

# 项目 1

## 创建 VB.NET 程序开发环境

得益于广阔应用场景及巨大市场需求的有力支撑,2022年,我国软件和信息技术服务业(以下简称软件业)运行稳步向好,软件业务收入跃上10万亿元台阶,盈利能力保持稳定,软件业务出口保持增长。

本项目介绍 VB.NET 应用程序开发环境及简单程序设计流程,以此来介绍 VB 开发 Windows 应用程序开发技术的结构、特点和开发流程。让读者对 Windows 应用程序开发有一个直观的认识,为本教材后续介绍的 Windows 应用程序开发打下基础。

### 知识目标:

- (1) 了解 .NET 平台的基本结构;
- (2) 了解 Windows 应用开发技术的原理;
- (3) 掌握 VB.NET 简单语句的结构;
- (4) 掌握 VB.NET 程序调试的流程。

### 能力目标:

- (1) 掌握安装 Visual Studio 2022 的方法;
- (2) 掌握创建 VB.NET 应用程序的方法;
- (3) 掌握简单 VB.NET 应用程序的设计流程;
- (4) 掌握简单 VB.NET 控件的使用方法。

### 课程思政目标:

- (1) 引导学生树立正确的世界观、人生观、价值观,自觉践行社会主义核心价值观;
- (2) 培养学生发现问题、分析问题和解决问题的能力;
- (3) 明确软件开发专业人员工作性质的社会价值。

## 任务 1.1 创建 VB.NET 程序开发环境

### 1.1.1 安装 Visual Studio 2022 编程环境

Visual Studio 2022 能够开发的程序包括常见的 Visual Basic、Visual C#、Visual C++ 和 Python 等。Visual Basic 开发是 Visual Studio 2022 一个重要的组成部分。

微软于 2021 年发布 Visual Studio 2022 的首个预览版。2022 版本更快、更易于使用、更具有轻量级特点,专为学习者和构建工业规模解决方案的人员设计。

## 1. Visual Studio 2022 编程环境可以开发的常用项目类型

(1) 桌面应用程序：可以使用 Visual Studio 2022 开发 Windows 桌面应用程序，如 Windows Forms 应用程序、WPF 应用程序和 Universal Windows Platform (UWP) 应用程序。

(2) Web 应用程序：可以使用 Visual Studio 2022 开发 Web 应用程序，如 ASP.NET 应用程序、ASP.NET Core 应用程序和 Node.js 应用程序。

(3) 移动应用程序：可以使用 Visual Studio 2022 开发移动应用程序，如 Android 应用程序、iOS 应用程序和 Windows Phone 应用程序。

(4) 云应用程序：可以使用 Visual Studio 2022 开发云应用程序，如 Azure 应用程序和 AWS 应用程序。

(5) 游戏开发程序：可以使用 Visual Studio 2022 开发游戏，如 Unity 游戏和游戏插件。

(6) 数据库应用程序：可以使用 Visual Studio 2022 开发数据库应用程序，如 SQL Server 应用程序和 MySQL 应用程序。

## 2. 安装 Visual Studio 编程环境

Visual Studio 2022 最低系统要求如表 1-1 所示。

表 1-1 Visual Studio 2022 安装所需系统

类别	说明
支持的操作系统	在以下 64 位操作系统上支持 Visual Studio 2022： (1) Windows 11 版本 21H2 或更高版本，如家庭版、专业版、专业教育版、专业工作站版、企业版和教育版。 (2) Windows 10 版本 1909 或更高版本，如家庭版、专业版、教育版和企业版。 (3) Windows Server Core 2022。 (4) Windows Server Core 2019。 (5) Windows Server 核心 2016。 (6) Windows Server 2022 标准和数据中心。 (7) Windows Server 2019 标准和数据中心。 (8) Windows Server 2016 标准和数据中心
硬件	(1) 建议使用四核或更好的处理器，如 ARM64 或 x64 处理器。不支持 ARM 32 处理器。 (2) 内存至少 4GB。许多因素都会影响所使用的资源。对于典型的专业解决方案，建议使用 16GB 内存。 (3) 硬盘要有 850MB~210GB 可用空间，具体取决于安装的功能；典型安装需要 20~50GB 的可用空间。建议在固态硬盘上安装 Windows 和 Visual Studio 以提高性能。 (4) 支持最低显示分辨率为 WXGA (1366 像素×768 像素) 的显卡。Visual Studio 最适宜的分辨率为 1920 像素×1080 像素或更高。

以下是 Visual Studio Community 2022 版本安装步骤。

(1) 启动安装程序，如图 1-1 所示。

(2) 单击“继续”按钮，进入如图 1-2 所示的界面，显示下载及安装进度。

(3) 下载安装完成以后，进入如图 1-3 所示的界面。

(4) 选择安装的选项，如图 1-4 所示。



编程环境的安装过程



图 1-1 启动安装程序界面



图 1-2 显示下载及安装进度



图 1-3 安装选项选择界面



图 1-4 选择需要安装的模块

(5) 单击“安装”按钮，进入如图 1-5 所示的界面，显示下载和安装进度。

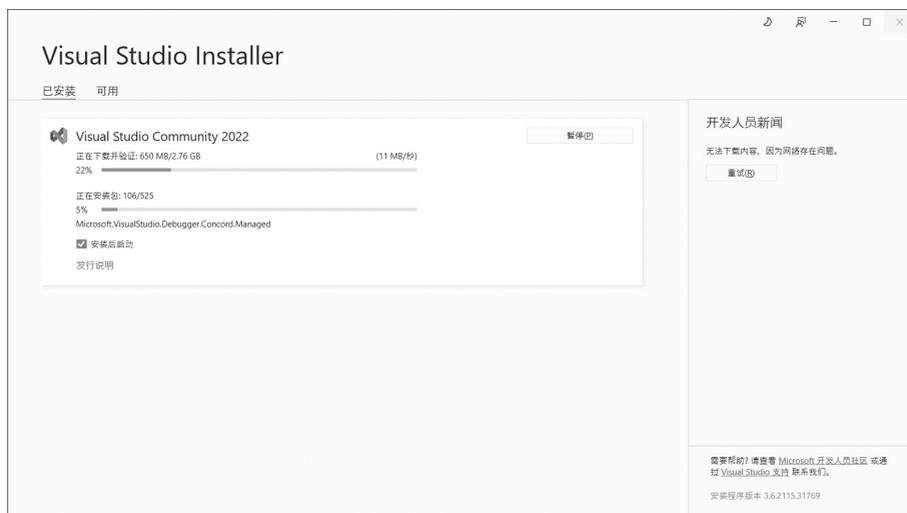


图 1-5 下载和安装界面

(6) 下载和安装需要一段时间。安装完成后，会出现如图 1-6 所示的界面，选择是否重启操作系统。



图 1-6 确定是否重启系统

(7) 重启操作系统之后，重新打开 Visual Studio，进入如图 1-7 所示的界面。



图 1-7 开发设置界面

(8) 从中选择“开发设置”“颜色主题”，进入如图 1-8 所示的界面。



图 1-8 启动后的 Visual Studio

## 1.1.2 熟悉 Visual Studio 2022 编程环境

Visual Studio 2022 将程序开发中用到的各种功能集成在一个公共的工作环境中，称为 IDE。在该编程开发环境中提供了各种控件、窗口和方法，用户可以方便地进行各种应用程序的开发，以及在各种开发界面之间切换，可以在很大程度上节约开发时间。

### 1. Visual Studio 2022 创建 VB.NET Windows 应用程序

(1) 启动 Visual Studio 2022，初始界面如图 1-9 所示。



图 1-9 Visual Studio 2022 启动后的初始界面

(2) 选择“创建新项目”，然后进入如图 1-10 所示的创建新项目的界面。

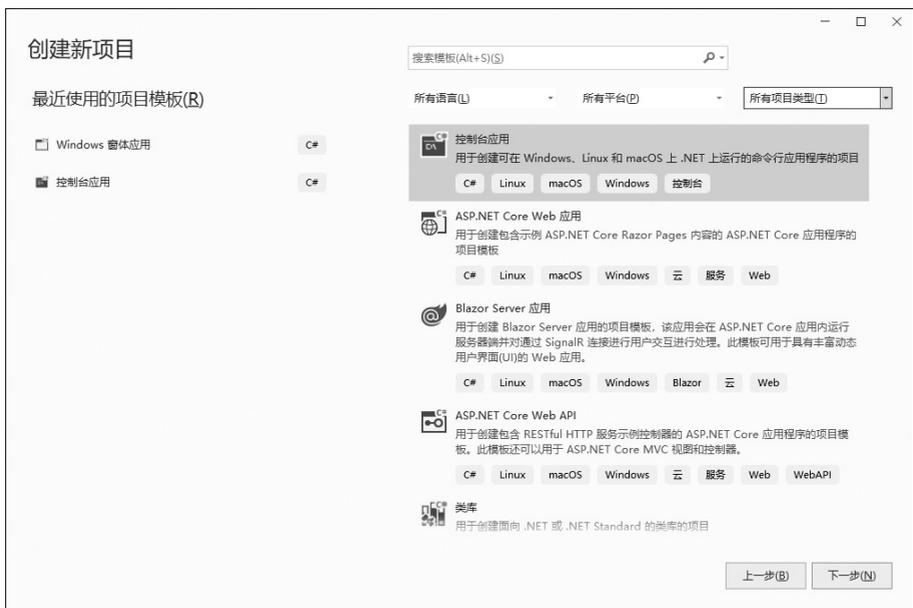


图 1-10 创建 VB.NET Windows 应用程序

(3) 在“所有语言”下拉列表中选择 Visual Basic, 在“所有项目类型”下拉列表中选择“桌面”, 然后在出现的可以创建项目的模板里选择“Windows 窗体应用”, 如图 1-11 所示。

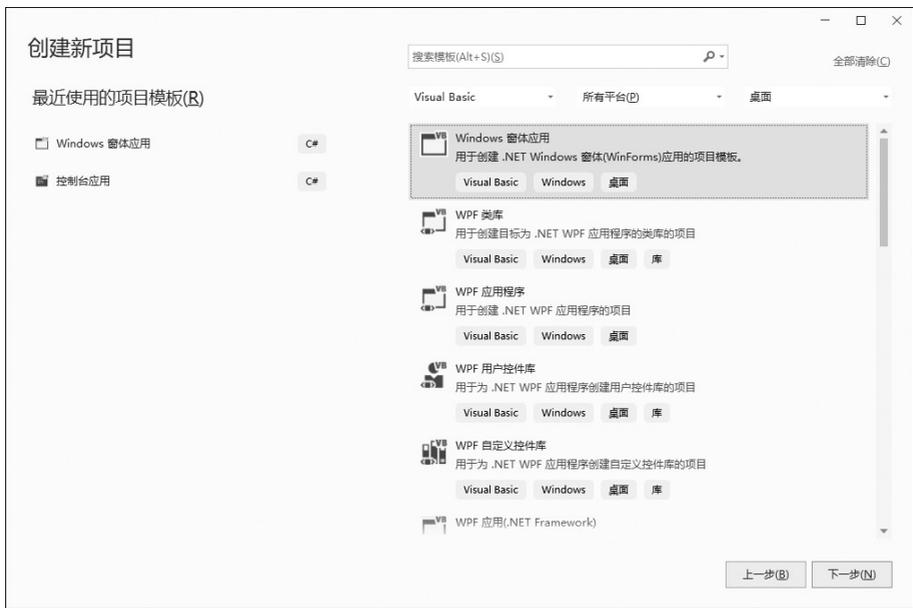


图 1-11 选择创建“Windows 窗体应用”模板

(4) 单击“下一步”按钮, 进入如图 1-12 所示的配置新项目的界面。

(5) 设置好项目名称, 选择项目的保存位置, 设置好解决方案名称, 然后单击“下一步”按钮, 进入如图 1-13 所示的界面选择框架版本。



图 1-12 配置新项目



图 1-13 选择框架版本

(6) 单击“创建”按钮，将完成项目的创建，进入如图 1-14 所示的界面。

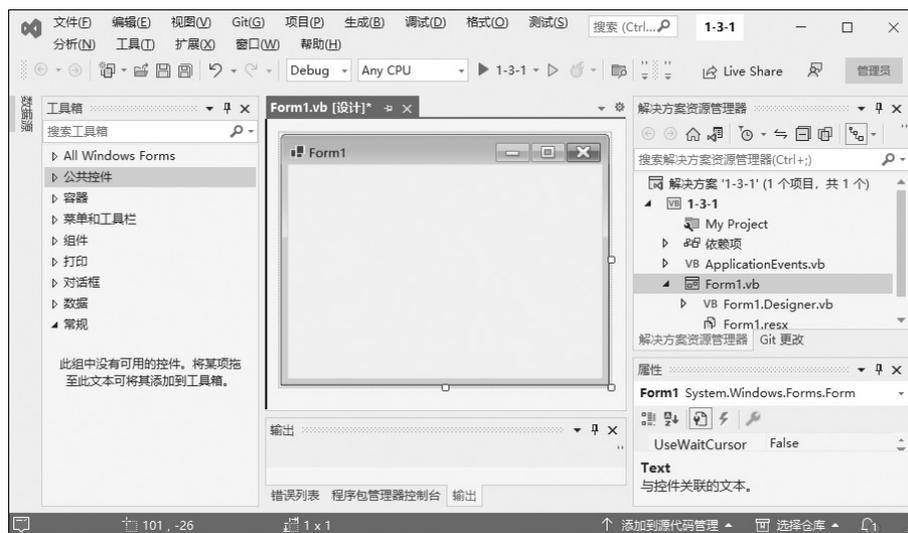


图 1-14 新建的 VB.NET Windows 应用程序



图 1-15 “文件”菜单的下拉菜单项

## 2. Visual Studio 2022 的开发环境的组成

Visual Studio 2022 的开发环境主要由菜单、工具栏、窗体、工具箱、属性窗口和解决方案资源管理器等几部分组成。

(1) 菜单。菜单包括“文件”“编辑”“视图”等，其中包含了开发 VB.NET 程序常见的命令。

① “文件”菜单如图 1-15 所示，它包含的主要菜单项说明如下。

新建：支持新建项目、仓库、文件等。

打开：支持打开已有的项目 / 解决方案、文件夹、网站、文件等。

关闭：关闭正在编写的项目。

关闭解决方案：关闭正在编写的解决方案。

退出：退出 Visual Studio 2022 编程环境。

② “编辑”菜单包含的主要菜单项有“转到”“查找和替换”“撤销”“重做”“剪切”“复制”“粘贴”等，如图 1-16 所示。

③ “视图”菜单包含的主要菜单项有“代码”“设计器”“解决方案资源管理器”“服务器资源管理器”“类视图”“工具箱”等，如图 1-17 所示。具体说明如下。

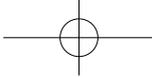


图 1-16 “编辑”菜单的下拉菜单项



图 1-17 “视图”菜单的下拉菜单项

代码：打开代码编辑界面。



设计器：打开设计器编辑界面。

解决方案资源管理器：打开解决方案资源管理器窗口。

服务器资源管理器：打开和服务器以及数据库相关内容的操作界面。

类视图：打开类视图窗口。

工具箱：打开工具箱窗口。

④ Git 菜单包含的菜单项有“克隆”“创建 Git 仓库”“本地存储库”“设置”，如图 1-18 所示。



图 1-18 Git 菜单的下拉菜单项

⑤ “项目”菜单包含的主要菜单项有“添加窗体 (Windows 窗体)”“添加用户控件 (Windows 窗体)”“添加类”“设为启动项目”等，如图 1-19 所示。具体说明如下。

添加窗体 (Windows 窗体)：向已有的应用程序中添加新的窗体。

添加用户控件 (Windows 窗体)：添加用户自定义的控件。

添加类：添加 VB.NET 的类。

设为启动项目：将正在编辑的项目设为启动项。

⑥ “调试”菜单包含的主要菜单项有“开始调试”“开始执行 (不调试)”等，如图 1-20 所示。具体说明如下。



图 1-19 “项目”菜单的下拉菜单项



图 1-20 “调试”菜单的下拉菜单项

开始调试：启动正在编辑的项目进行调试。

开始执行 (不调试)：启动正在运行的项目。

(2) 工具栏。工具栏在菜单栏的下面，如图 1-21 所示。工具栏提供了常用命令的快速访问按钮，单击某个按钮，可执行对应的操作，效果和使用菜单是一样的。

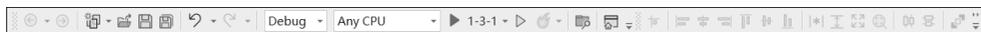


图 1-21 工具栏的界面

(3) 窗体。在创建了一个 Windows 应用程序后,系统会自动生成一个默认的窗体,也就是应用程序界面。在开发过程中,用户编程使用的各种控件就是布局在窗体之上的,当程序运行时,用户所看到的就是窗体。应用程序的设计界面及各功能窗口如图 1-22 所示。

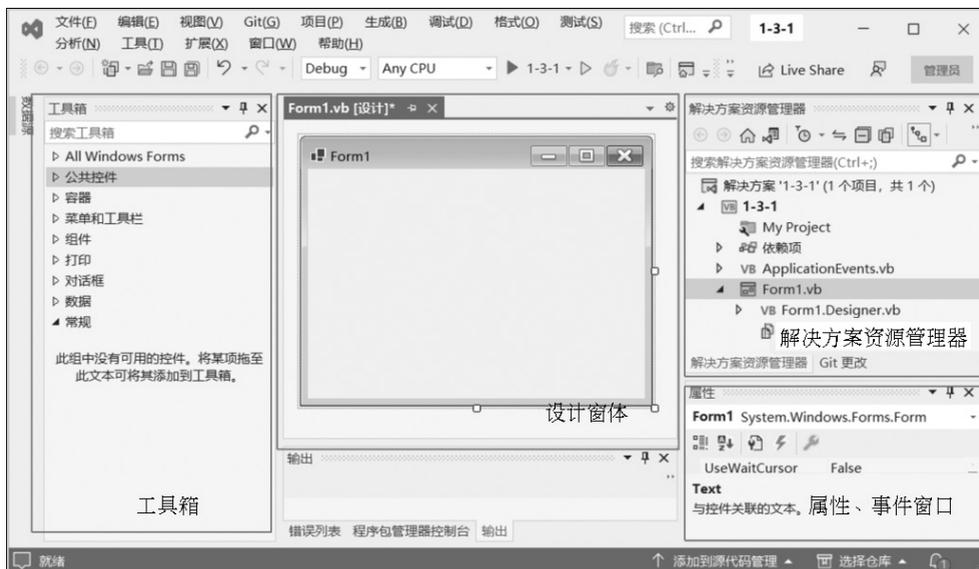


图 1-22 应用程序的设计界面及各功能窗口

(4) 工具箱。工具箱中提供了各种控件、容器、菜单和工具栏、组件、对话框和数据等。在默认情况下,工具箱将控件和各种组件按照功能的不同进行了分类,如图 1-23 所示。

用户在编程过程中可以根据需要选择各种控件和组件。如果所需要的控件或者组件在工具箱中找不到,可以在工具箱区域内右击,在出现的快捷菜单中选择“选择项”命令,打开“选择工具箱项”对话框,在该对话框中选择相应的控件或组件,如图 1-24 所示。



图 1-23 工具箱

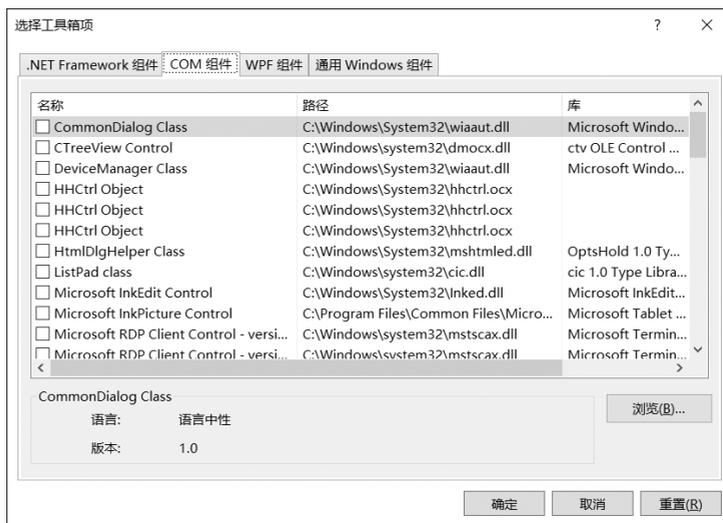


图 1-24 “选择工具箱项”对话框