

Universidad Abierta Para Adultos



Educación superior a distancia virtual

Formación General

CARRERA

Ingeniería en Software

ASIGNATURA

Programación II

FACILITADOR

Diógenes Amaury Martínez Silverio

PARTICIPANTE

Vladimir Núñez Calderón

MATRICULA

100057484

TEMA

Semana V

FECHA

20 de febrero del 2023

Santiago de los Caballeros, República Dominicana

INTRODUCCION

El DataGridView de C# es un control de interfaz de usuario utilizado para mostrar datos en forma de tabla. Es fácil de usar y se puede personalizar para satisfacer las necesidades del usuario. Permite la edición de datos en línea y la navegación rápida a través de grandes conjuntos de datos. En este informe se describirá la estructura, forma de uso, ventajas y desventajas del DataGridView en C#.

INVESTIGAR EN LA WEB ACERCA DEL COMPONENTE DATAGRIDVIEW, ESTRUCTURA, FORMA DE USO, VENTAJAS Y DESVENTAJAS

El componente C# DataGridView es un control de interfaz de usuario para mostrar datos en formato tabular. Consta de encabezados de columna, encabezados de fila, celdas que contienen datos y barras de desplazamiento. Se puede personalizar configurando propiedades como color de fondo, fuente, estilo de borde, etc.

Para usar un DataGridView en C#, debe crear una instancia del control y vincular los datos mediante una fuente de datos. Proporciona edición en línea, estilos de tabla personalizados y navegación rápida de grandes conjuntos de datos.

Las ventajas incluyen su facilidad de uso, la personalización de la edición, la selección de varias filas y varias columnas y la capacidad de mostrar grandes conjuntos de datos.

Las desventajas incluyen un rendimiento limitado cuando se trabaja con grandes conjuntos de datos, la necesidad de configurar ciertas propiedades y la inadecuación para presentar datos complejos o jerárquicos.

Entrada de Diario

Cuenta Nombre Cuenta Tipo Monto

	Cuenta	Nombre Cuenta	Debito	Credito
*				

Total Debito Total Credito
 0 0

```

private void btnAgregar_Click(object sender, EventArgs e)
{
    float totalDebito, totalCredito;
    totalCredito = 0;
    totalDebito = 0;

    if(txtCuenta.Text == "" || txtMonto.Text == "" || txtNombreCuenta.Text == "" ||
       sltTipo.Text == "")
    {
        MessageBox.Show("Todos los campos son obligatorios");
        return;
    }

    if(sltTipo.Text == "Debito")
    {
        totalDebito = float.Parse(txtTotalDebito.Text) + float.Parse(txtMonto.Text);
        dgvEntradas.Rows.Add(txtCuenta.Text, txtNombreCuenta.Text, txtMonto.Text, "");
        txtTotalDebito.Text = totalDebito.ToString();
    }

    if (sltTipo.Text == "Credito")
    {
        totalCredito = float.Parse(txtTotalCredito.Text) + float.Parse(txtMonto.Text);
        dgvEntradas.Rows.Add(txtCuenta.Text, txtNombreCuenta.Text, "", txtMonto.Text);
        txtTotalCredito.Text = totalCredito.ToString();
    }

    txtMonto.Clear();
    txtCuenta.Clear();
    txtNombreCuenta.Clear();
    sltTipo.Text = "";
    txtCuenta.Focus();
}

```

Eliminar Fila

```
private void btnEliminarFila_Click(object sender, EventArgs e)
{
    int sumaCredito, sumaDebito;
    var debito =
    Convert.ToInt32(dgvEntradas.Rows[dgvEntradas.CurrentCell.RowIndex].Cells[2].Value != "" ?
    dgvEntradas.Rows[dgvEntradas.CurrentCell.RowIndex].Cells[2].Value : 0);
    var credito =
    Convert.ToInt32(dgvEntradas.Rows[dgvEntradas.CurrentCell.RowIndex].Cells[3].Value != "" ?
    dgvEntradas.Rows[dgvEntradas.CurrentCell.RowIndex].Cells[3].Value : 0);

    sumaCredito = int.Parse(txtTotalCredito.Text) - credito;
    sumaDebito = int.Parse(txtTotalDebito.Text) - debito;

    txtTotalDebito.Text = sumaDebito.ToString();
    txtTotalCredito.Text = sumaCredito.ToString();
    dgvEntradas.Rows.RemoveAt(dgvEntradas.CurrentCell.RowIndex);
}
```