



TRƯỜNG ĐẠI HỌC THỦY LỢI
Khoa CNTT – Bộ môn CNPM

LẬP TRÌNH NÂNG CAO



Giảng viên: Lý Anh Tuấn
Email: luanla@wru.vn



Tổng quan về .Net Framework và ngôn ngữ lập trình C#



Tổng quan về .Net Framework

- Được phát triển bởi Microsoft
- Là một nền tảng lập trình và thực thi ứng dụng chủ yếu trên hệ điều hành Microsoft Windows
- Bao gồm môi trường Common Language Runtime (CLR) và tập các thư viện hỗ trợ lập trình .Net Framework Class Library





Common Language Runtime (CLR)

CLR là một máy ảo, cung cấp các dịch vụ:

- An ninh phần mềm (*security*)
- Quản lý bộ nhớ (*memory management*)
- Xử lý lỗi ngoại lệ (*exception handling*).





.Net Framework Class Library

Là những thư viện hỗ trợ việc xây dựng các chương trình phần mềm như:

- Lập trình giao diện
- Truy cập, kết nối cơ sở dữ liệu
- Ứng dụng web
- Các giải thuật, cấu trúc dữ liệu
- Giao tiếp mạng
- ...





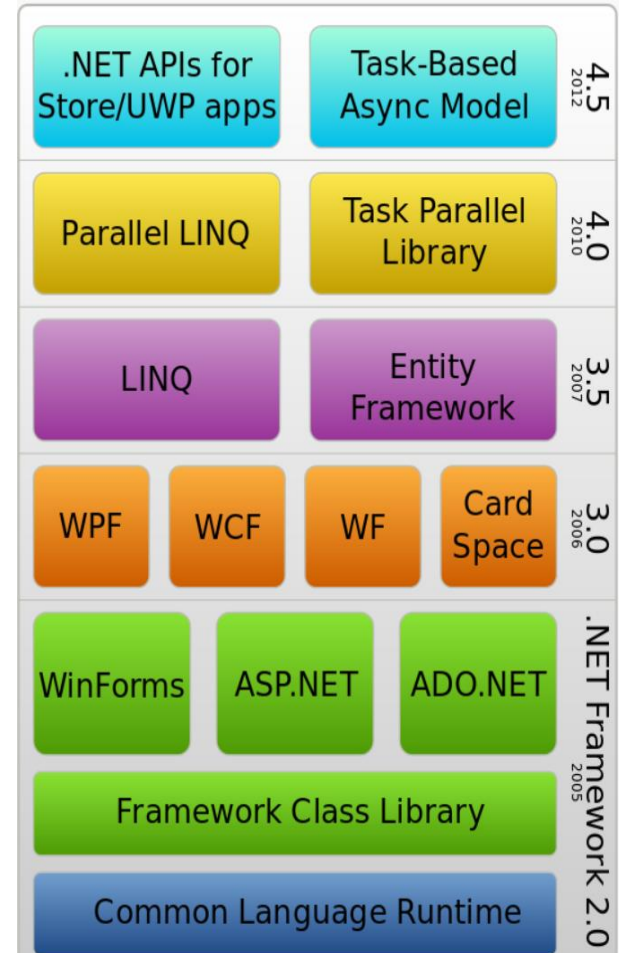
Một số thư viện nền tảng

Namespace	Description
System	Chứa các lớp cơ bản
System.IO	Chứa các lớp cho thao tác Input và Output
System.Net	Chứa các lớp liên quan đến network protocol
System.Collections	Chứa các lớp liên quan đến xử lý tập hợp
System.Data	Chứa các lớp của ADO.NET
System.Drawing	Chứa các lớp thực thi chức năng GUI
System.Threading	Chứa các lớp lập trình MultiThread
System.Web	Chứa các lớp liên quan đến HTTP protocol
System.Xml	Chứa các lớp liên quan XML



.Net Framework

- Qua nhiều giai đoạn phát triển, đến nay .Net Framework đã tích hợp rất nhiều thành phần thiết kế sẵn giúp cho việc lập trình được nhanh hơn, đơn giản hơn.
- Hỗ trợ đa ngôn ngữ: C++.Net, VB.Net, Jscip.Net, F#, C#.





Giới thiệu về ngôn ngữ lập trình C#

- C# là ngôn ngữ lập trình đơn giản:
 - C# khá giống C / C++ về diện mạo, cú pháp, biểu thức, toán tử.
 - Các chức năng của C# được lấy trực tiếp từ ngôn ngữ C / C++ nhưng được cải tiến để làm cho ngôn ngữ đơn giản hơn.





Giới thiệu về ngôn ngữ lập trình C#

- C# là ngôn ngữ hiện đại, có những tính năng:
 - Xử lý ngoại lệ
 - Thu gom bộ nhớ tự động
 - Có những kiểu dữ liệu mở rộng
 - Bảo mật mã nguồn





Giới thiệu về ngôn ngữ lập trình C#

- C# là ngôn ngữ hướng đối tượng với những đặc tính:
 - Sự đóng gói (encapsulation)
 - Sự kế thừa (inheritance)
 - Tính đa hình (polymorphism)





Giới thiệu về ngôn ngữ lập trình C#

- C# là ngôn ngữ mạnh mẽ và mềm dẻo khi được dùng để tạo ra các ứng dụng:
 - Xử lý văn bản
 - Xử lý đồ họa
 - Xử lý bảng tính
 - Thậm chí tạo ra những trình biên dịch cho các ngôn ngữ khác.





Giới thiệu về ngôn ngữ lập trình C#

- C# là một ngôn ngữ lập trình trực quan
- Là một trong những ngôn ngữ được phát triển ở nền tảng .Net
- Để sử dụng được C# cần cài đặt .Net Framework
- Để sử dụng được các tính năng lập trình trực quan cần cài đặt Microsoft Visual Studio





Một số dạng ứng dụng của C#

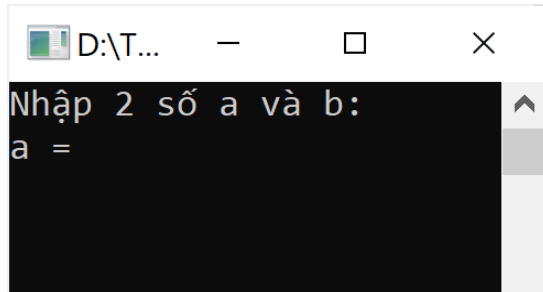
- Chương trình Console
 - Giao tiếp với người dùng bằng bàn phím
 - Chỉ sử dụng cửa sổ dòng lệnh, không có giao diện đồ họa
- Chương trình winform
 - Giao tiếp với người dùng bằng bàn phím và chuột
 - Có giao diện đồ họa và xử lý sự kiện
- Chương trình webform
 - Kết hợp với ASP.NET, C# đóng vai trò xử lý ngầm
 - Có giao diện đồ họa và xử lý sự kiện



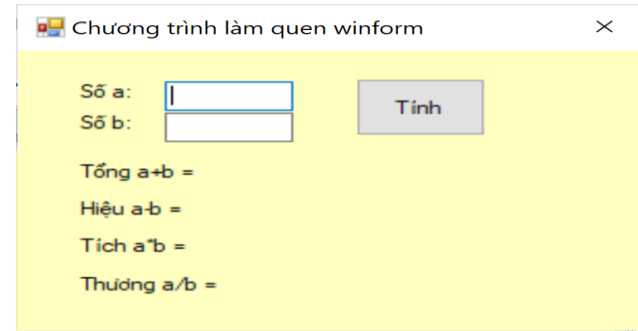


Giao diện các loại ứng dụng trong C#

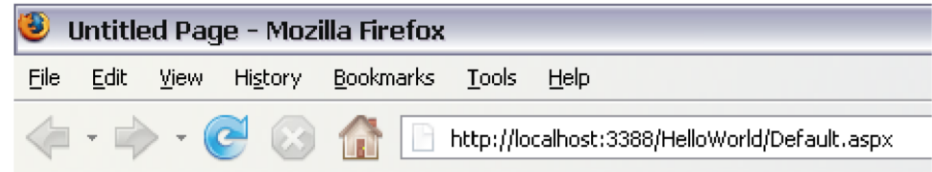
- Chương trình Console:



- Chương trình winform:



- Chương trình webform:



Hello World!



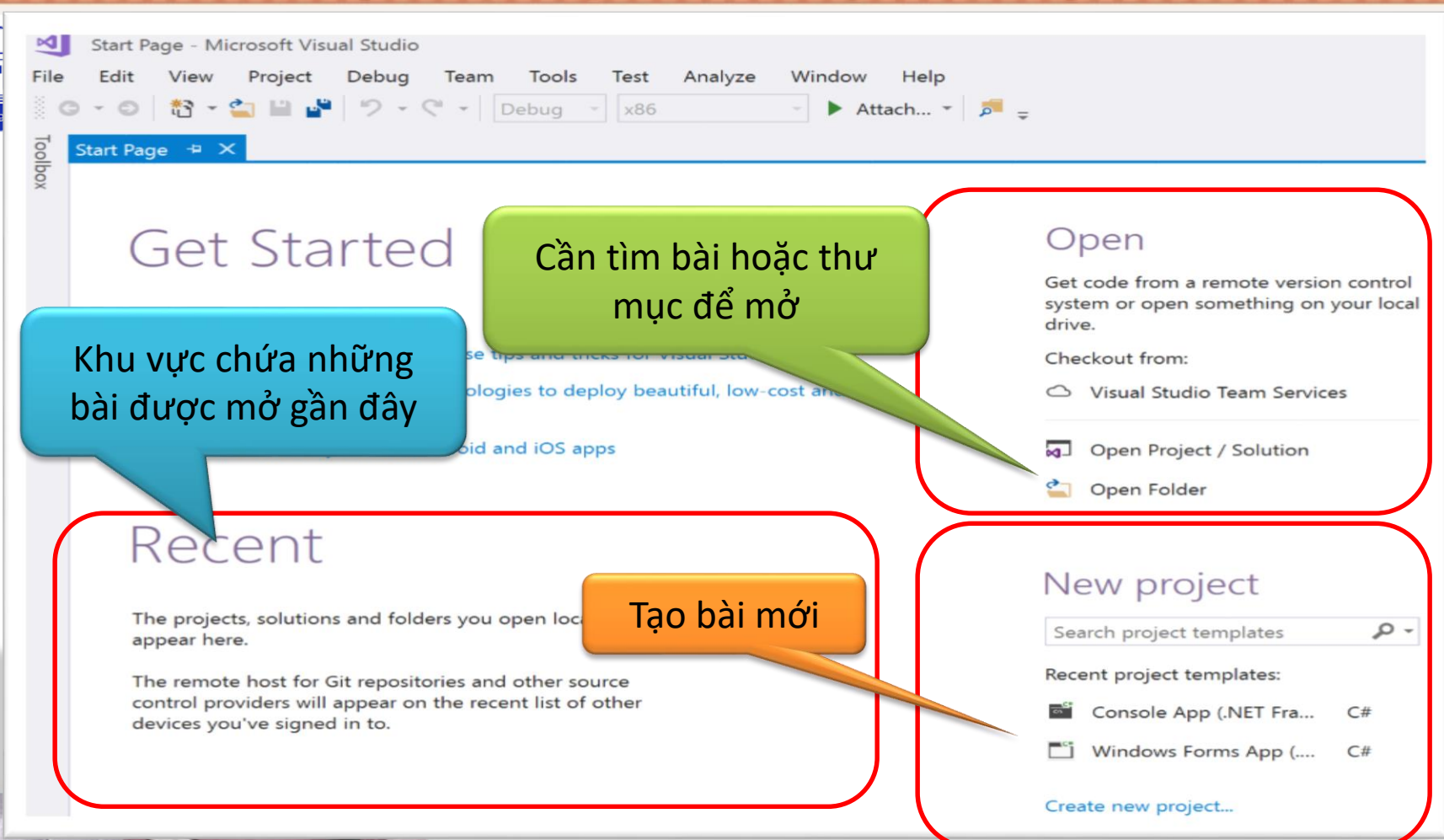


Cách tạo một chương trình Console trong C#



Khởi động phần mềm Visual Studio





Tạo mới một chương trình

The image shows the 'New Project' dialog in Visual Studio. The 'Visual C#' category is selected in the left sidebar. In the main list, 'Windows Forms App (.NET Framework)' and 'Console App (.NET Framework)' are highlighted with red boxes. Below the list, the 'Name' and 'Solution name' fields are both set to 'ConsoleApp1'. The 'Location' is 'D:\'. The 'Framework' is '.NET Framework 4.6.1'. The 'Browse...' button is highlighted, and the 'Create directory for solution' checkbox is checked. The 'OK' button is highlighted at the bottom right.

Chọn Windows Form App để tạo một chương trình chạy bằng giao diện windows

Lựa chọn Visual C# để đảm bảo chương trình được viết bằng C#

Lựa chọn Console App để tạo một chương trình chạy bằng dòng lệnh

Đặt lại tên cho project của mình (1 project là 1 bài toán nhỏ)

Check vào ô này để hệ thống tự tạo thư mục mới để lưu trữ dự án

Nhớ bấm nút Browse để lựa chọn nơi lưu trữ dự án

Đặt lại tên cho solution (1 solution là 1 chương trình lớn gồm 1 hoặc nhiều bài toán nhỏ)

Cuối cùng bấm nút OK để hoàn tất



Cấu trúc thư mục solution

(D:) > MySolution

Tên thư mục chứa solution

Name

Mỗi project được lưu trong một thư mục con

MyApp

NhapXuat

Tên file solution

MySolution.sln

Mỗi thư mục project con sẽ chứa những file mã chương trình có phần mở rộng là **.cs**

bin

obj

Properties

NhapXuat.csproj

Program.cs



Làm việc với một chương trình Console trong C#

The screenshot displays the Visual Studio IDE with the following components:

- Code Editor:** Shows the `Program.cs` file with the following code:

```
1 using System;  
2 using System.Collections.Generic;  
3 using System.Linq;  
4 using System.Text;  
5 using System.Threading.Tasks;  
6  
7 namespace Nhapxuat  
8 {  
9     class Program  
10    {  
11        static void Main(string[] args)  
12        {  
13        }  
14    }  
15 }
```
- Solution Explorer:** Located on the right, it shows the project structure:
 - Solution 'BTChuong1' (1 project)
 - Nhapxuat
 - Properties
 - References
 - App.config
 - Program.cs

- Callouts:**
- A green callout points to the Solution Explorer, stating: "Một Solution có thể có nhiều Project."
- A blue callout points to the `Program` class in the code editor, stating: "Mỗi Project có thể có nhiều Class"
- An orange callout points to the Solution Explorer, stating: "Cửa sổ Solution Explorer cho biết thông tin về Solution"



Làm việc với một chương trình Console trong C#

Sau khi viết xong chương trình, ấn vào nút Start để thực hiện dịch và chạy chương trình

Mỗi Project được tạo một không gian tên riêng.

```
using System.Threading.Tasks;

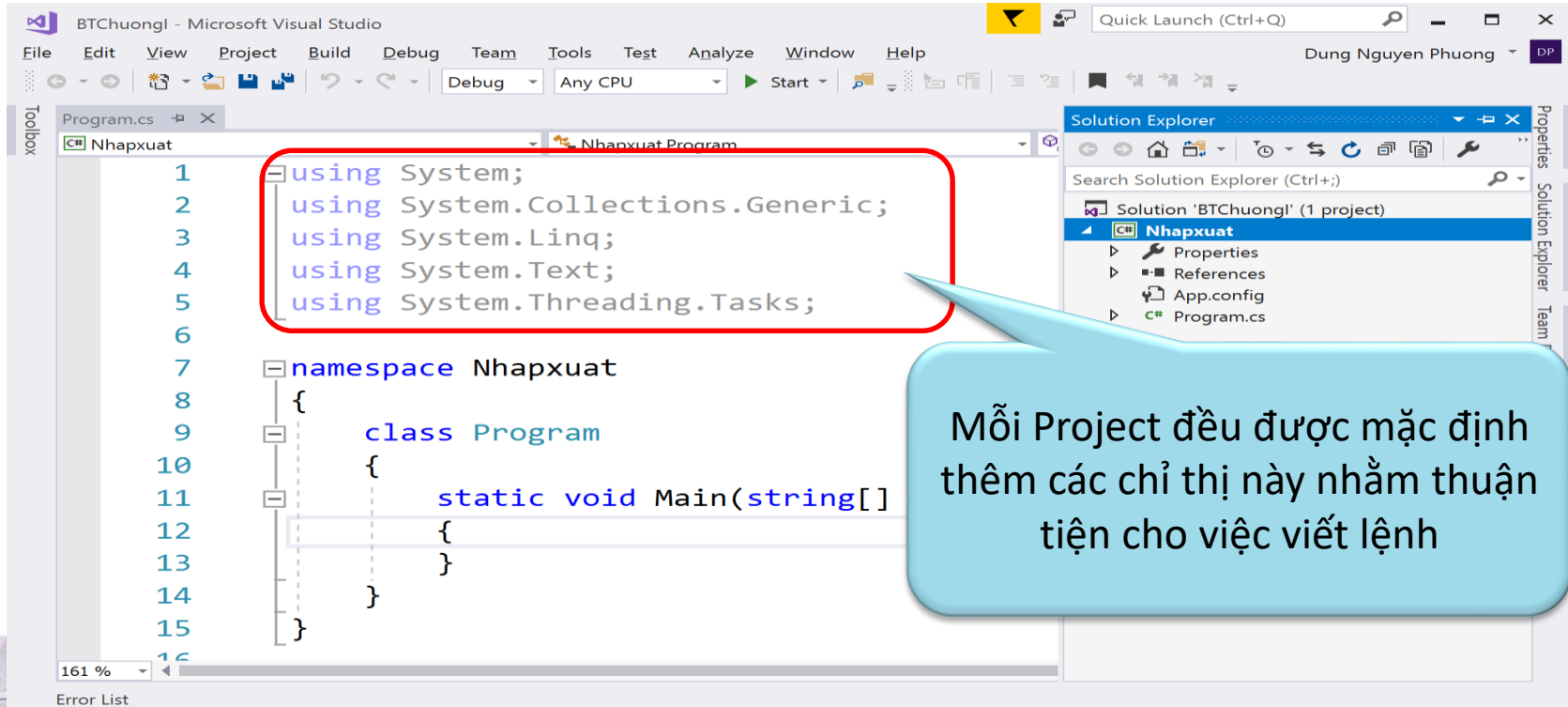
namespace Nhapxuat
{
    class Program
    {
        static void Main(string[] args)
        {
        }
    }
}
```

Mỗi Project có một hàm Main() riêng. Các lệnh sẽ được viết vào trong hàm Main()

Ready Ln 12 Col 10 Ch 10



Làm việc với một chương trình Console trong C#





Lệnh nhập/xuất trong C#

- Việc nhập, xuất dữ liệu ra màn hình Console trong C# sử dụng lệnh `ReadLine()`, `WriteLine()`
- 2 lệnh này thuộc lớp `Console` trong namespace `System`
- Do đó ở đầu chương trình sử dụng chỉ thị `using System` thì trong chương trình chỉ cần viết `Console.WriteLine()` mà không cần viết `System.Console.WriteLine()`;





Lệnh nhập/xuất trong C#

- VD:

```
namespace Nhapxuat
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello, what your name?");
            string name = Console.ReadLine();
            Console.WriteLine("Hello " + name);
            Console.ReadLine();
        }
    }
}
```





Nhập dữ liệu vào một biến

- Chú ý:
 - Lệnh `ReadLine()` dùng để nhập một dòng dữ liệu
 - Muốn nhập dữ liệu kiểu số **cần chuyển đổi kiểu** dữ liệu cho dòng dữ liệu nhập vào
 - Lệnh chuyển kiểu: là các lệnh thuộc lớp `Convert` trong `namespace System`
 - VD: `int a;`
`a = Convert.ToInt32(Console.ReadLine());`





Các kiến thức đã học trong bài

- Tổng quan về .Net Framework
- Tổng quan về ngôn ngữ lập trình C#
- Cách tạo chương trình Console trong C#

