

GUIA PRACTICA DIRIGIDA

Competencia:

El desarrollo de esta guía práctica te permitirá conocer cómo se puede obtener el promedio final de un curso obedeciendo a las reglas de negocio establecida para su cálculo (de acuerdo a la institución donde se aplica).

Descripción:

La guía se basa en el proyecto **ProyWinC_Sem01**. La guía se contempla lo siguiente:

- a) La entrega de 3 archivos (que componen el formulario Form3) los cuales serán acoplados al proyecto en mención.
- b) El enunciado correspondiente (el pdf actual) para obtener el promedio, el cual se deberá codificar según se indica líneas más adelante.

Todo esto será compartido por su instructor en el ISIL + en un archivo zip para que sea desarrollado durante la sesión. Se tendrá un tiempo de 20 minutos.

Planteamiento del problema:





En el instituto “León Learning” se dictan cursos sobre TI. Son cursos cortos que contienen solo 3 evaluaciones. Los cursos son de 2 tipos: Teóricos y Prácticos. Se requiere elaborar una interface que permita el ingreso de las 3 notas en cajas de texto y la selección del tipo de curso mediante botones de opción. Tras el ingreso de las notas se debe, mediante un botón de comando, mostrar en una etiqueta el promedio obtenido, de acuerdo a la siguiente regla de negocio:

- a) Para cursos teóricos, el promedio se calcula tomando en cuenta las 3 notas (promedio simple).
- b) Para el caso de cursos prácticos el promedio se calcula eliminando la nota más baja. En este caso la aplicación debe mostrar en un mensaje cuál de las 3 notas es la que va a eliminar.
- c) Para cualquiera de los 2 casos la nota mínima aprobatoria es 13. Si el promedio es aprobatorio deberá mostrarse en color azul, de lo contrario en color rojo.
- d) No se consideran por ahora validaciones de notas en cuanto a sus rangos de 0 a 20, pero si se debe manejar la excepción en caso se ingresen valores incorrectos, por ejemplo, ingresar valores textuales como notas, enviando un mensaje de error si fuera este el caso.

Procedimiento:

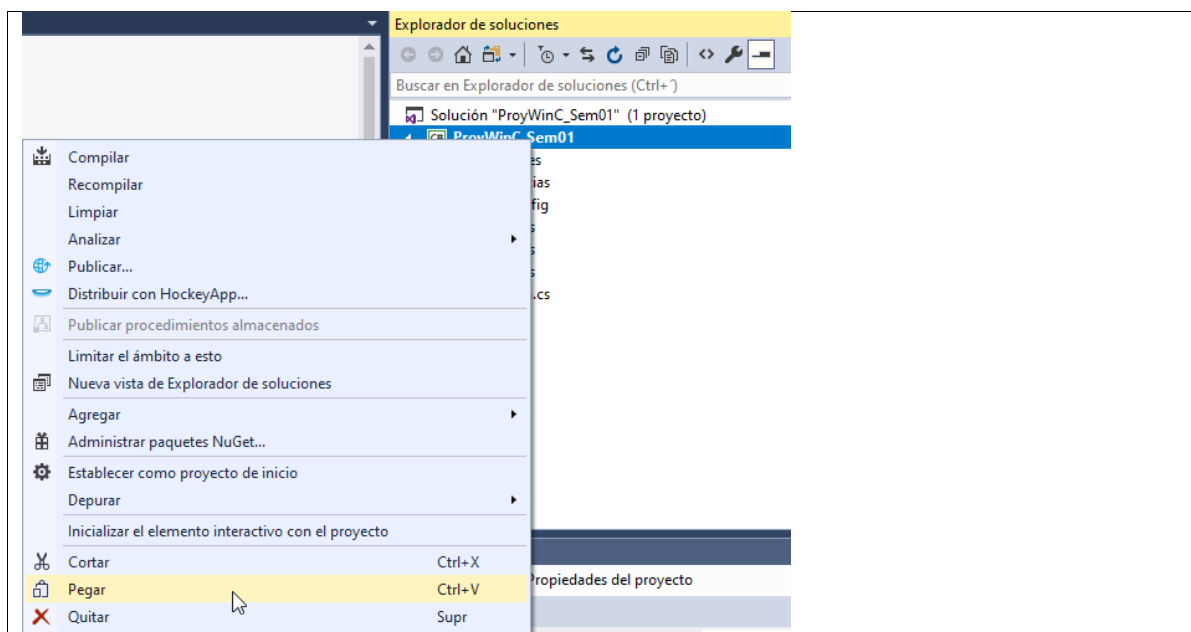
Paso 1:

Descargue la guía “Guía Práctica – Calculo promedio” desde el ISIL +. El archivo zip desempaquetelo en la carpeta de su preferencia. Se encontrarán los siguientes archivos:

Nombre	Fecha de modificación	Tipo	Tamaño
 Form3.cs	11/01/2021 01:13	Visual C# Source F...	3 KB
 Form3.Designer.cs	7/04/2020 22:50	Visual C# Source F...	9 KB
 Form3.resx	7/04/2020 22:50	Microsoft .NET M...	6 KB
 GUIA PRACTICA - Calculo Promedios	13/04/2021 21:04	Adobe Acrobat D...	721 KB

Paso 2:

Abra el proyecto **ProyWinC_Sem01**. Seleccione desde la carpeta donde hizo la descarga los 3 archivos que componen el formulario Form3. Seleccionados los 3 pulse “Control+C” o clic derecho y “Copiar”. La idea es clara: copiar los 3 archivos al proyecto. Indicada la orden de copiar, vaya al proyecto **ProyWinC_Sem01** y en la ventana del Explorador de Soluciones colóquese en el nombre del proyecto, haga clic derecho y seleccione “Pegar”. Vera como se adiciona al proyecto el formulario Form3.



El diseño del formulario Form3 es el siguiente:

Las cajas de texto se llaman **txtN1**, **txtN2** y **txtN3**. En ellas se ingresarán las notas. Los botones de opción se llaman **optTeoria** y **optPractica** y permitirán elegir el tipo de curso. El botón **btnCalcular** permitirá efectuar el cálculo del promedio, el cual será mostrado en la etiqueta de resultado llamada **lblPromedio**.

Paso 3:

Haga doble clic en el botón etiquetado como Calcular (btnCalcular) y vera que la plantilla le genera el evento clic del botón btnCalcular. En dicho evento debe codificar lo siguiente:

```
private void btnCalcular_Click(object sender, EventArgs e)
{
    // Creamos un vector de tipo Single de 3 posiciones
    Single[] notas = new Single[3];
    // Una variable para almacenar el promedio calculado
    Single promedio = 0;

    // Guardamos las 3 notas en el vector
    notas[0] = Convert.ToSingle(txtN1.Text.Trim());
    notas[1] = Convert.ToSingle(txtN2.Text.Trim());
    notas[2] = Convert.ToSingle(txtN3.Text.Trim());

    // Ordenamos el vector en forma ascendente, de tal forma que en la posicion
    // 0 del vector quede la nota mas baja

    Array.Sort(notas);

    // Preguntamos por el tipo de curso
    if (optTeoria.Checked == true)
    {
        // Si es de teoria calculamos en funcion de las 3 notas
        promedio = (notas[0] + notas[1] + notas[2]) / 3;
    }
}
```

```
else
{
    //Si es de practica primero se muestra la nota a eliminar , la mas baja, es decir
    // la que esta en la posicion 0 del vector notas
    MessageBox.Show("Se elimino la nota :" + notas[0], "Mensaje");
    // Y calculamos el promedio tomando las 2 notas mas altas ubicadas
    // en la posicion 1 y 2 del vector notas.
    promedio = (notas[1] + notas[2]) / 2;
}

// Mostramos el promedio
lblPromedio.Text = promedio.ToString();

// Si el promedio es aprobatorio se pinta en azul de lo contrario en rojo
if (promedio >= 13)
{
    lblPromedio.ForeColor = Color.Blue;
}
else
{
    lblPromedio.ForeColor = Color.Red;
}
```

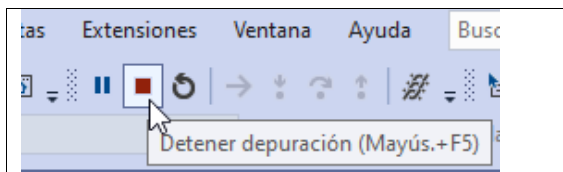
Paso 4:

Terminado de codificar haga las pruebas del caso, ingresando notas validas (numéricas entre 0 y 20). No olvide indicar que el Form3 sea el formulario de inicio en la clase Program.

Paso 5:

Por último, si hace una prueba e ingresa como nota 1 el valor “Pepe”, y 2 notas correctas más, vera que al calcular el promedio la aplicación entrara en un estado de depuración (stand bye) dado que el valor “Pepe” es un valor textual (String) y no podemos almacenarlo en un vector de valores numéricos reales (Single).

Detenga la ejecución de la aplicación con el botón **Detener depuración** ubicado en la parte central de su barra de herramientas:



Procederemos a colocar el código del evento clic del botón btnCalcular, dentro de una estructura de tipo try con lo que cualquier línea de código que genere una excepción (error) será capitalizada dentro del bloque catch, que colocaremos al final, tal como se muestra:

```
private void btnCalcular_Click(object sender, EventArgs e)
{
    try
    {
        // Creamos un vector de tipo Single de 3 posiciones
        Single[] notas = new Single[3];
        // Una variable para almacenar el promedio calculado
        Single promedio = 0;
    }
```

```
    else
    {
        lblPromedio.ForeColor = Color.Red;
    }
}

catch (Exception ex)
{
    MessageBox.Show("Error..." + ex.Message, "Error");
}
```

Compile el proyecto, y proceda hacer pruebas del caso. Ahora, con el bloque try-catch, si se ingresan valores textuales como nota, la aplicación no entrara en modo depuración, sino que se enviara un mensaje indicando que el tipo de dato proporcionado como nota no tiene el formato correcto. La instancia de excepción **ex** y la propiedad **message** de la misma, es la que proporciona el texto del mensaje.

¡Bien jugado!