

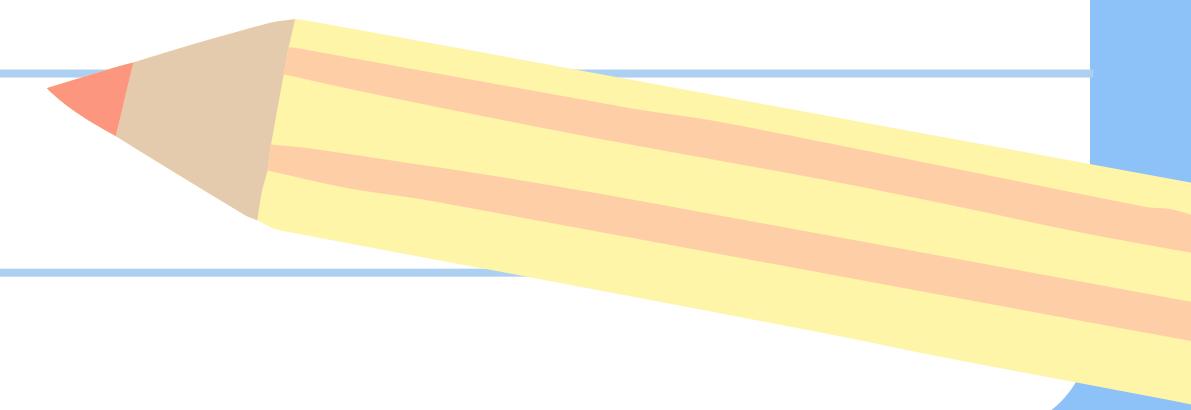
**¿QUE TIPO DE
APLICACIONES SE
PUEDEN HACER CON
WINDOWS FORMS DE
VISUAL STUDIO**

NOMBRE DEL ALUMNO: Luis Carlos Acosta Álvarez.

INTRODUCCIÓN

Con Visual Studio y la tecnología Windows Forms, puedes crear una amplia variedad de aplicaciones de escritorio que interactúan con el usuario a través de una interfaz gráfica. Estas aplicaciones pueden realizar diversas tareas, como:

- Gestión de datos:
- Aplicaciones para trabajar con bases de datos, almacenar y recuperar información, realizar cálculos, etc.
- Software de gestión:
- Aplicaciones para gestionar clientes, inventarios, tareas, proyectos, entre otros.
- Herramientas de usuario:
- Aplicaciones para realizar tareas específicas, como edición de texto, procesamiento de imágenes, reproductor de medios, etc.
- Aplicaciones de utilidad:
- Herramientas para automatizar procesos, realizar cálculos, convertir formatos, etc.
- Aplicaciones de comunicación:
- Aplicaciones para interactuar con usuarios a través de redes, compartir información, etc.



INTRODUCCIÓN

En esencia, con Windows Forms puedes crear cualquier aplicación que requiera una interfaz gráfica de usuario para interactuar con el usuario de una forma visual y eficiente. [GeeksforGeeks](#) menciona: Visual Studio es un entorno de desarrollo integrado (IDE) que te permite diseñar la interfaz gráfica de usuario de manera sencilla y eficiente. Puedes arrastrar y soltar controles (botones, cuadros de texto, listas, etc.) para construir la interfaz, y luego escribir el código para que la aplicación realice las acciones necesarias.

Para más detalles y ejemplos, puedes consultar la documentación oficial de [Windows Forms en Learn Microsoft](#) y la documentación de Visual Studio en Microsoft.



EJEMPLO:

Crea un Nuevo Proyecto:

Abre Visual Studio.

En la ventana de inicio, haz clic en "Crear un proyecto nuevo".

Busca o selecciona la plantilla "Aplicación de Windows Forms (.NET Framework)" (asegúrate de seleccionar la versión .NET Framework, ya que es más común para este tipo de ejemplos).

Haz clic en "Siguiente".

Elige un nombre para tu proyecto (por ejemplo, "MiAplicacion").

Selecciona una ubicación para guardar el proyecto y haz clic en "Crear".

Diseña la Portada (Formulario Principal - Form1.cs):

En el Explorador de soluciones (a la derecha), haz doble clic en Form1.cs para abrir el diseñador.

Agrega Controles: Desde el Cuadro de herramientas (a la izquierda), arrastra y suelta los siguientes controles al formulario:

Un Label para el título de la aplicación (por ejemplo, "Mi Fantástica Aplicación").

Otro Label para un subtítulo o alguna información adicional (opcional).

Un Button con el texto "Iniciar".

Personaliza los Controles:

Selecciona cada control y ve a la ventana de Propiedades (generalmente debajo del Explorador de soluciones).

Modifica las propiedades más importantes:

Text: Cambia el texto que se muestra en el control (por ejemplo, el texto del botón a "Iniciar").

Font: Cambia el tipo de letra, el tamaño y el estilo. Haz que el título sea más grande y llamativo.

TextAlign (para Labels): Ajusta la alineación del texto (por ejemplo, a MiddleCenter).

Size: Cambia el tamaño de los controles y del formulario para que se vean bien.

StartPosition (para el Formulario): Cambia a CenterScreen para que la aplicación se abra en el centro de la pantalla.

FormBorderStyle (para el Formulario): Podrías cambiarlo a FixedSingle o FixedDialog si no quieres que el usuario pueda redimensionar la portada.

Agrega un Nuevo Formulario para la Aplicación Principal (Form2.cs):

En el Explorador de soluciones, haz clic derecho en el nombre de tu proyecto ("MiAplicacion").

Selecciona "Agregar" -> "Nuevo elemento...".

En la ventana "Agregar nuevo elemento", selecciona "Windows Forms" y cambia el nombre a Form2.cs. Haz clic en "Agregar".

Diseña Form2 como quieras. Por ejemplo, podrías agregar otro Label con un mensaje de bienvenida.

Agrega Código para el Botón "Iniciar" en Form1.cs:

Haz doble clic en el botón "Iniciar" que agregaste a Form1. Esto abrirá el editor de código (Form1.cs) y generará automáticamente un evento Click para el botón.

Dentro de este evento (entre las llaves {}), agrega el siguiente código para abrir Form2 y cerrar Form1:

C#

```
private void button1_Click(object sender, EventArgs e)
{
    Form2 formularioPrincipal = new Form2();
    formularioPrincipal.Show(); // Muestra Form2
    this.Hide(); // Opcional: Oculta Form1 en lugar de cerrarlo
    // Si quieras cerrar Form1 completamente:
    // this.Close();
}
```

5. Agrega Código para Cerrar la Aplicación desde `Form2.cs` (Opcional):

- Si ocultaste `Form1` en lugar de cerrarlo, es posible que quieras un botón en `Form2` para cerrar toda la aplicación.
- Agrega un `Button` a `Form2`.
- Haz doble clic en el botón y agrega el siguiente código al evento `Click`:

C#

```
private void buttonCerrar_Click(object sender, EventArgs e)
{
    Application.Exit();
}
```

6. Ejecuta la Aplicación:

- Presiona `F5` o haz clic en el botón "Iniciar" (el triángulo verde) en la barra de herramientas de Visual Studio.
- Deberías ver la portada (`Form1`).
- Haz clic en el botón "Iniciar" para abrir el segundo formulario (`Form2`).

Código Completo (Ejemplo):

`Form1.cs` (Código detrás de la portada):

C#



```
using System;
using System.Windows.Forms;

namespace MiAplicacion
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            Form2 formularioPrincipal = new Form2();
            formularioPrincipal.Show();
            this.Hide();
        }
    }
}
```

```
namespace MiAplicacion
{
    partial class Form1
    {
        /// <summary>
        /// Variable del diseñador necesaria.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Limpiar los recursos que se estén usando.
        /// </summary>
        /// <param name="disposing">true si los recursos administrados se deben desechar;
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Código generado por el Diseñador de Windows Forms

        /// <summary>
        /// Método necesario para admitir el Diseñador. No se puede modificar
        /// el contenido de este método con el editor de código.
        /// </summary>
        private void InitializeComponent()
        {
            this.labelTitulo = new System.Windows.Forms.Label();
            this.buttonIniciar = new System.Windows.Forms.Button();
            this.labelSubtitulo = new System.Windows.Forms.Label();
            this.SuspendLayout();
            // 
            // labelTitulo
            // 
            this.labelTitulo.AutoSize = true;
            this.labelTitulo.Font = new System.Drawing.Font("Microsoft Sans Serif", 24F,
            this.labelTitulo.Location = new System.Drawing.Point(50, 50);
            this.labelTitulo.Name = "labelTitulo";
        }
    }
}
```

**¡GRACIAS POR
SU ATENCIÓN!**