

Aplicaciones para MS Windows



Aplicaciones para MS Windows

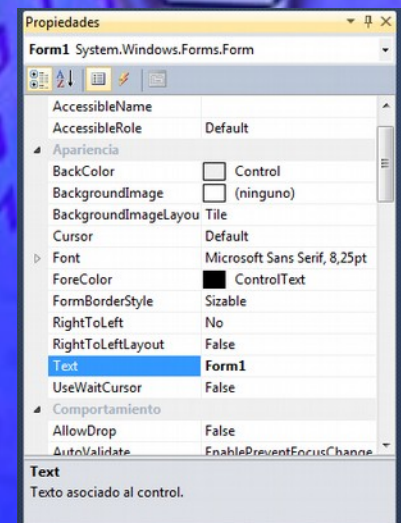
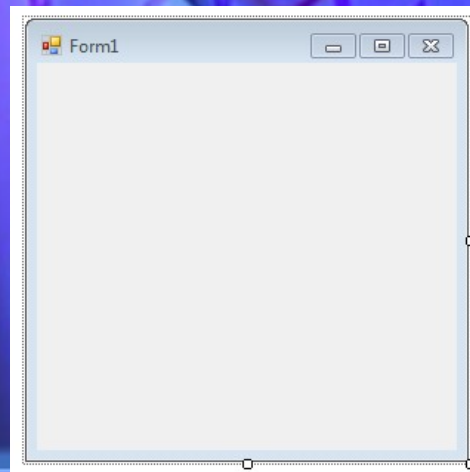
- En la plataforma .NET el espacio de nombres que ofrece las clases necesarias para construir aplicaciones de escritorio bajo MS Windows se denomina Windows Forms. Este es también el nombre genérico que se le otorga ahora a este tipo de programas basados en ventanas.
- Windows Forms es una de las dos tecnologías que se utiliza en Visual C# para crear aplicaciones cliente inteligentes basadas en MS Windows que se ejecutan en .NET Framework. Windows Forms es especialmente adecuado para escenarios de desarrollo rápido de aplicaciones donde la prioridad principal no es una interfaz gráfica de usuario compleja. El Diseñador de Windows Forms se utiliza para crear la interfaz de usuario, y permite obtener acceso a otras características de diseño y ejecución.
- Windows Forms está constituido por multitud de clases especializadas que ofrecen funcionalidades para el trabajo con ventanas, botones, menues, campos de texto y todo este tipo de controles habituales en las aplicaciones de escritorio.

Aplicaciones para MS Windows

- .NET Framework ofrece soporte extensivo para el desarrollo de aplicaciones para el sistema operativo MS Windows.
- El componente principal es Windows Forms = WinForms.
- La palabra “form” es prestada de Visual Basic y es analogía de desarrollo rápido de aplicaciones (RAD).
- El IDE de Visual Studio provee un diseñador para la creación de WinForms.
- WinForms permite crear la interfaz gráfica (GUI) de las aplicaciones para MS Windows. (Windows Forms Designer).
- Utiliza el namespace: System.Windows.Forms.
- Permite crear aplicaciones conectadas a base de datos: Cliente/Servidor o Distribuidas.
- Componentes:
 - Formularios (Forms)
 - Controles (Controls)
 - Componentes (Components)
 - Cuadros de diálogo (Dialog boxes)

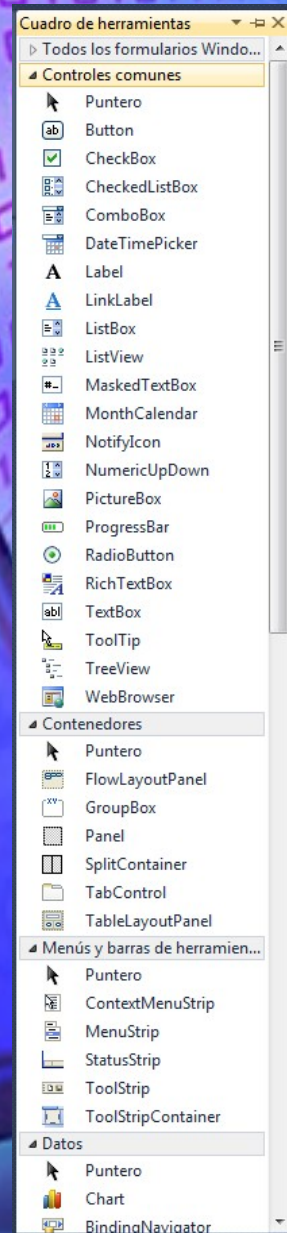
Componentes – Formularios

- Cada formulario es una ventana que puede ser utilizada para presentar información al usuario.
- Los formularios pueden tener cualquier forma, pero generalmente son rectangulares.
- Son objetos y tienen propiedades que definen su apariencia.
- Exponen métodos y eventos que definen su comportamiento e interacción con el usuario.
- Por defecto, hereda de la clase: `System.Windows.Forms.Form`.
- Se puede crear, heredando de otra clase formulario.



Componentes – Controles

- Son elementos visuales y forman la interfaz de usuario.
- Se agregan sobre la superficie de un formulario para definir la funcionalidad del mismo.
- Se dibujan ellos mismos sobre su contenedor.
- Se puede manipular sus propiedades, métodos y eventos para definir su apariencia y comportamiento.
- Se heredan desde `System.Windows.Forms.Control`.
- La barra de herramientas (Visual Studio) contiene muchos controles que pueden ser agregados a los formularios (en tiempo de diseño) como: `TextBox`, `Button`, `Label`, `Panel`, `ComboBox`, etc.
- Los controles también pueden ser creados y agregados a un formulario en tiempo de ejecución.

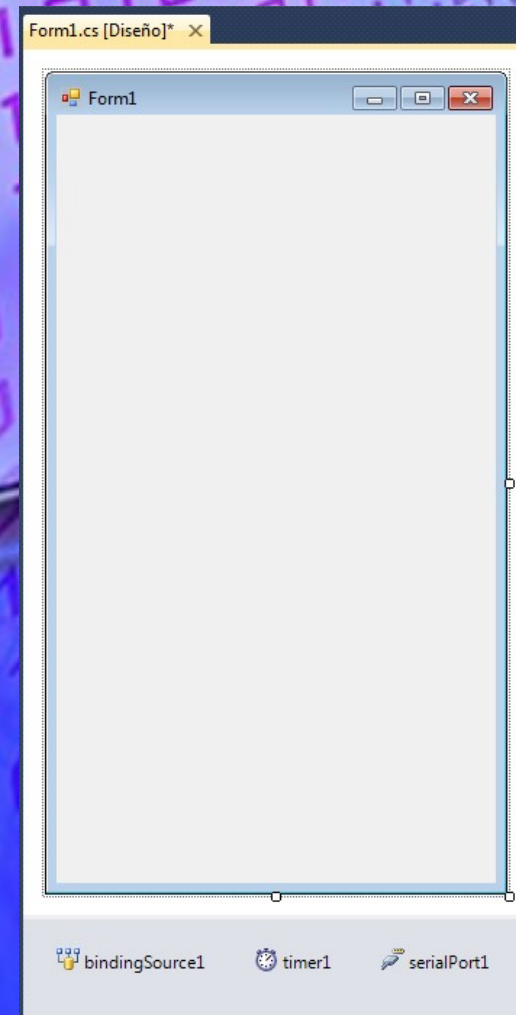


Controles

Función	Controles
Edición de texto	Textbox, RichTextBox, MaskedTextBox
Mostrar Texto	Label, LinkLabel, statusStrip
Seleccionar de una lista	CheckedListBox, ComboBox, DomainUpDown, ListBox, ListView, NumericUpDown, TreeView
Mostrar gráficos	PictureBox
Almacenar gráficos	ImageList
Ajustar valores	CheckBox, RadioButton, TrackBar
Mostrar contenido	DataGridView, WebBrowser
Ajustar fechas	DateTimePicker, MonthCalendar
Menues de control	MenuStrip, ContextMenuStrip
Comandos	Button, NotifyIcon, ToolStrip
Agrupadores	FlowLayoutPanel, GroupBox, Panel, TabControl, TableLayoutPanel, SplitContainer

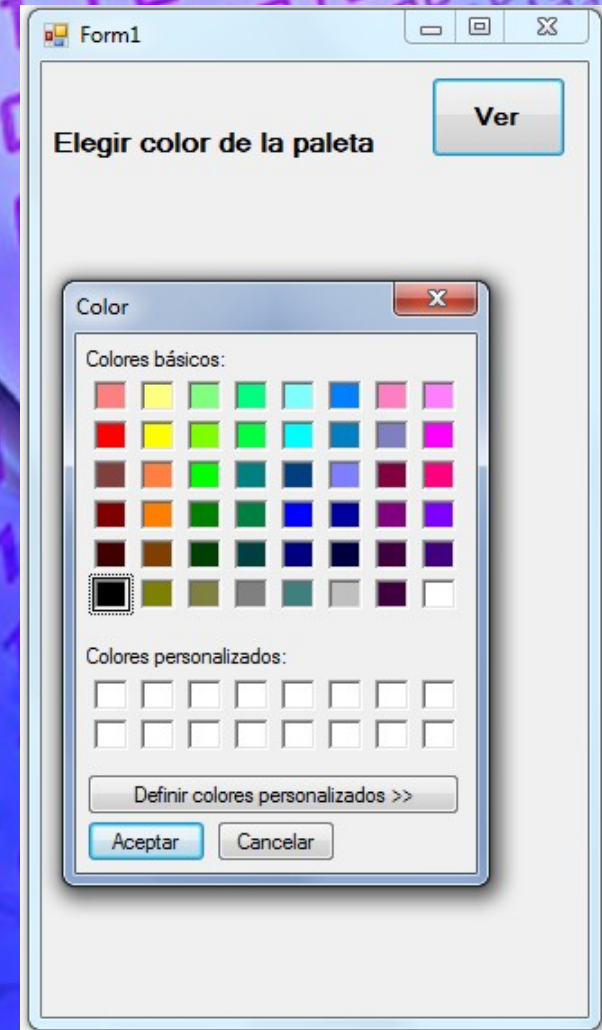
Componentes – Componentes

- Son elementos no-visuales.
- Permiten agregar funcionalidad adicional a las aplicaciones.
- Cuando se agregan componentes a un formulario, aparecen en la parte inferior, NO en el mismo formulario.
- Se puede manipular sus propiedades, métodos y eventos de la misma manera en que se manipulan los controles.
- Ejemplos: ImageList, BindingSource, BackgroundWorker, timer, serialPort, etc.
- Se heredan desde System.ComponentModel.Component.
- Un control es un componente con interfaz de usuario.



Componentes – Cuadros de dialogo

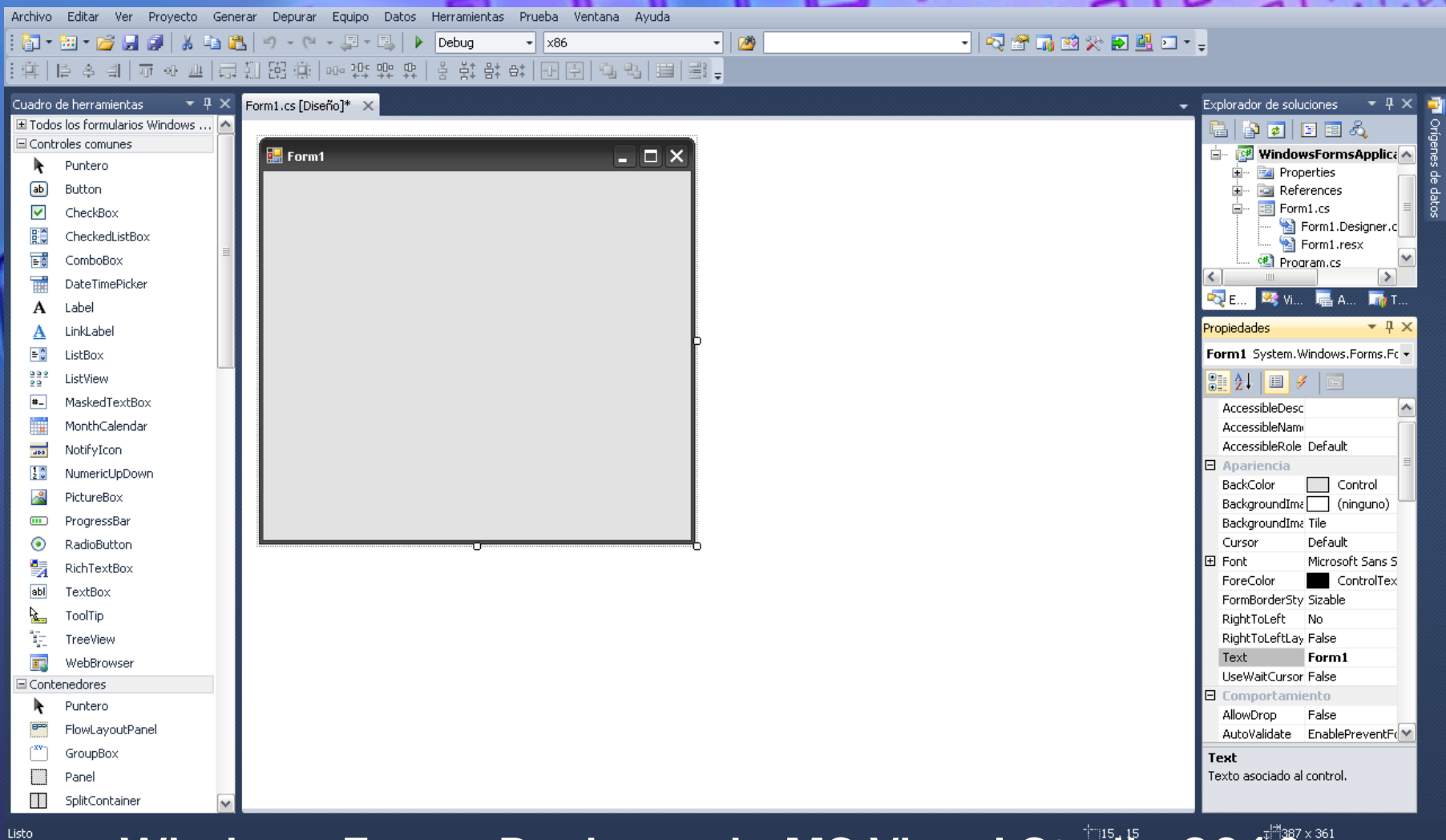
- Se utilizan para interactuar con el usuario y obtener información.
- Un cuadro de diálogo es un formulario que tiene un borde y tamaño predefinido.
- Siempre devuelven valores como resultado de la interacción con el usuario.
- Exponen la propiedad DialogResult para saber como fue cerrado el cuadro de diálogo.
- .NET Framework provee varios cuadros de diálogo predefinidos: OpenFileDialog, SaveFileDialog, ColorDialog, FontDialog, PrintDialog, PrintPreviewDialog.



Creación de aplicaciones

- Las aplicaciones tradicionales basadas en Microsoft Windows y las aplicaciones cliente-servidor se empiezan a crear mediante el diseño de los formularios utilizando el Diseñador de Windows Forms. Primero se asignan ciertas características a los formularios y se colocan controles en ellos y, a continuación, se escribe el código necesario para ampliar la funcionalidad de los controles y formularios. También es posible heredar de otros formularios.
- Utilizar el diseñador (Windows Forms Designer).
- Agregar todos los formularios necesarios para la aplicación.
- Asignar características a los formularios mediante sus propiedades.
- Agregar controles (drag and drop) a los formularios.
- Escribir código para capturar eventos y aumentar la funcionalidad de los controles y formularios.
- Puede heredar formularios y controles.
- Establecer la forma de ejecución de la aplicación windows.

Creación de aplicaciones

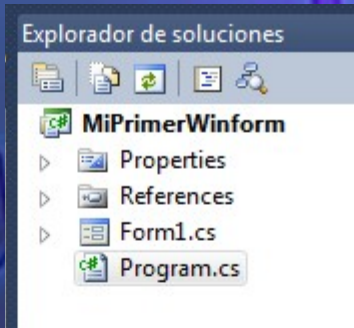


Windows Forms Designer de MS Visual Studio 2010

www.profmatiasgarcia.com.ar

Ejecución de aplicaciones

- Una aplicación Windows puede tener un número indefinido de formularios.
- La forma de ejecución se encuentra en el archivo Program.cs que es parte de la aplicación.
- Este archivo contiene el método Main() que define el punto de entrada para la aplicación.
- El formulario de inicio es controlado por la sentencia Application.Run(new Form1());
- Form1 puede ser cambiado por cualquier formulario que será el formulario inicial.



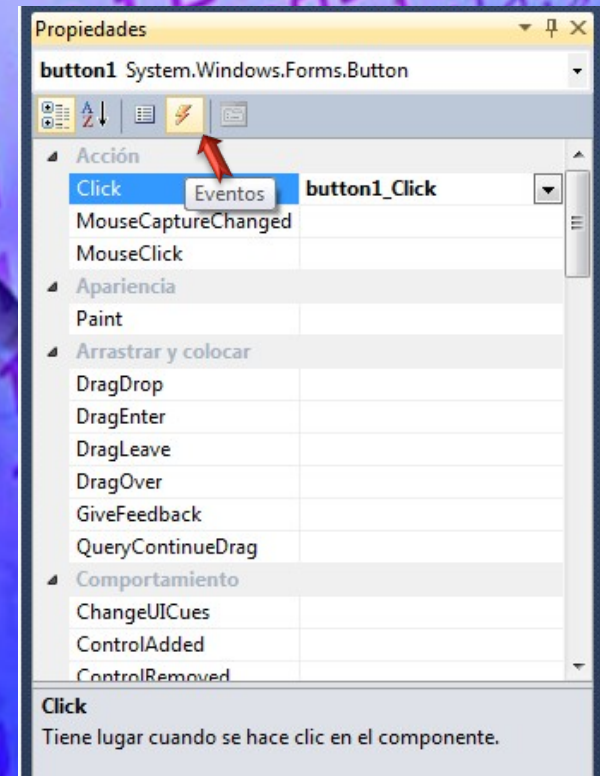
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows.Forms;

namespace MiPrimerWinform
{
    static class Program
    {
        /// <summary>
        /// Punto de entrada principal para la aplicación.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new FormularioInicio());
        }
    }
}
```


Eventos

- Los eventos son parte fundamental de las aplicaciones Windows.
- Se pueden interceptar los eventos y programar la lógica necesaria.
- Los eventos de los controles se pueden establecer en la ventana de Propiedades/Eventos.
- Cuando se utiliza la ventana de propiedades/eventos para crear un manejador de evento, se crea un delegado que referencia al método y lo suscribe al evento. Ejemplo:

```
this.clear.Click += new  
System.EventHandler(this.clearClick);
```



Notación Húngara

- Ésta es una lista de prefijos de control y objeto de Visual Basic proporcionado como referencia para el uso coherente de notación húngara. Esta lista ayudará a estandarizar la estructura, codificación de estilo y la lógica de una aplicación.

Prefijo	Control	Ejemplo
btn	Button (Botones)	btnSalir
cbo	ComboBox (Lista desplegable)	cboCiudad
chk	CheckBox (Casillas de verificación)	chkImpresora
dat	Data (Control de datos)	datBiblio
dbcbo	DataCombo (Lista despegable enlazada a datos)	dbcboFormato
dbgrd	DataGrid (Cuadrícula enlazada a datos)	dbgrdResultadoConsulta
dblst	DataList (Lista enlazada a datos)	dblstTipoTarea
dlg	Diálogo Común	dlgArchivoAbrir
frm	Form (Formulario)	frmEntrada
gbox	GroupBox (Marco)	gboxLenguaje
gra	Gráfico	graIngresos
grd	Grid (Cuadrícula)	grdPrecios
img	Image (Imagen)	imgBoton
lbl	Label (Etiqueta)	lblStatus
lst	ListBox (Lista)	lstNombres
mdi	MDIForm (Formulario MDI padre)	mdiNota
mnu	Menu	mnuArchivoAbrir
opt	RadioButton (Boton de opción)	optMasculino
pic	PictureBox (Cuadro para imagen)	picStatus
shp	Shape (Forma)	shpCirculo
txt	TextBox (Cajas de texto)	txtNombre

Webgrafía & Licencia:

- Textos tomados, corregidos y modificados de diferentes páginas de Internet, tutoriales y documentos.
- Este documento se encuentra bajo Licencia Creative Commons 2.5 Argentina (BY-NC-SA), por la cual se permite su exhibición, distribución, copia y posibilita hacer obras derivadas a partir de la misma, siempre y cuando se cite la autoría del **Prof. Matías E. García** y sólo podrá distribuir la obra derivada resultante bajo una licencia idéntica a ésta.
- Autor:

Matías E. García

Prof. & Tec. en Informática Aplicada

www.profmatiasgarcia.com.ar

info@profmatiasgarcia.com.ar



www.profmatiasgarcia.com.ar