

LẬP TRÌNH TRONG MÔI TRƯỜNG NET



NHÀ XUẤT GIÁO DỤC - 2010

Visual studio.NET – Phần 1

Visual studio.NET là một môi trường tích hợp triển khai phần mềm(*Intergrated Development Environmet, IDE*). Nó được thiết kế để lập ra một tiến trình viết mã, gỡ rối, và biên dịch thành một assembly một cách dễ dàng. Visual studio.NET cho bạn một ứng dụng *multiple-document-interface* rất tinh vi, trong đó bạn có thể liên kết mọi thứ để phát triển đoạn mã của bạn. Nó bao gồm:

Một *Text Editor* : trong đó bạn có thể viết đoạn mã C#. *Text e ditor* này thì hơi phức tạp, và rất rành cú pháp C#. Tức là, khi bạn gõ các câu lệnh vào, nó sẽ tự động bố trí đoạn của bạn, ví dụ như bằng cách thụt canh cột các dòng lệnh, cho khớp cặp dấu {}, và tô màu những từ khoá. Ngoài ra, nó sẽ thực hiện kiểm tra vài cú pháp khi bạn gõ và sẽ gạch dưới những dòng mã bị sai. Nó còn có thêm một chức năng đặc biệt là *Intelliense*, nó sẽ tự động hiển thị

tên của các lớp, trường hay phương thức khi bạn bắt đầu gõ chúng. Khi bạn bắt đầu đánh các tham số cho phương thức, nó sẽ hiển thị danh sách tham số. Màn hình bên dưới sẽ chỉ đặc trưng này với một lớp cơ sở .NET là *ListBox* :

Một Design view editor, nó cho phép bạn đặt giao diện người dùng và các control dữ liệu truy cập trong dự án của bạn. Khi bạn làm như vậy, Visual studio.NET sẽ tự động thêm những mã C# cần thiết cho tập tin nguồn của bạn để tạo những control này trong dự án của bạn.

Các cửa sổ hỗ trợ cho phép bạn xem và sửa đổi những khía cạnh khác nhau trên dự án của bạn. Ví dụ có những cửa sổ cho bạn thấy những lớp hình thành đoạn mã nguồn cũng như các thuộc tính trên các lớp Window Form hoặc Web Form. Bạn cũng có thể sử dụng những cửa sổ này để khai báo các tùy chọn biên dịch.

Biên dịch trong lòng môi trường: Để thay cho việc chạy trình biên dịch C# từ dòng lệnh, bạn có thể chọn một tùy chọn menu để biên dịch và Visual Studio.NET sẽ gọi trình biên dịch cho bạn. Nó cũng có thể chạy một chương trình khả thi đã được biên dịch, như vậy bạn có thể biết chương trình chạy tốt hay không, và bạn có thể chọn giữa hai cấu hình xây dựng chương trình khác nhau : *debug build* hoặc *release build*.

Một Intergate Debugger hỗ trợ việc gỡ rối xuyên ngôn ngữ trong khuôn viên IDE. Ngoài ra bạn có thể gỡ rối trong một lúc nhiều chương trình. Bạn có thể chỉnh sửa đoạn mã ngay trong Text editor Visual studio.NET để sửa chữa bug, rồi cho biên dịch lại và cho chạy lại chương trình đã được sửa chữa ngay tại chỗ bỏ lỡ vì lỗi.

Intergated MSDN help Visual studio.NET có thể gọi tài liệu MSDN cho bạn. Ví dụ như khi bạn không biết ý nghĩa của một từ khoá thì bạn chọn nó và nhấn F1 thì nó sẽ gọi MSDN lên để giải thích từ đó cho bạn.

Truy cập đến một chương trình khác: Nếu tất cả các tiện ích trên chưa đủ thì Visual studio.NET có thể gọi các tiện ích khác để cho phép bạn kiểm tra và sửa đổi các khía cạnh khác của máy tính bạn hay mạng mà bạn không phải rời khỏi môi trường phát triển. Giữa nhiều công cụ có sẵn, bạn có thể kiểm tra việc chạy các dịch vụ, và sự kết nối dữ liệu , và có một cửa sổ *internet explorer* cho phép bạn lướt Web.

Chắc chắn rằng bạn đã có kinh nghiệm trong C++ hay VB trước khi bạn làm quen với phiên bản Visual studio.NET, do đó bạn biết rằng nhiều chức năng ở trên không mới mẽ. Tuy nhiên những gì mới trong Visual studio.NET là nó liên kết tất cả chức năng trong môi trường phát triển của VS 6. Có nghĩa là những ngôn ngữ gì bạn sử dụng trong VS6, bạn sẽ tìm thấy một vài chức năng mới trong Visual Studio.NET.

Từ bất kỳ nền nào, bạn sẽ tìm thấy tầm nhìn tổng thể của môi trường phát triển đã thay đổi để điều tiết các chức năng mới, những IDE xuyên ngôn ngữ đơn, và sự hợp nhất với .NET. Có nhiều menu tùy chọn và thanh công cụ tùy chọn mới, và nhiều tiện ích khác trong VS6 đã được đổi tên. Vì thế bạn cần bỏ một khoảng thời gian để làm quen với việc trình bày và làm chủ trong Visual studio.NET.

Mục đích chính của Visual studio.NET là bảo đảm bạn làm quen với tất cả các khái niệm về việc xây dựng và gỡ rối trong một ứng dụng C# và có thể sử dụng các chức năng cụ thể, điều mà không thể làm được trong môi trường phát triển của bất kỳ ngôn ngữ nào trước.

Màn hình sau sẽ thể hiện màn hình của Visual studio.NET khi bạn viết mã:

Tạo một dự án:

Khi bạn cài đặt Visual studio.NET, bạn sẽ muốn bắt đầu dự án đầu tiên. Với Visual studio.NET, ít khi bạn bắt đầu với một tập tin trắng mà bạn sẽ gõ vào đoạn mã C# từ điểm bắt đầu. Thay vào đó bạn sẽ cho Visual studio.NET biết loại dự án mà bạn muốn tạo, sau đó nó sẽ tự động tạo ra một đoạn mã C# để tạo thành một sườn outline cho dự án đó. Và bạn chỉ có việc là thêm mã của bạn vào sườn đó. Ví dụ, bạn muốn viết một ứng dụng trên Window GUI, thì Visual studio.NET sẽ bắt đầu tạo một tập tin chứa đoạn mã nguồn C# cho phép tạo một form cơ bản. Form này có khả năng giao tiếp với Window, và tiếp nhận những tình huống. Nó cho phép được thu nhỏ hoặc phóng to hoặc thay đổi kích thước ... Và lúc này bạn sẽ thêm các chức năng bạn mong muốn có đối với biểu mẫu. Nếu ứng dụng của bạn được dùng theo kiểu dòng lệnh thì visual studio.NET sẽ cho bạn một namespace cơ bản, lớp,

và phương thức Main() để bắt đầu. Dĩ nhiên là nếu bạn muốn bắt đầu từ số không, thì Visual studio.NET cũng cho bạn chọn một ứng dụng rỗng.

Khi bạn tạo ra một dự án, nó sẽ dàn dựng những tùy chọn biên dịch cần thiết để cung cấp cho trình biên dịch C# có hay không để biên dịch một ứng dụng dòng lệnh, một thư viện, hay một ứng dụng Window. Ngoài ra, những thư viện lớp cơ sở nào bạn sẽ cần. Tuy nhiên bạn có thể sửa đổi những cài đặt này nếu bạn cần.

Khi bạn bắt đầu với visual studio.NET lần đầu tiên, bạn sẽ được thấy một **start page**. Đó là một trang HTML chứa các liên kết khác nhau để dẫn bạn đến với các trang web hữu ích, cho phép bạn tạo dáng và cấu hình của Visual studio.NET, hoặc mở những dự án hiện hữu hoặc khởi động dự án mới . Trên giữa màn hình start page, bạn thấy liệt kê một số dự án mà bạn đã làm việc gần đây nhất:

Chọn một kiểu dự án:

Bạn có thể tạo một dự án mới bằng việc click trên các liên kết thích hợp trên trang bắt đầu, hay click trên menu file chọn New/project. Hay bạn có thể chọn các mẫu trong hộp dialog New project.

Các hộp Dialog này sẽ hỏi bạn muốn loại loại mã nguồn nào cần được tạo ra, và những tùy chọn biên dịch nào bạn muốn. Những trình biên dịch nào bạn muốn biên dịch mã của bạn: C#, VB.NET hay C++.

Tuy nhiên, trong sách này chỉ đưa ra những tùy chọn có thể ứng dụng trong dự án C#

Nếu bạn chọn ..	Bạn sẽ lấy những mã C# và tùy chọn biên dịch được tạo ra
Windows Application	Một mẫu trống cơ bản
Class Library	Một lớp .NET có thể được gọi bởi các mã khác
Windows Control Library	Một lớp .NET có thể được gọi bởi mã khác và có một giao diện người dùng
ASP.NET Web Application	Một ASP.NET site web cơ sở : trang ASP.NET và lớp C# tạo ra HTML gửi đến trình duyệt từ các trang này

Nếu bạn chọn ..	Bạn sẽ lấy những mã C# và tùy chọn biên dịch được tạo ra
ASP.NET Web Service	Một lớp C# hành động như một Web service trọn vẹn
Web Control Library	Một control có thể được gọi lên bởi trang ASP.NET, để tạo ra mã HTML đưa bề mặt của một điều khiển khi thể hiện trên trình duyệt.
Console Application	Một ứng dụng chạy tại chế độ dòng lệnh hay một cửa sổ console
Windows Service	Một dịch vụ chạy trên nền Windows NT và Windows 2000
Empty Project	Không có gì hết. Bạn phải viết tất cả
Empty Web Project	Như empty project nhưng các cài đặt biên dịch được cài vào cấu trúc trình biên dịch để tạo mã cho trang AS.NET.
New Project In Existing Folder	Những tập tin dự án mới cho một empty project

Các dự án console mới:

Khi ta click OK ở hộp dialog trên để tạo một ứng dụng console. Visual studio.NET sẽ cung cấp một số tập tin, bao gồm một tập tin mã nguồn, *class1.cs* chứa mã sường khởi tạo.

Hình dưới sẽ chỉ cho ta thấy những mã mà visual studio.NET viết cho ta:

Ở đây chúng ta có một chương trình C# và chưa có ý nghĩa thực thi nhưng nó chứa những yêu cầu cơ bản của một chương trình C# như là: một namespace, một lớp chứa phương thức *main()*. Đoạn mã này sẵn sàng để chạy nếu bạn nhấn f5 hay chọn *debug/start*. Tuy nhiên, trước hết chúng ta phải thêm một dòng lệnh để chương trình bạn làm gì đó.

```
static void Main(string[] args)
{
    //
```