

Tecnicatura Universitaria en Programación

Programación 2

Plan 2024

Eje temático N° 1 - Guía Práctica

Windows Forms

Ejercicio 1

Objetivo

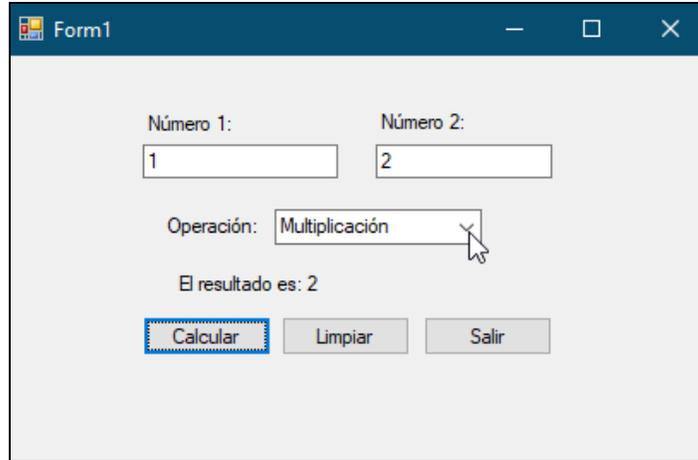
Desarrollar una aplicación de Windows Forms en C# que permita al usuario realizar operaciones aritméticas básicas entre dos números. La aplicación debe proporcionar un `TextBox` para cada número, un `ComboBox` para seleccionar la operación deseada (suma, resta, multiplicación, división), y un `Label` para mostrar el resultado. Además, la aplicación debe incluir botones para limpiar los campos y para cerrar la aplicación.

Instrucciones

1. Crear un formulario con dos `TextBox` para que el usuario ingrese los números.
2. Añadir un `ComboBox` que permita seleccionar una operación aritmética (suma, resta, multiplicación, división).
3. Incluir un `Label` que mostrará el resultado de la operación.
4. Añadir un `Button` para ejecutar la operación seleccionada y mostrar el resultado en el `Label`.
5. Incluir un `Button` para limpiar los `TextBox` y el `Label`.
6. Añadir un `Button` para cerrar la aplicación.

Detalles

- Al hacer clic en el botón de ejecutar, el resultado de la operación seleccionada debe mostrarse en el `Label`.
- Al hacer clic en el botón de limpiar, se deben vaciar ambos `TextBox` y el `Label`.
- Al hacer clic en el botón de cerrar, la aplicación debe finalizar.



Ejercicio 2

Objetivo

Desarrollar una aplicación de Windows Forms en C# que permita al usuario ingresar y gestionar datos de un estudiante. La aplicación debe incluir campos para ingresar el apellido, nombre, año de ingreso y legajo del estudiante, así como opciones para seleccionar la carrera (Ingeniería o Tecnicatura) y las tecnologías conocidas (C#, HTML, Javascript). La interfaz debe contar con botones para cargar, finalizar la carga y mostrar los datos ingresados, con comportamientos específicos para habilitar y deshabilitar controles.

Instrucciones

1. Crear un formulario con los siguientes controles:
 - a. `TextBox` para ingresar el apellido del estudiante.
 - b. `TextBox` para ingresar el nombre del estudiante.
 - c. `MaskedTextBox` para ingresar el año de ingreso del estudiante.
 - d. `MaskedTextBox` para ingresar el legajo del estudiante.
 - e. `RadioButtons` para seleccionar la carrera (Ingeniería o Tecnicatura).
 - f. `CheckBoxes` para seleccionar las tecnologías conocidas (C#, HTML, Javascript).
2. Añadir un `Button` denominado "Cargar Datos" que, al hacer clic, deshabilite dicho botón y habilite todos los demás controles.
3. Añadir un `Button` denominado "Finalizar Carga" que, al hacer clic, deshabilite todos los controles excepto los botones "Cargar Datos" y "Mostrar Datos".
4. Añadir un `Button` denominado "Mostrar Datos" que, al hacer clic, muestre en un `MessageBox` toda la información ingresada del estudiante.

5. Al iniciar la aplicación, todos los controles (excepto el botón "Cargar Datos") deben estar deshabilitados.
6. Al hacer clic en el botón "Cargar Datos", se deben habilitar todos los controles excepto el botón "Mostrar Datos".
7. Al hacer clic en el botón "Finalizar Carga", se deben deshabilitar todos los controles excepto los botones "Cargar Datos" y "Mostrar Datos".
8. Al hacer clic en el botón "Mostrar Datos", se debe mostrar un `MessageBox` con toda la información del estudiante.

Detalles

- La información del estudiante debe incluir el apellido, nombre, año de ingreso, legajo, carrera seleccionada y tecnologías conocidas.
- El `MessageBox` debe mostrar todos estos datos de manera clara y organizada.

Alumno: Fernando Cardona

Cargar Datos

Apellido

Nombre

Año Ingreso

Legajo

Carrera Ingeniería Tecnicaura

Tecnologías que conozco C# HTML Javascript

Finalizar Carga

Mostrar Datos

Salir

Ejercicio 3

Objetivo

Desarrollar una aplicación de Windows Forms en C# que permita al usuario calcular el salario semanal de un empleado basado en las horas trabajadas. La aplicación debe permitir al usuario ingresar la cantidad de horas trabajadas y mostrar el salario calculado. La interfaz debe

incluir un campo para la entrada de datos, un botón para realizar el cálculo y otro botón para limpiar los campos.

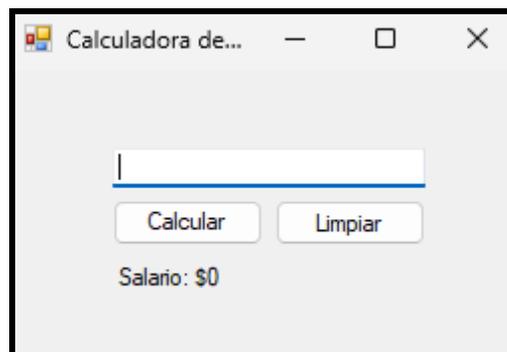
Instrucciones

Crear un formulario con los siguientes controles:

- Un `TextBox` para ingresar la cantidad de horas trabajadas en una semana.
- Un `Label` para mostrar el salario semanal calculado.
- Un `Button` para realizar el cálculo del salario.
- Un `Button` para limpiar los campos.
- Configurar el `TextBox` para que el usuario pueda ingresar el número de horas trabajadas.
- Al hacer clic en el botón de cálculo:
 - Leer la cantidad de horas ingresadas.
 - Calcular el salario semanal según las siguientes reglas:
 - Si el número de horas es de 40 o menos, el salario es \$150 por hora.
 - Si el número de horas es mayor a 40, las horas extras (más de 40) se pagan a \$250 por hora.
- Mostrar el salario calculado en el `Label`.
- Al hacer clic en el botón de limpiar:
 - Vaciar el contenido del `TextBox` y del `Label`.

Detalles

- El cálculo del salario debe considerar las horas normales y las horas extras.
- El botón de cálculo debe activar el cálculo y mostrar el resultado en el `Label`.
- El botón de limpiar debe restablecer el formulario, vaciando el `TextBox` y el `Label`.



Ejercicio 4

Objetivo

Desarrollar una aplicación de Windows Forms en C# para una agencia de seguros de automóviles que calcula el precio de la póliza basado en varios factores. La aplicación debe incluir labels para mostrar los aranceles aplicables y cambiar de color según el monto de la póliza que debe abonar. El formulario debe mostrar el nombre, la edad, el sexo del conductor y el monto de la póliza calculado.

Instrucciones

Formulario de Entrada:

- `TextBox` para ingresar el nombre del conductor.
- `NumericUpDown` para ingresar la edad del conductor.
- `RadioButtons` para seleccionar el sexo del conductor (Masculino/Femenino).
- `ComboBox` para seleccionar el tipo de vehículo (Sedán, SUV, Deportivo).
- `NumericUpDown` para ingresar los años de experiencia de manejo.

Aranceles y Labels:

- Añadir los siguientes `Labels` para mostrar los aranceles aplicables:
 - `Label` para el arancel base según sexo y edad.
 - `Label` para el ajuste por tipo de vehículo.
 - `Label` para el ajuste por experiencia de manejo.
- Inicialmente, estos `labels` deben tener un color gris de fondo.

Reglas de Cálculo del Precio de la Póliza:

- **Sexo y Edad:**
 - Hombres menores de 25 años pagan \$1000.
 - Hombres de 25 años o más pagan \$700.
 - Mujeres menores de 21 años pagan \$800.
 - Mujeres de 21 años o más pagan \$1200.
- **Tipo de Vehículo:**
 - Sedán: \$0 (sin ajuste).
 - SUV: +\$200.
 - Deportivo: +\$400.
- **Experiencia de Manejo:**
 - Menos de 2 años: +\$300.
 - De 2 a 5 años: \$0 (sin ajuste).

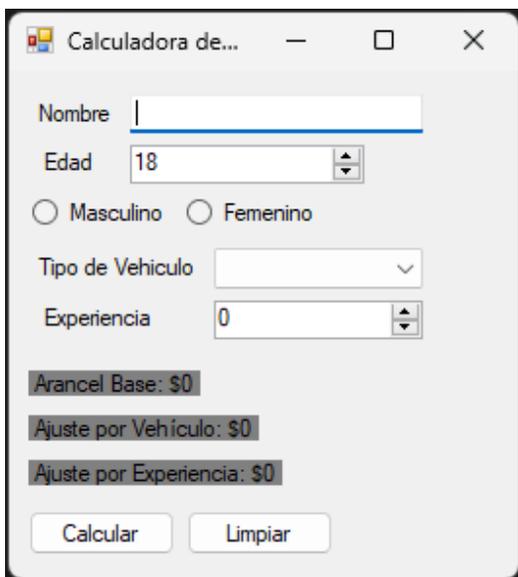
- Más de 5 años: -\$150.

Botones y Funcionalidades:

- Añadir un **Button** para calcular el monto de la póliza.
- Añadir un **Button** para limpiar los campos.
- Al hacer clic en el botón de cálculo:
 - Validar que todos los campos han sido completados correctamente.
 - Calcular el monto de la póliza basado en las reglas proporcionadas.
 - Cambiar el color de los labels para reflejar el arancel aplicable:
 - El **label** del arancel base debe cambiar a verde si es el monto calculado.
 - El **label** del ajuste por tipo de vehículo debe cambiar a verde si se aplica un ajuste.
 - El **label** del ajuste por experiencia de manejo debe cambiar a verde si se aplica un ajuste.
 - Mostrar el nombre, la edad, el sexo del conductor y el monto total de la póliza en un **MessageBox**.
- Al hacer clic en el botón de limpiar:
 - Vaciar todos los campos del formulario.
 - Restablecer los colores de los labels a gris.

Validaciones Adicionales:

- Verificar que la edad sea un número positivo.
- Asegurarse de que se haya seleccionado un sexo y un tipo de vehículo.
- Confirmar que los años de experiencia de manejo sean un número no negativo.



Calculadora de...

Nombre |

Edad 18

Masculino Femenino

Tipo de Vehiculo

Experiencia 0

Arancel Base: \$0

Ajuste por Vehiculo: \$0

Ajuste por Experiencia: \$0

Calcular Limpiar