Introduction to Programming

Creating and Running Your First C# Program

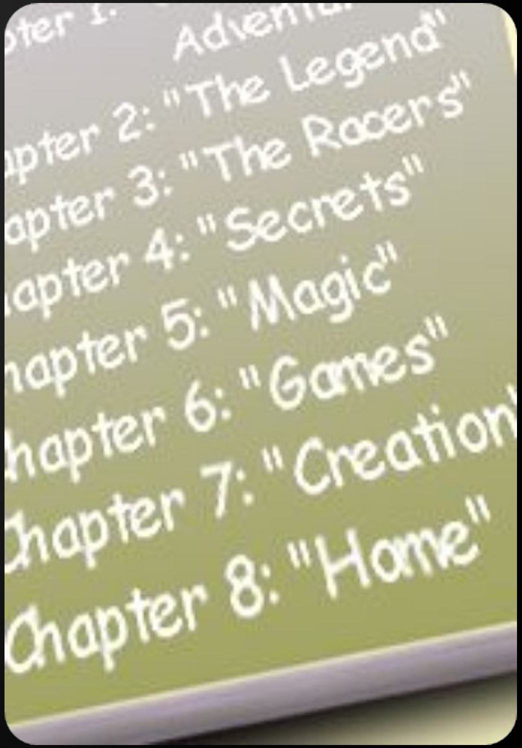
Svetlin Nakov

Telerik Corporation

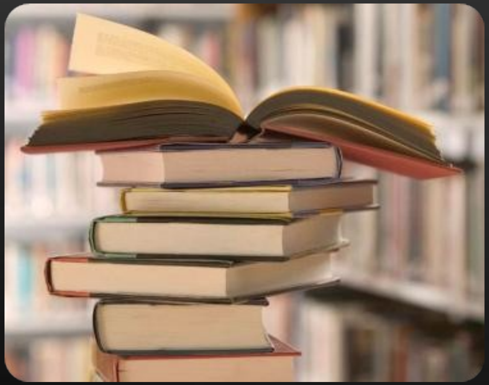
www.telerik.com

Table of Contents

1. What is Computer Programming?
2. Your First C# Program
3. What is .NET Framework?
4. What is Visual Studio?
5. What is MSDN Library?



pter 1: Adventu
pter 2: "The Legend"
apter 3: "The Racers"
apter 4: "Secrets"
apter 5: "Magic"
hapter 6: "Games"
hapter 7: "Creation
Chapter 8: "Home"



What is Computer Programming?

Define: Computer Programming

Computer programming: creating a sequence of instructions to enable the computer to do something

Definition by Google

Programming Phases

- Define a task/problem = Specification
- Plan your solution = Design
  - Find suitable algorithm to solve it
  - Find suitable data structures to use
- Write code = Implementation
- Fix program error (bugs)

  = Testing & Debugging
- Make your customer happy = Deployment

Your First C# Program

Hello World
Microsoft

First Look at C#

Sample C# program:

```
using System;

class HelloCSharp
{
    static void Main()
    {
        Console.WriteLine("Hello, C#");
    }
}
```

C# Code – How It Works?

Include the standard namespace "`System`"

Define a class called "`HelloCSharp`"

```
using System;

class HelloCSharp
{
    static void Main()
    {
        Console.WriteLine("Hello, C#");
    }
}
```

Define the `Main()` method – the program entry point

Print a text on the console by calling the method "`WriteLine`" of the class "`Console`"

C# Code Should Be Well Formatted

Class names should use `PascalCase` and start with a `CAPITAL` letter.

```
using System;

class HelloCSharp
{
    static void Main()
    {
        Console.WriteLine("Hello, C#");
    }
}
```

The { symbol should be alone on a new line.

The } symbol should be under the corresponding {.

The block after the { symbol should be indented by a `TAB`.

Example of Bad Formatting

Such formatting makes the source code unreadable.

```
         using
System
                                                ;

    class         HelloCSharp                   {
           static
void       Main(                  )   {   Console
  .      WriteLine    ("Hello, C#"     )
;Console.
   WriteLine        (               "Hello again"
                    )                         ;}}
```

What is "C#"?

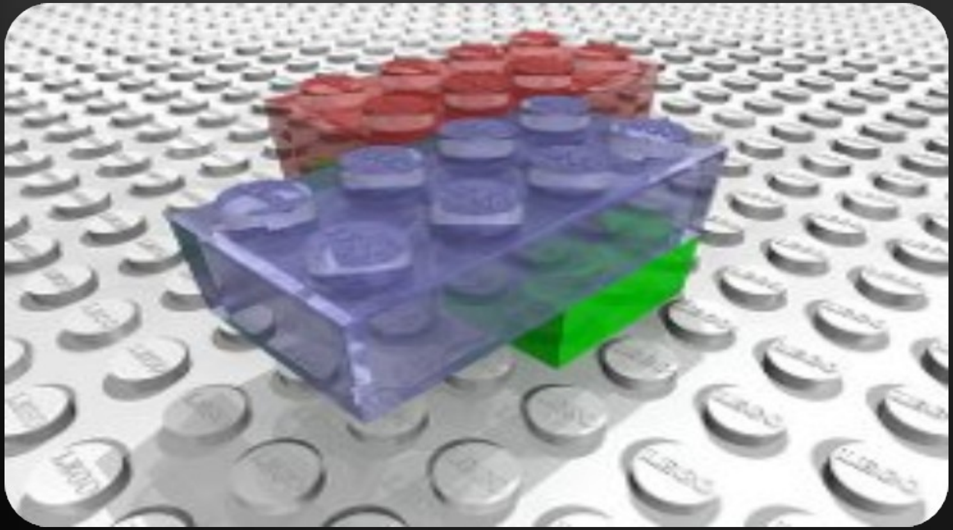
- Programming language
  - A syntax that allow to give instructions to the computer
- C# features:
  - New cutting edge language
  - Extremely powerful
  - Easy to learn
  - Easy to read and understand
  - Object-oriented

What You Need to Program?

- Knowledge of a programming language
  - C#
- Task to solve
- Development environment, compilers, SDK
  - Visual Studio, .NET Framework SDK
- Set of useful standard classes
  - Microsoft .NET Framework FCL
- Help documentation
  - MSDN Library

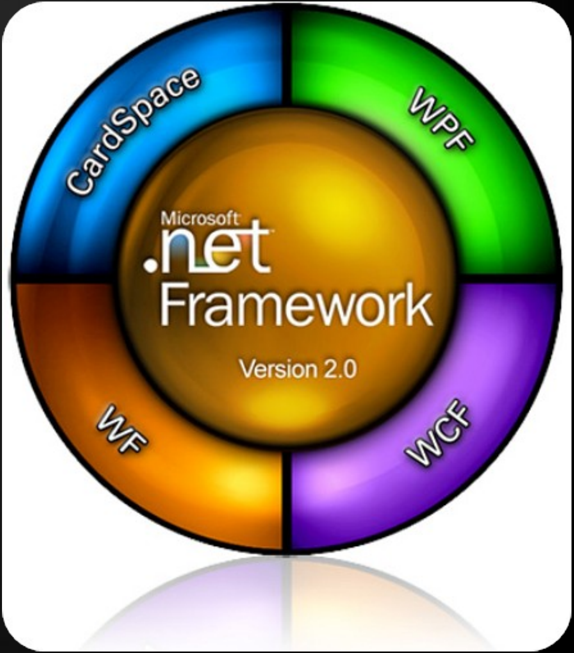
Your First C# Program

Live Demo



#include <iostream.h>
main()
{
cout <<
"Hello World!"
<< endl;
return 0;
}

What is .NET Framework?

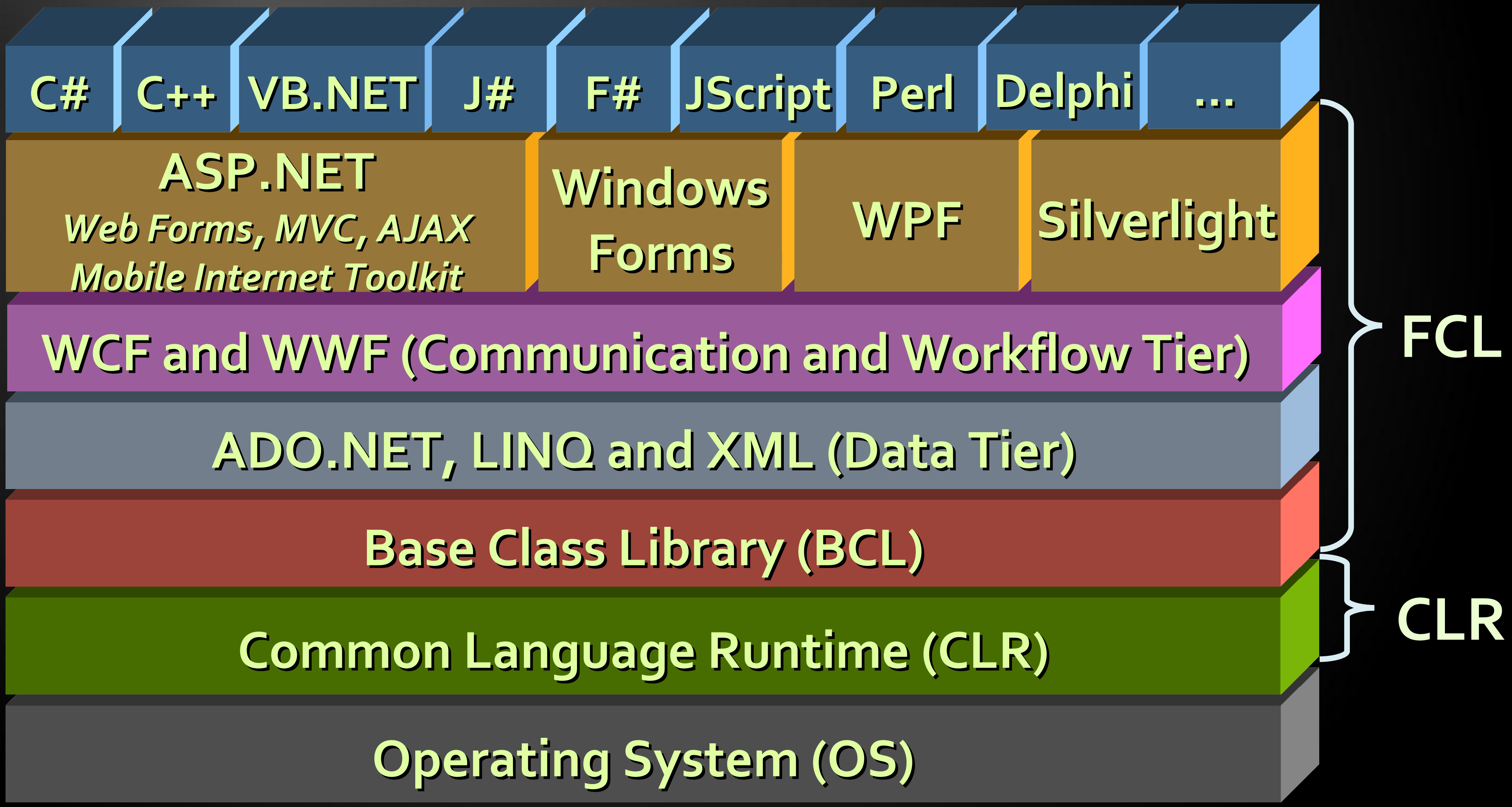

CardSpace
WPF
WF
WCF
Microsoft
.net
Framework
Version 2.0

What is .NET Framework?

- Environment for execution of .NET programs
- Powerful library of classes
- Programming model
- Common execution engine for many programming languages
  - C#
  - Visual Basic .NET
  - Managed C++
  - ... and many others

Inside .NET Framework

- Building blocks of .NET Framework


C#
C++
VB.NET
J#
F#
JScript
Perl
Delphi
…
ASP.NET
Web Forms, MVC, AJAX
Mobile Internet Toolkit
Windows Forms
WPF
Silverlight
WCF and WWF (Communication and Workflow Tier)
ADO.NET, LINQ and XML (Data Tier)
Base Class Library (BCL)
Common Language Runtime (CLR)
Operating System (OS)
FCL
CLR

CLR – The Heart of .NET Framework

- Common Language Runtime (CLR)
  - Managed execution environment
    - Executes .NET applications
    - Controls the execution process
  - Automatic memory management (garbage collection)
  - Programming languages integration
  - Multiple versions support for assemblies
  - Integrated type safety and security

Framework Class Library

- Framework Class Library (FCL)
  - Provides basic functionality to developers:
    - Console applications
    - WPF and Silverlight rich-media applications
    - Windows Forms GUI applications
    - Web applications (dynamic Web sites)
    - Web services, communication and workflow
    - Server & desktop applications
    - Applications for mobile devices

What is Visual Studio?

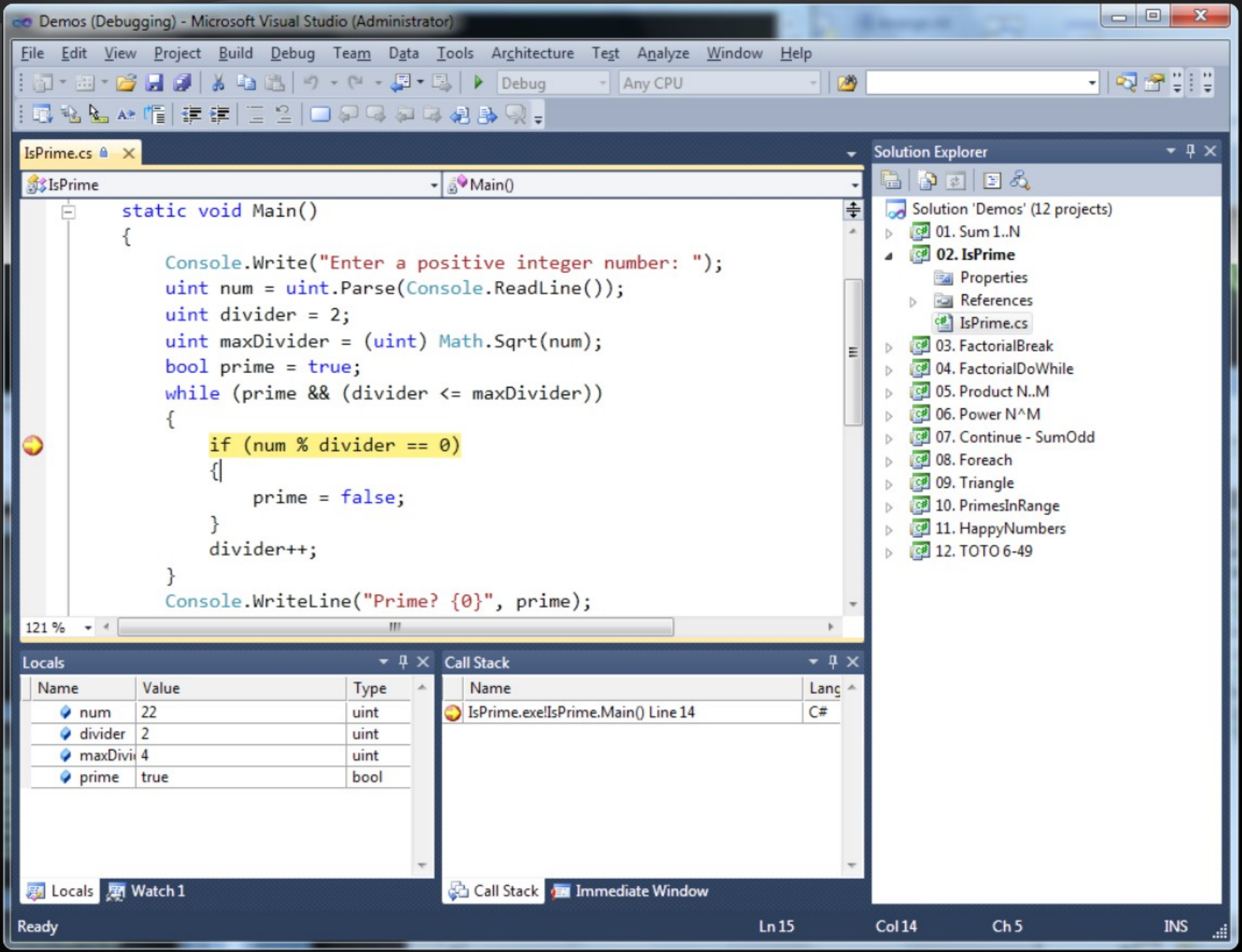
Visual Studio

- Visual Studio – Integrated Development Environment (IDE)
- Development tool that helps us to:
  - Write code
  - Design user interface
  - Compile code
  - Execute / test / debug applications
  - Browse the help
  - Manage project's files

Benefits of Visual Studio

- Single tool for:
  - Writing code in many languages (C#, VB, …)
  - Using different technologies (Web, WPF, …)
  - For different platforms (.NET CF, Silverlight, …)
- Full integration of most development activities (coding, compiling, testing, debugging, deployment, version control, …)
- Very easy to use!

Visual Studio – Example


Demos (Debugging) - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug Team Data Tools Architecture Test Analyze Window Help
Debug
Any CPU
IsPrime.cs
IsPrime
Main()
121 %
Locals
Watch 1
Call Stack
Immediate Window
Ready
Ln 15
Col 14
Ch 5
INS

```
static void Main()
{
    Console.Write("Enter a positive integer number: ");
    uint num = uint.Parse(Console.ReadLine());
    uint divider = 2;
    uint maxDivider = (uint) Math.Sqrt(num);
    bool prime = true;
    while (prime && (divider <= maxDivider))
    {
        if (num % divider == 0)
        {
            prime = false;
        }
        divider++;
    }
    Console.WriteLine("Prime? {0}", prime);
```

Locals

Name	Value	Type
num	22	uint
divider	2	uint
maxDivi	4	uint
prime	true	bool

Call Stack

Name	Lang
IsPrime.exe!IsPrime.Main() Line 14	C#

Solution Explorer

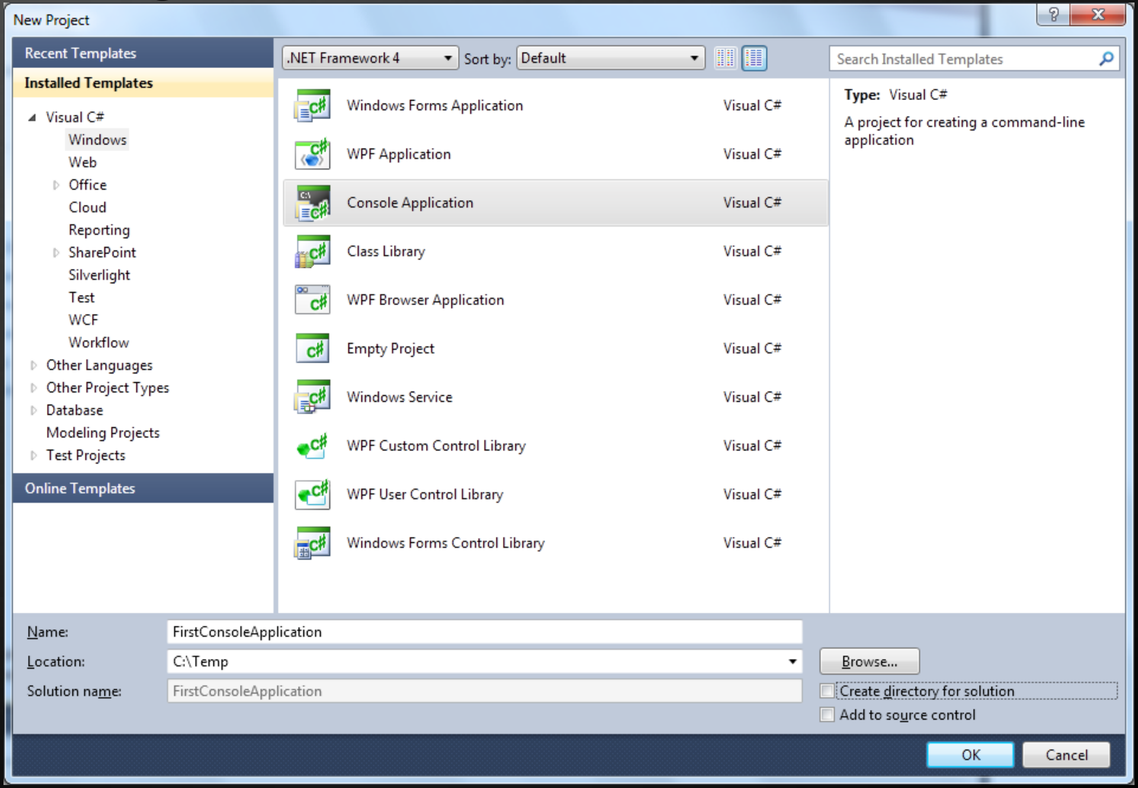
- Solution 'Demos' (12 projects)
  - 01. Sum 1..N
  - **02. IsPrime**
    - Properties
    - References
    - IsPrime.cs
  - 03. FactorialBreak
  - 04. FactorialDoWhile
  - 05. Product N..M
  - 06. Power N^M
  - 07. Continue - SumOdd
  - 08. Foreach
  - 09. Triangle
  - 10. PrimesInRange
  - 11. HappyNumbers
  - 12. TOTO 6-49

Visual Studio

Compiling, Running and Debugging C# Programs

Creating New Console Application

- File → New → Project ...
- Choose C# console application
- Choose project directory and name



New Project

Windows Forms Application	Visual C#
WPF Application	Visual C#
Console Application	Visual C#
Class Library	Visual C#
WPF Browser Application	Visual C#
Empty Project	Visual C#
Windows Service	Visual C#
WPF Custom Control Library	Visual C#
WPF User Control Library	Visual C#
Windows Forms Control Library	Visual C#

Type: Visual C#

A project for creating a command-line application

Name: FirstConsoleApplication

Location: C:\Temp

Solution name: FirstConsoleApplication

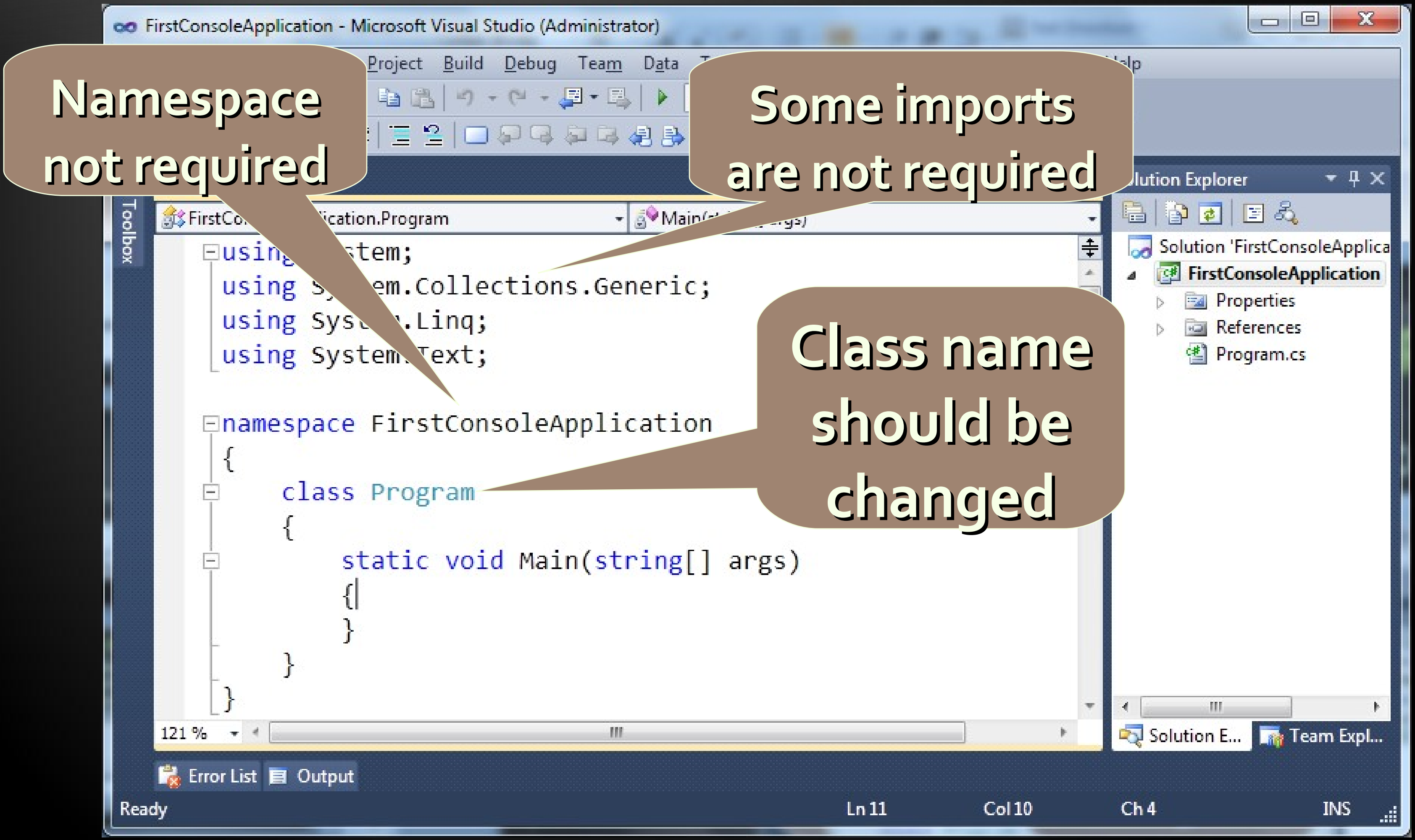
☐ Create directory for solution

☐ Add to source control

Recent Templates
Installed Templates
Visual C#
Windows
Web
Office
Cloud
Reporting
SharePoint
Silverlight
Test
WCF
Workflow
Other Languages
Other Project Types
Database
Modeling Projects
Test Projects
Online Templates
.NET Framework 4
Sort by:
Default
Search Installed Templates
Browse...
OK
Cancel

Creating New Console Application (2)

1. Visual Studio creates some source code for you


FirstConsoleApplication - Microsoft Visual Studio (Administrator)
Project
Build
Debug
Team
Data
Toolbox
FirstCo
ication.Program
Main(
Solution Explorer
Solution 'FirstConsoleApplica
FirstConsoleApplication
Properties
References
Program.cs
121 %
Solution E...
Team Expl...
Error List
Output
Ready
Ln 11
Col 10
Ch 4
INS

```
using     stem;
using S     em.Collections.Generic;
using Sys    .Linq;
using System.Text;

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

Namespace not required

Some imports are not required

Class name should be changed

Compiling Source Code

- The process of compiling includes:
  - Syntactic checks
  - Type safety checks
  - Translation of the source code to lower level language (MSIL)
  - Creating of executable files (assemblies)
- You can start compilation by
  - Using Build->Build Solution/Project
  - Pressing [F6] or [Shift+Ctrl+B]

```
0. 0
1. 0 mov    eax_0.4, v2.4
1. 1 mov    #0xFFFFFFFF.4, v1.4
2. 0 jz     v1.4, @5
3. 0 setz   [es.2:v2.4].1, v4.1
3. 1 add    v2.4, #1.4, v2.4
3. 2 sub    v1.4, #1.4, v1.4
3. 3 jz     v4.1, @5
4. 0 goto   @2
5. 0 bnot   v1.4, result.4
6. 0
```

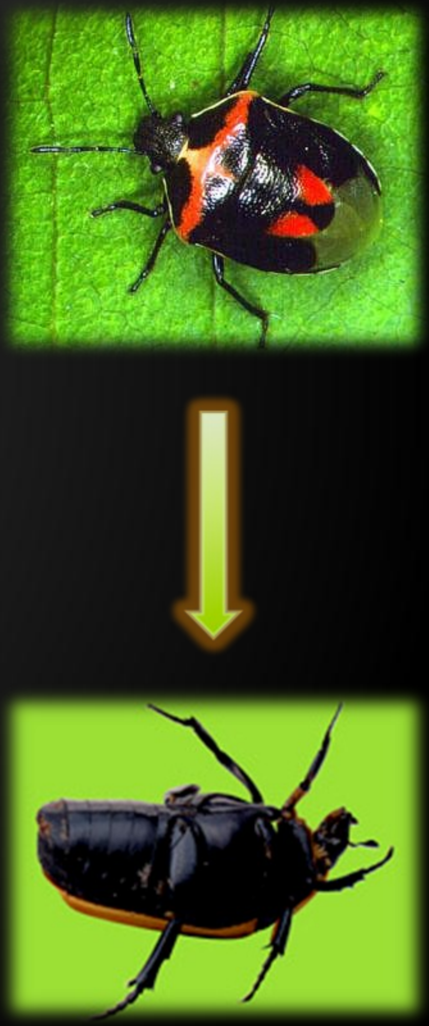
Running Programs

- The process of running application includes:
  - Compiling (if project not compiled)
  - Starting the application
- You can run application by:
  - Using Debug->Start menu
  - By pressing [F5] or [Ctrl+F5]

* NOTE: Not all types of projects are able to be started!

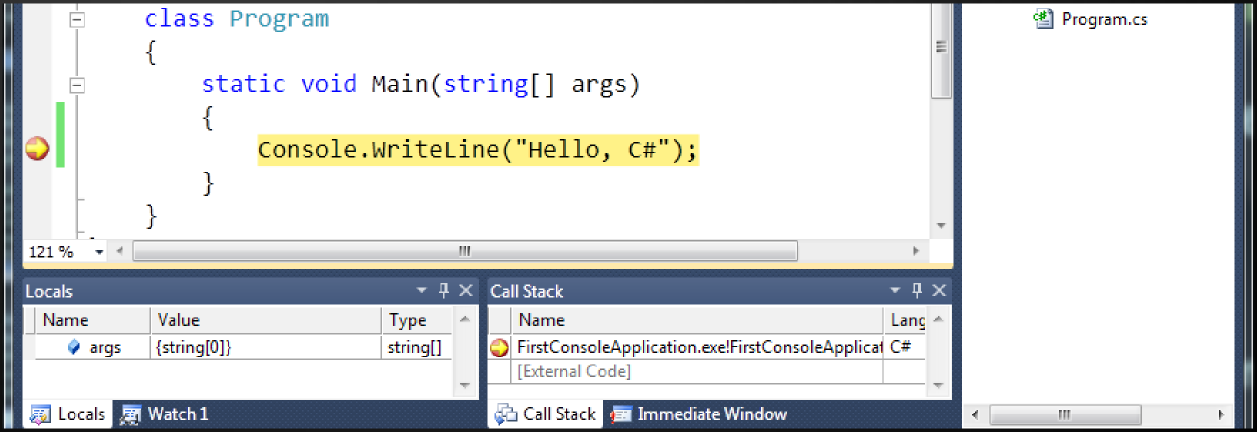
Debugging The Code

- The process of debugging application includes:
  - Spotting an error
  - Finding the lines of code that cause the error
  - Fixing the code
  - Testing to check if the error is gone and no errors are introduced
- Iterative and continuous process



Debugging in Visual Studio

- Visual Studio has built-in debugger
- It provides:
  - Breakpoints
  - Ability to trace the code execution
  - Ability to inspect variables at runtime



```
class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("Hello, C#");
    }
}
```

Locals

Name	Value	Type
args	{string[0]}	string[]

Call Stack

Name	Lang
FirstConsoleApplication.exe!FirstConsoleApplicat	C#
[External Code]	

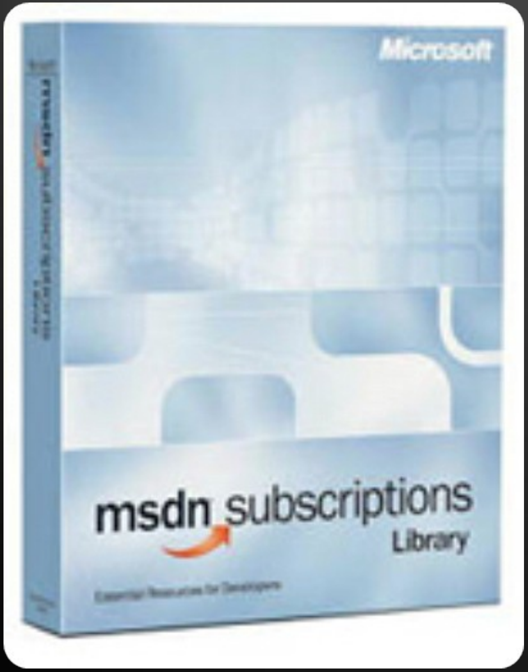
Program.cs
121 %
Locals
Watch 1
Call Stack
Immediate Window

Visual Studio

Compiling, Running and Debugging C# Programs

Live Demo

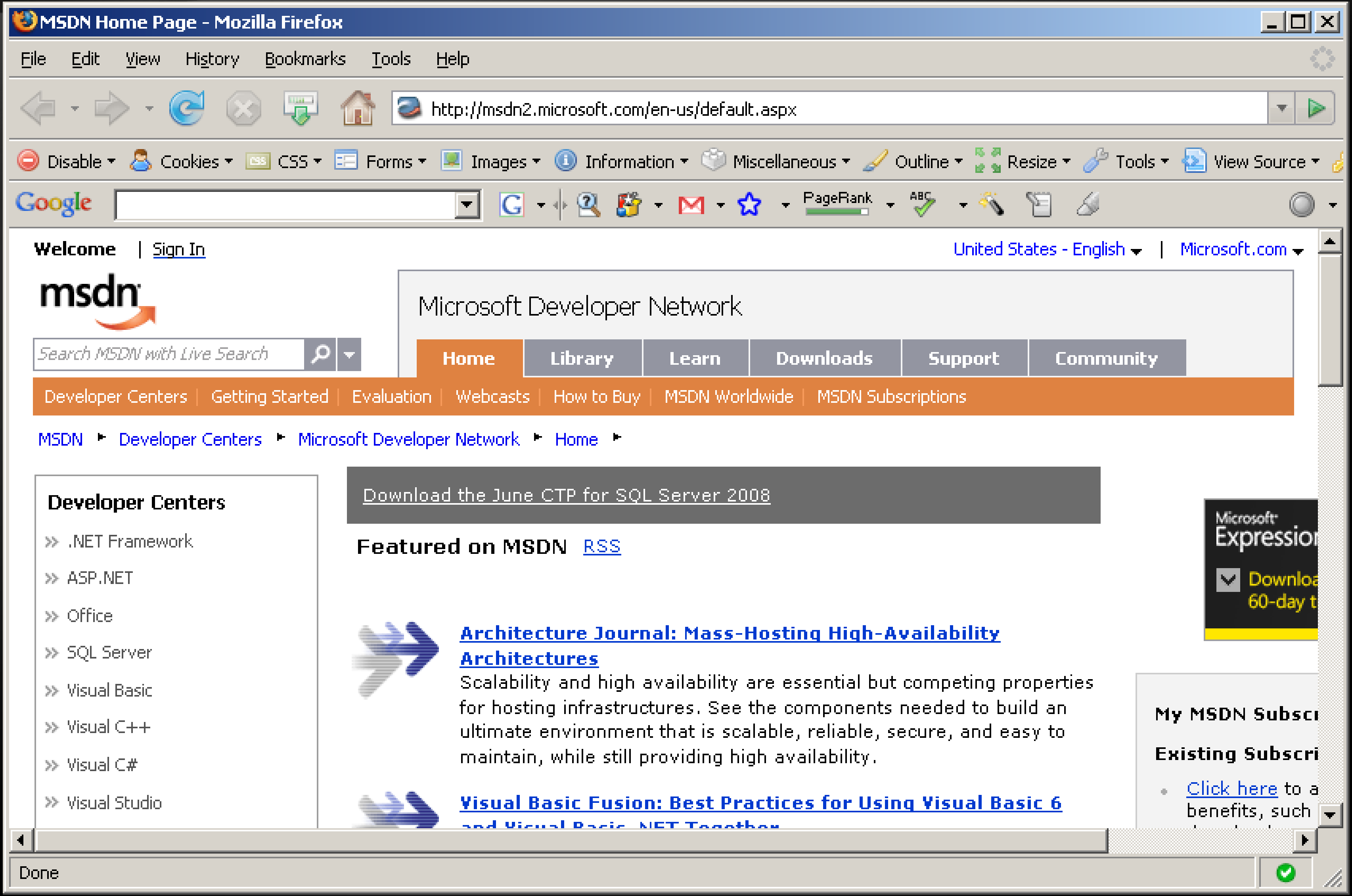
What is MSDN Library?


Microsoft
msdn subscriptions
Library
Essential Resources for Developers

What is MSDN Library?

- Complete documentation of all classes and their functionality
  - With descriptions of all methods, properties, events, etc.
  - With code examples
- Related articles
- Library of samples
- Use local copy or the Web version at http://msdn.microsoft.com/

MSDN Library


MSDN Home Page - Mozilla Firefox
File
Edit
View
History
Bookmarks
Tools
Help
Disable
Cookies
CSS
Forms
Images
Information
Miscellaneous
Outline
Resize
Tools
View Source
Google
PageRank
Done

http://msdn2.microsoft.com/en-us/default.aspx

Welcome | Sign In

United States - English | Microsoft.com

msdn

Search MSDN with Live Search

Microsoft Developer Network

Home | Library | Learn | Downloads | Support | Community

Developer Centers | Getting Started | Evaluation | Webcasts | How to Buy | MSDN Worldwide | MSDN Subscriptions

MSDN ▸ Developer Centers ▸ Microsoft Developer Network ▸ Home ▸

Developer Centers

- .NET Framework
- ASP.NET
- Office
- SQL Server
- Visual Basic
- Visual C++
- Visual C#
- Visual Studio

Download the June CTP for SQL Server 2008

Featured on MSDN RSS

Architecture Journal: Mass-Hosting High-Availability Architectures
Scalability and high availability are essential but competing properties for hosting infrastructures. See the components needed to build an ultimate environment that is scalable, reliable, secure, and easy to maintain, while still providing high availability.

Visual Basic Fusion: Best Practices for Using Visual Basic 6 and Visual Basic .NET Together

Microsoft Expressio
Downloa
60-day t

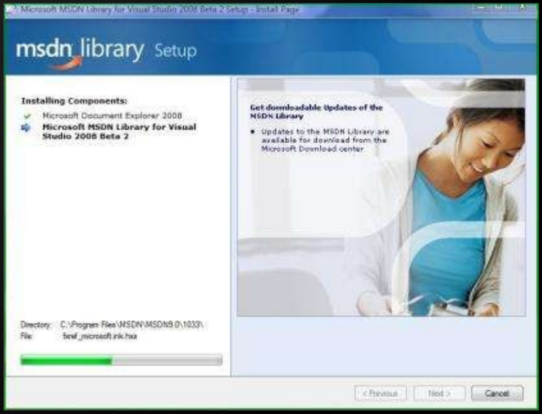
My MSDN Subscr

Existing Subscri

- Click here to a benefits, such

How to Use MSDN Library?

- Offline version
  - Use the table of contents
  - Use the alphabetical index
  - Search for phrase or keyword
  - Filter by technology
  - Browse your favorite articles
- Online version
  - Use the built-in search


Microsoft MSDN Library for Visual Studio 2008 Beta 2 Setup - Install Page

msdn library Setup

Installing Components:

✓ Microsoft Document Explorer 2008

Microsoft MSDN Library for Visual Studio 2008 Beta 2

Get downloadable Updates of the MSDN Library

- Updates to the MSDN Library are available for download from the Microsoft Download center

Directory: C:\Program Files\MSDN\MSDN9.0\1033\
File: 5nef_microsoft.ink.hxa

MSDN Library

Browsing and Searching Documentation

Live Demo

Questions?

how
where
when
who
why
what

Exercises

1. Familiarize yourself with:
   - Microsoft Visual Studio
   - Microsoft Developer Network (MSDN) Library Documentation
     - Find information about `Console.WriteLine()` method.
2. Create, compile and run a "Hello C#" console application.
3. Modify the application to print your name.
4. Write a program to print the numbers 1, 101 and 1001.

Exercises (2)

1. Install at home:
    1. Microsoft .NET Framework
    2. Microsoft Visual Studio (or Visual C# Express)
    3. Microsoft Developer Network (MSDN)
2. Create console application that prints your first and last name.
3. Create a console application that prints the current date and time.
4. Create a console application that calculates and prints the square of the number 12345.

Exercises (3)

1. Write a program that prints the first 10 members of the sequence: 2, -3, 4, -5, 6, -7, ...
2. Provide a short list with information about the most popular programming languages. How do they differ from C#?
3. Describe the difference between C# and .NET Framework.
4. * Write a program to read your age from the console and print how old you will be after 10 years.

*NOTE: If you have any difficulties, search in Google.