

## HelloWorld

### 思政目标

- 培养学生的爱国情怀
- 引导学生科技创新
- 鼓励学生自强不息

### 技能目标

- 了解 C#
- 学会安装 VS2013
- 熟悉 VS2013 开发环境
- 学会使用 VS2013 创建项目
- 掌握 VS2013 的断点调试方法
- 了解项目的文档构成

### 项目导读

学习一门语言，首先要从编辑环境开始，本项目主要介绍 C#的编辑环境 VS2013。

# 任务一 安装 VS2013 开发环境

## 任务引入

C#怎么读？有些人读作“C 井(jǐng)”，这样对吗？C#有什么特点呢？C#用什么环境开发呢？怎样安装 C#呢？下面一一作答。

## 任务分析

作为一种高级语言，C#有很多优点，其中比较突出的优点就是开发环境十分好用，操作方便，初学者很容易上手。

## 知识准备

C#（读作“C Sharp”）是一种面向对象、面向组件的新式编程语言。C#源于 C 语言系列，在 TIOBE 排行榜上稳居前列。它是由微软（Microsoft）公司开发的。目前，C#开发人员广泛使用的 IDE（Integrated Development Environment，集成开发环境）是微软公司的 Visual Studio，其最新版本是 Visual Studio 2022（VS2022）。VS2022 有多个版本，其中社区版 Visual Studio Community 2022 是面向学生、开放源代码参与者和个人的免费且功能齐全的 IDE。本书使用的开发环境是 VS2013，可以在官网界面的左下角找到“较早的下载项”链接并单击进入，再找到 VS2013 的下载界面。本书中的所有代码均使用 Visual Studio Ultimate 2013 编写并调试通过。Visual Studio 的软件兼容性是向后兼容（向“落后”版本兼容）的，即用 VS2022 可以打开并运行用 VS2013 开发的项目。

VS2013 安装完成后，在首次启动时，需要指定常用的开发类型，随后 VS2013 将针对此开发类型的设置进行优化。如图 1-1 所示，在“选择默认环境设置”对话框中选择“Visual C#开发设置”选项。此对话框只会出现一次，后期如果想重新设置默认环境，可以选择“工具”→“导入和导出设置”→“重置所有设置”命令。



图 1-1 “选择默认环境设置”对话框

## 任务二 创建控制台应用程序

### 任务引入

C#能开发像汇编语言、C 语言等黑屏白字的命令程序吗?当然能!

### 任务分析

命令程序在 VS2013 中被称为控制台应用程序,非常适合初学者在学习语法时使用。

### 项目实施

VS2013 常用的应用程序有 3 种,分别为控制台应用程序、Windows 窗体应用程序、ASP.NET Web 应用程序。本书在讲解 C#语法时使用控制台应用程序,在开发窗体项目时使用 Windows 窗体应用程序。

## C#程序设计

### 一、创建控制台应用程序项目

启动 VS2013 以后，选择“文件”→“新建”→“项目”命令，打开“新建项目”对话框，在左侧选择“Visual C#”选项，如果首次运行软件时在“选择默认环境设置”对话框中没有选择“Visual C#开发设置”选项，会默认选择其他选项，需要手动切换到“Visual C#”选项，并选择“控制台应用程序”选项，如图 1-2 所示。设置“位置”为“D:\CSharp\”，“名称”为“CHelloWorld”，约定控制台应用程序名称的首字符是“C”。

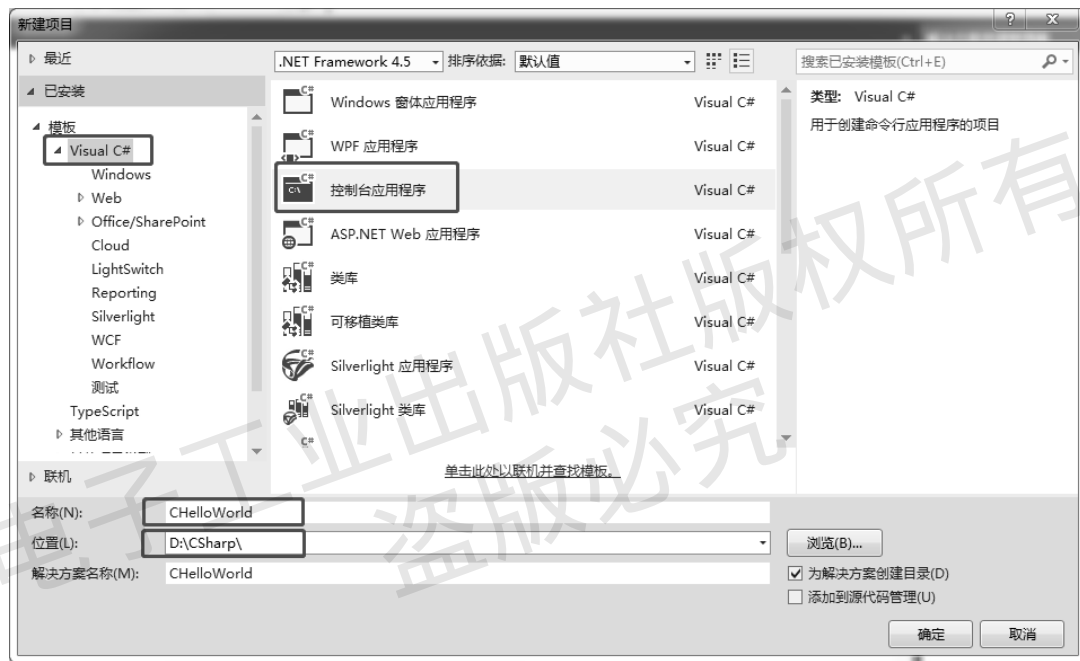


图 1-2 “新建项目”对话框

### 二、设置窗口布局

创建项目以后，先对环境布局进行设置，选择“视图”菜单，分别打开“工具箱”“服务器资源管理器”“解决方案资源管理器”“属性”“错误列表”5个窗口，并通过移动光标来改变5个窗口的位置，最终达到如图 1-3 所示的布局效果。也可以通过“窗口”→“重置窗口布局”命令来设置，窗口布局可以根据个人的开发习惯进行设置。其中，“工具箱”和“属性”窗口在控制台应用程序中用不到，主要为后面的 Windows 窗体应用程序做准备，“服务器资源管理器”窗口也暂时用不到，在以后编写数据库的相关程序时使用。在“解决方案资源管理器”窗口中列出了项目中包含的所有文档，“错误列表”窗口用于项目调试，可以列出程序的出错信息。

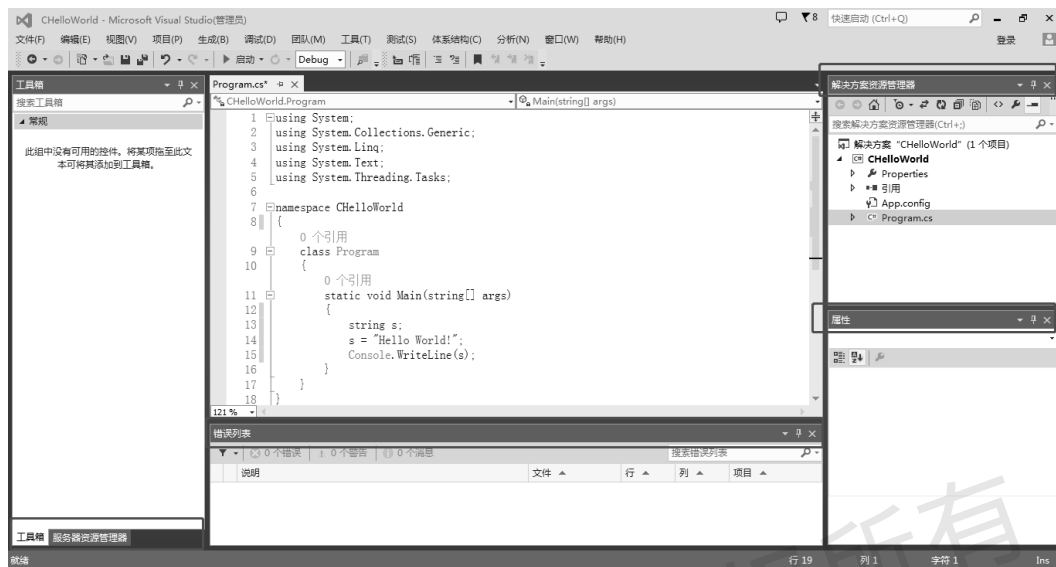


图 1-3 布局效果

### 三、设置颜色主题

可以通过“工具”→“选项”命令，打开“选项”对话框，如图 1-4 所示。在左侧选择“环境”→“常规”选项，在右侧对“颜色主题”选项进行设置，系统默认为“蓝色”，也可以根据个人习惯设置为“深色”，这样就变成了以深色为主的酷炫效果。

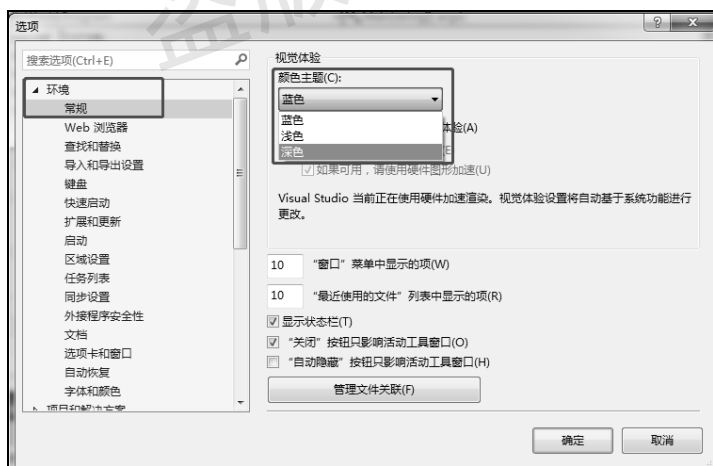


图 1-4 “选项”对话框 1

### 四、设置行号和字体

如图 1-5 所示，在“选项”对话框中，选择左侧的“文本编辑器”→“C#”选项，并勾选右侧的“行号”复选框，这样在代码编写界面的左侧就会显示行号。同理，还可以对文字大小等进行设置。文字大小可以通过滚动鼠标中键进行调整，方法是先把光标

## C#程序设计

定位到代码编写界面，然后在按住“Ctrl”键的同时滚动鼠标中键。

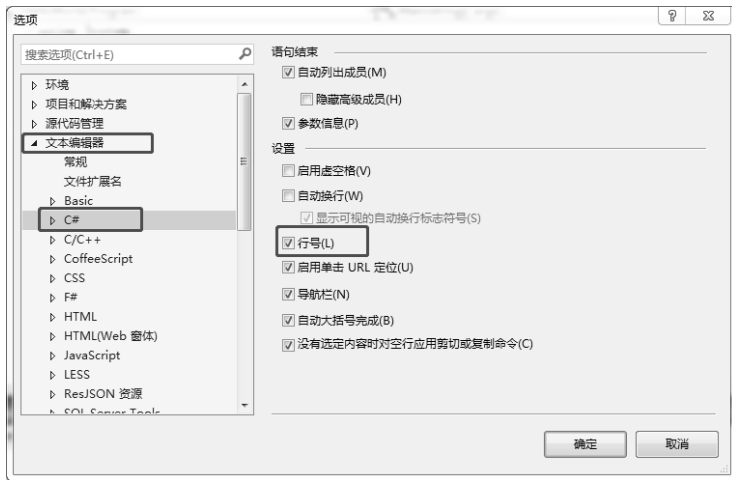


图 1-5 “选项”对话框 2

## 五、录入代码

窗口布局、字体等设置完成以后，在代码编写界面显示的是系统自动生成的代码，在 Main()方法中录入 3 行代码，完成第一个程序。由于 VS2013 中有代码提示功能，所以在录入代码时，对于包含字符个数较多的关键字或变量名，只需要先录入前几个字母，然后配合方向键和回车键即可完成快速录入。完整代码如下：

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

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

代码说明：

- ① 关键字 using 用于引入命名空间。此项目自动引入 5 个命名空间。
- ② 关键字 namespace 用于定义此项目的命名空间为 CHelloWorld。

③ 关键字 `class` 用于定义类 `Program`。类名首字母一般大写，C#对字母是大写还是小写十分敏感。

④ `Main()`方法是程序的入口地址（或称主方法名），首字母大写，后面的小括号是参数列表。

⑤ 一对大括号指明了命名空间、类、方法（又叫作函数）的作用范围。

⑥ `Main()`方法中的 3 行代码是手动录入的，包含 3 条语句，每条语句都是以分号结束的。3 条语句的功能如下：

```
string s;           //用关键字 string 定义一个变量 s
s="Hello World!";  //为变量 s 赋值
Console.WriteLine(s); //调用类 Console 的 WriteLine() 方法,把变量 s 的值输出到
                    //控制台上
```

## 六、运行

代码录入完成后，选择“调试”→“开始执行不调试”命令，或直接按“Ctrl+F5”组合键（键盘上有“Fn”键的计算机需要按“Fn+Ctrl+F5”组合键）运行并查看结果。以上代码运行结果如图 1-6 所示。按任意键关闭命令行窗口，结束程序运行。

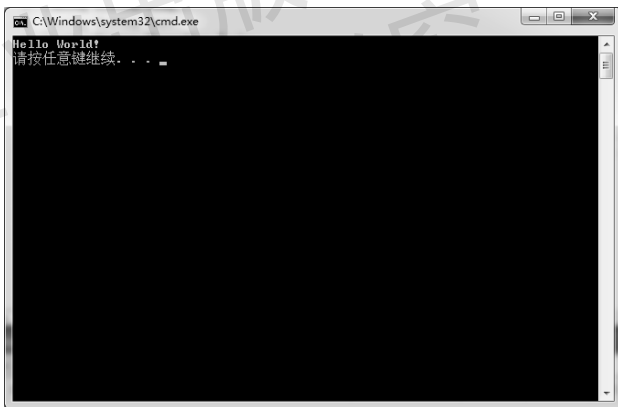


图 1-6 运行结果

## 七、断点调试

断点调试是 VS2013 中常用的排错方法，可以帮助程序员快速找到逻辑错误的原因。在调试之前应先设置断点，断点一般设置在可能出错的语句的上一条语句中。假设在本程序中，要把断点设置在第 13 行，即语句 `string s;` 所在行，则操作步骤如下。

① 单击第 13 行最前面的灰色区域，就会把这一行设置为断点，同时在行首会出现一个小红点（即断点），且这行的代码底纹变为深红色，如图 1-7 所示。

② 选择“调试”→“启动调试”命令或直接按“F5”键，也可以直接单击“启动”按钮，开始调试，程序运行后会停在这一行，等待下一步操作。若此时将光标悬停到变

## C#程序设计

量 s 上, 那么可以看到变量 s 的当前值为 null, 即空字符串, 如图 1-8 所示。

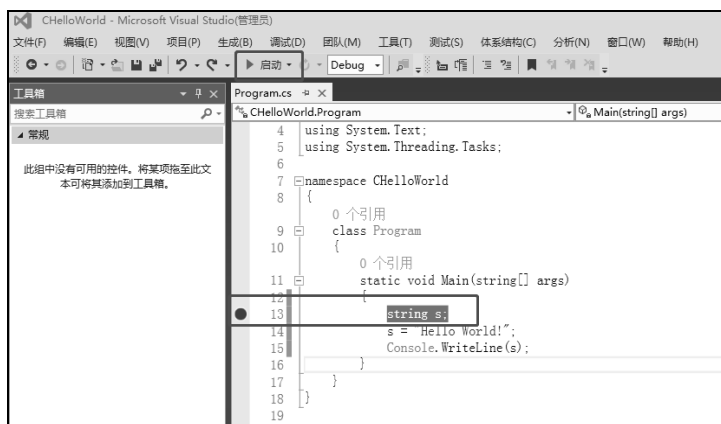


图 1-7 设置断点

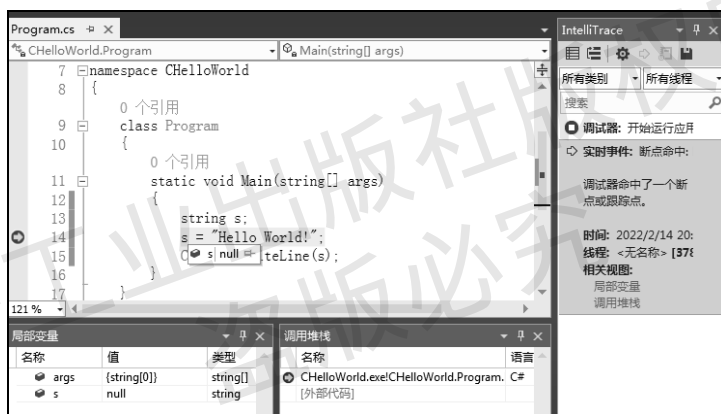


图 1-8 调试结果 1

③ 继续选择“调试”→“逐语句”命令或直接按“F11”键, 执行完当前语句, 进入下一条语句, 此时将光标悬停到变量 s 上, 就可以看到变量 s 的当前值为"Hello World!", 如图 1-9 所示。

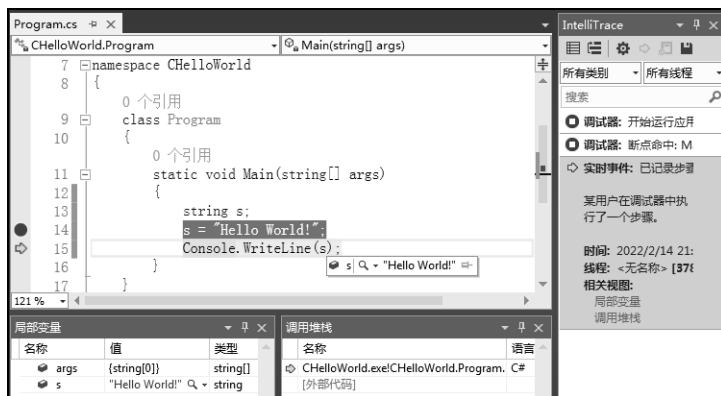


图 1-9 调试结果 2



④ 如此重复，不停地按“F11”键执行下去，可以跟踪变量的变化情况，直到找出错误。如果想取消断点，可以单击断点所在的代码行首，断点将会消失，即取消了断点。

## 八、项目文件构成

如图 1-10 所示，分别查看“解决方案资源管理器”窗口和项目存放的物理目录，通过对比可以发现以下内容。

在解决方案“CHelloWorld”下包含一个项目，即 CHelloWorld 和一个解决方案文件，即 CHelloWorld.sln（后缀.sln 表示是 VS2013 解决方案文件）。当然，通过后续的学习，学生还可以知道一个解决方案可以包含多个项目。

项目 CHelloWorld 中包含 3 个目录和 3 个文件夹。其中，在文件夹 bin 和 obj 中可以找到可执行文件 CHelloWorld.exe，双击即可运行，在文件夹 Properties 中存放的是项目资源相关文件，App.config 是项目配置文件，CHelloWorld.csproj 是 C# 项目架构文件，Program.cs 是 C# 源文件（手动编写的代码就存放在这里，其他代码都是 VS2013 自动生成的），如图 1-11 所示。

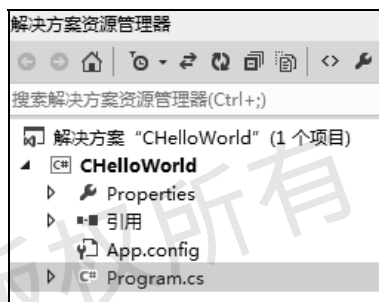


图 1-10 “解决方案资源管理器”窗口

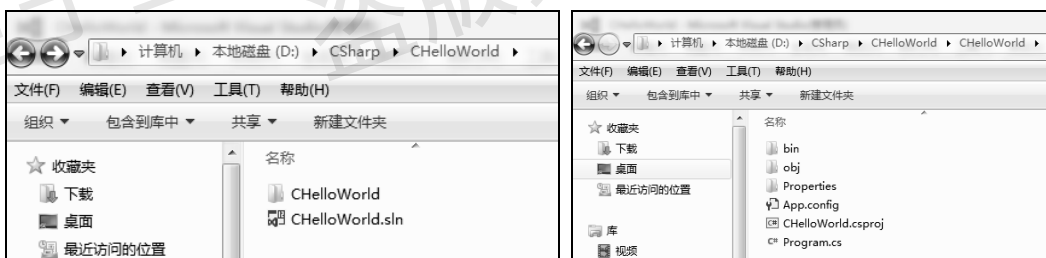


图 1-11 CHelloWorld 和 CHelloWorld.sln

## 任务三 创建 Windows 窗体应用程序

### 任务引入

C#除了可以开发命令程序，还能开发类似 Windows 中那样的窗口程序吗？当然能。

### 任务分析

Windows 中的窗口程序在 VS2013 中被称为 Windows 窗体应用程序，其应用非常广泛。

### 项目实施

#### 一、创建 Windows 窗体应用程序项目

启动 VS2013 以后，选择“文件”→“新建”→“项目”命令，打开“新建项目”对话框，在左侧选择“Visual C#”选项，并选择中间的“Windows 窗体应用程序”选项，如图 1-12 所示。设置“位置”为“D:\CSharp\”，“名称”为“WHelloWorld”。约定 Windows 窗体应用程序名称的首字符是“W”。

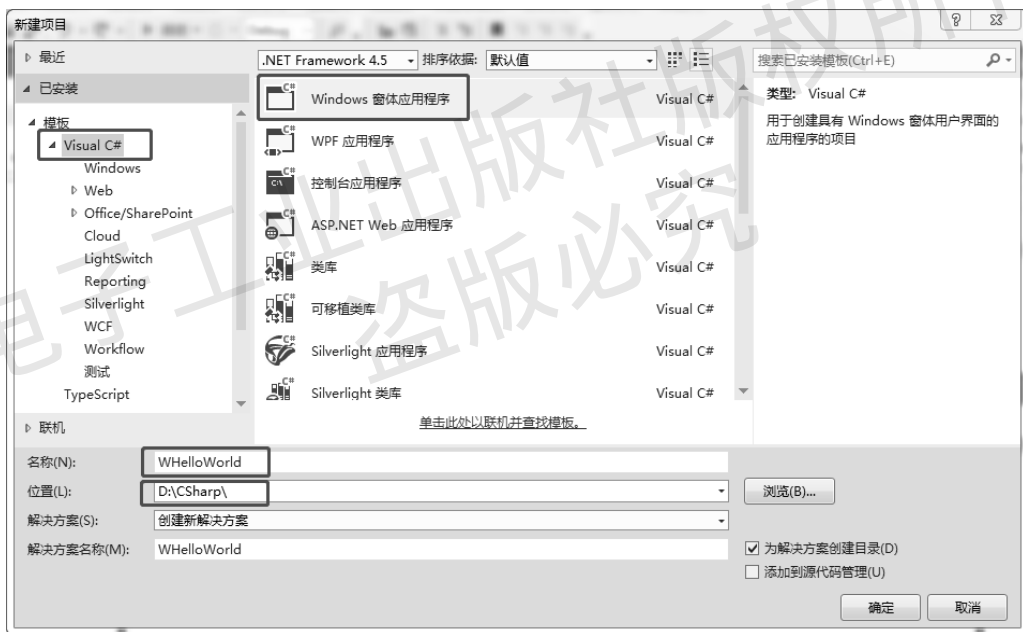


图 1-12 “新建项目”对话框

#### 二、添加控件

创建项目以后，选择“工具箱”窗口中的“公共控件”→“Button”选项，并将其拖动到 Form1 窗体中，这样就在 Form1 窗体中放置了一个按钮控件，如图 1-13 所示。或在“工具箱”窗口中选择“Button”选项，并将光标移动到 Form1 窗体中，光标会变为十字形，此时移动光标绘制一个按钮控件。这两种操作的效果是相同的。区别是前者放置的按钮大小是系统默认的，后者放置的按钮大小是绘制出来的。

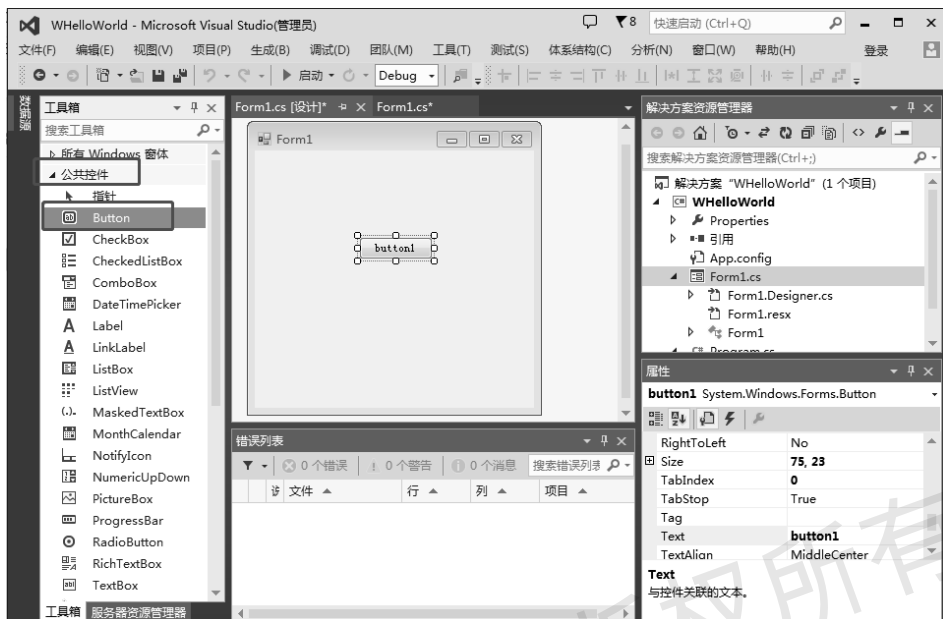


图 1-13 添加按钮控件

### 三、修改控件属性

添加按钮控件以后，先单击刚刚放置的按钮控件，然后在“属性”窗口中选择“Text”选项，将原来的“button1”修改为“点我一下”，如图 1-14 所示。此时，窗体中的按钮控件会显示“点我一下”。

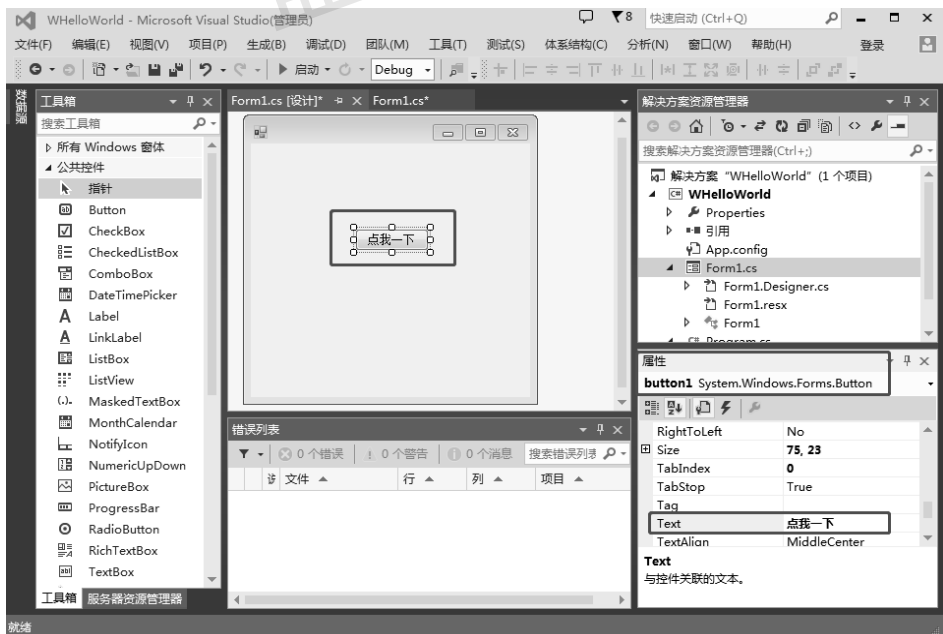


图 1-14 修改控件属性

## 四、给控件添加事件并录入代码

双击“点我一下”按钮，系统会自动为这个按钮控件添加 Click 事件，同时会自动切换到“Form1.cs”窗口，此时可以发现后台代码自动生成 `button1_Click()`方法。在这个方法中录入一行代码（见图 1-15），会发现“工具箱”和“属性”窗口中的内容都消失了，这是正常的，切换到“Form1.cs[设计]”窗口，这两个窗口中的内容就恢复了。

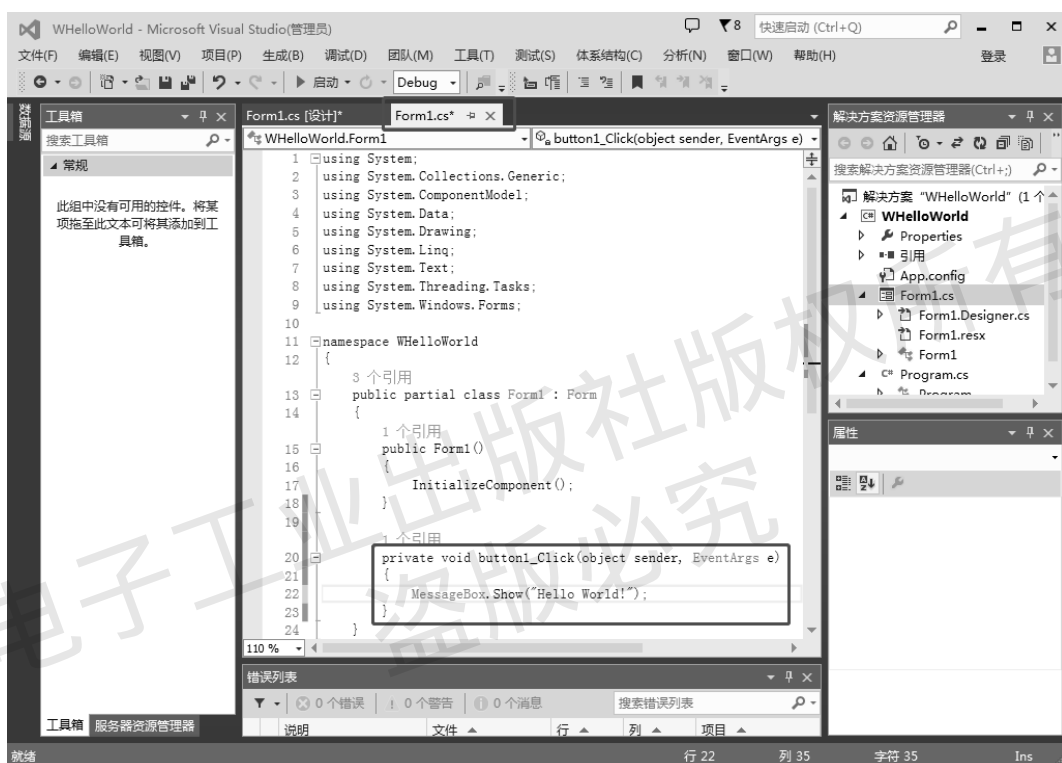


图 1-15 给控件添加事件并录入代码

完整代码如下：

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WHelloWorld
{
    public partial class Form1 : Form
```

```

{
    public Form1()
    {
        InitializeComponent();
    }

    private void button1_Click(object sender, EventArgs e)
    {
        MessageBox.Show("Hello World!");
    }
}
}

```

代码说明：

- ① 关键字 `using` 用于引入命名空间。此项目自动引入 9 个命名空间。
- ② 关键字 `namespace` 用于定义此项目的命名空间为 `WHelloWorld`。
- ③ 关键字 `class` 用于定义类 `Form1`；关键字 `partial` 用于说明类 `Form1` 是分部级，其中一部分在 `Form1.cs` 文件中，另一部分在 `Form1.Designer.cs` 文件（系统自动生成）中；冒号用于说明子类 `Form1` 继承自父类 `Form`。
- ④ `button1_Click()` 方法是在设计窗口界面双击按钮自动生成的方法，绑定了按钮的 `Click` 事件，后面的小括号是参数列表。
- ⑤ 程序的入口地址 `Main()` 方法在另一个文件 `Program.cs` 中。
- ⑥ `button1_Click()` 方法中的一行代码是手动录入的，包含一条语句，功能是调用类 `MessageBox` 的 `Show()` 方法，会弹出一个消息对话框，消息内容是“Hello World!”。

## 五、运行

在代码录入完成后，直接单击“调试”菜单栏下方的“启动”按钮，开始运行。运行后，单击“点我一下”按钮，弹出消息对话框，消息内容是“Hello World!”，运行结果如图 1-16 所示。关闭消息对话框和 `Form1` 窗体，结束程序运行。



图 1-16 运行结果

### 六、项目文件构成

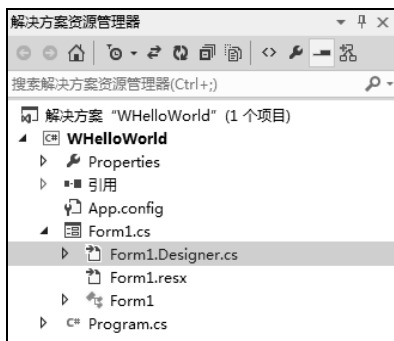


图 1-17 “解决方案资源管理器”窗口

如图 1-17 所示，分别查看“解决方案资源管理器”窗口和项目存放的物理目录，通过对比可以发现与控制台应用程序不同的是，Windows 窗体应用程序的入口地址 Main()方法存放在 C#源文件 Program.cs 中，并通过源文件 Program.cs 调用类 Form1，而类 Form1 是分部类，分别存放在文件 Form1.cs 和 Form1.Designer.cs 中。手动编写的代码存放在文件 Form1.cs 中，文件 Form1.Designer.cs 中存放的是系统自动生成的代码，文件 Form1.resx 中存放的是类 Form 中使用的资源，如图 1-18 所示。



图 1-18 WHelloWorld 和 WHelloWorld.sln

## 项目总结

本项目讲解了 C#开发环境 VS2013 的安装、设置和使用；分别用控制台应用程序和 Windows 窗体应用程序实现了入门程序 HelloWorld；讲解了断点调试排错方法的详细步骤，以后会经常使用；分析了解决方案的文件构成，对于以后涉及的代码分享和项目迁移等问题，最好把整个解决方案目录打包复制。

## 项目提升

创建两个程序，分别是控制台应用程序和 Windows 窗体应用程序。项目名称为自己姓名的全拼，输出一句话：“我是 XXX，这是我的第一个 C#程序。”