5.86	49
Wi-Fi	01/08/13 1:32 PM	11.71	7.80	44
Wi-Fi	12/26/12 12:32 PM	12.84	6.22	49

The interface displays a table of test results. The columns are labeled TYPE, TIME, DOWNLOAD, UPLOAD, and PING. Each row represents a single test, with the download speed highlighted in green. At the bottom, there is a trash icon and a menu icon (three dots).

**Settings**

Set a preferred server:  
Chicago, IL [arrow icon]

Display speeds in:  
Mbps

Internal IP:  
10.0.1.92

External IP:

Use my location:  
On [checkbox icon]

The settings screen allows users to configure the app. It includes options to set a preferred server (currently 'Chicago, IL'), choose the unit for displaying speeds (currently 'Mbps'), view the internal IP address (10.0.1.92), and toggle the 'Use my location' feature (currently 'On').

# WINDOWS UNIVERSAL APP

You can *tailor* the design to each device

User Interface

App Model

APIs

Tools

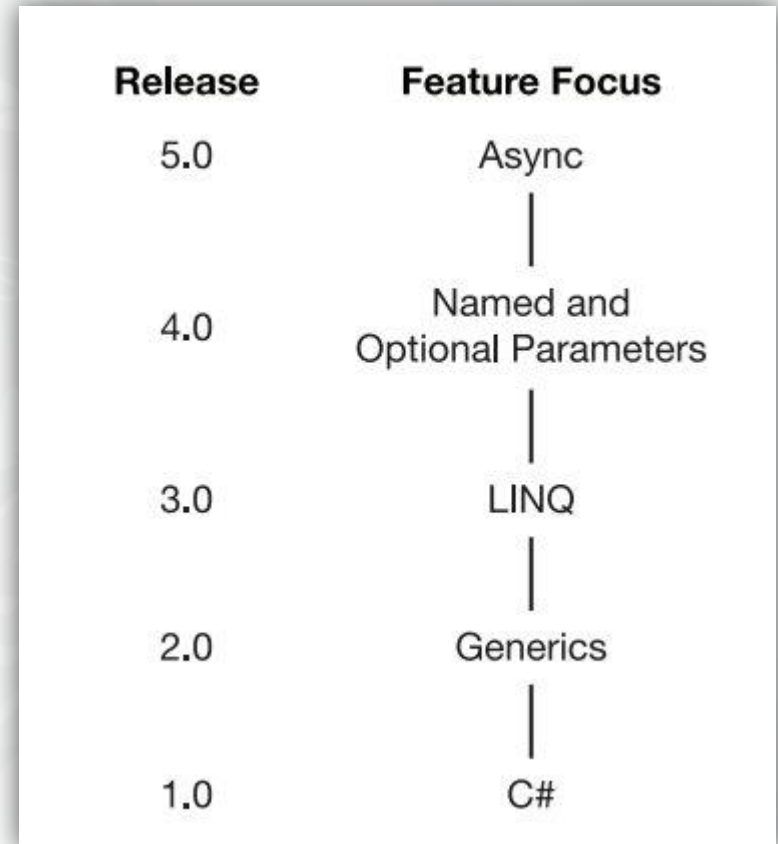
Store





# NGÔN NGỮ LẬP TRÌNH C#

- **C#** là một ngôn ngữ lập trình *thuần hướng đối tượng* được thiết kế để xây dựng các ứng dụng chạy dưới nền tảng **.NET Framework**.
- Phiên bản hiện tại là **C# 5.0**. Mỗi lần xuất bản ra một phiên bản mới đều tập trung vào một số những tính năng mới.



# CHƯƠNG TRÌNH C# ĐẦU TIÊN

- Soạn thảo chương trình **Hello.cs** sau:

```
using System;
using System.Collections.Generic;
using System.Text;

namespace CH01
{
    class Program
    {
        static void Main(string[] args)
        {
            System.Console.WriteLine("Hello World !!");
        }
    }
}
```

- Biên dịch chương trình:

`csc <Tên chương trình>`

- Sau khi biên dịch ta được 1 tập tin thực thi (exe), thực thi tập tin này

# Phân tích chương trình đầu tiên

- Tất cả các ứng dụng C# đều sử dụng một phương thức **main**

```
public static void main(string args[])
```

- Từ khoá **public** là một chỉ định truy xuất. Nó cho biết thành viên của lớp có thể được truy xuất từ bất cứ đâu trong chương trình.
- Từ khoá **static** cho phép main được gọi tới mà không cần tạo ra một thể hiện (instance) của lớp. Nó không phụ thuộc vào các thể hiện của lớp được tạo ra.
- Từ khoá **void** thông báo cho máy tính biết rằng phương thức sẽ không trả lại bất cứ giá trị nào khi thực thi chương trình.
- string args[]** là tham số dùng trong phương thức main. Khi không có một thông tin nào được chuyển vào main, phương thức được thực hiện với các dữ liệu rỗng – không có gì trong dấu ngoặc đơn.

# CÂU HỎI?