

## PRESENTACION

<b>NOMBRE</b>	<i>Yoedri David</i>
<b>APELLIDO</b>	<i>García Sierra</i>
<b>MATRICULA</b>	<i>21-EISM-6-050</i>
<b>MATERIA</b>	<i>Programación Orientada a Objetos</i>
<b>MAESTRO</b>	<i>Frank Reyes</i>
<b>PRACTICA</b>	<i>Nota Estudiantil</i>
<b>FECHA</b>	<i>9/10/2024</i>

## NOTA ESTUDIANTIL

The screenshot shows a Windows application window titled "NOTAS Y CALIFICACIONES". The main title bar has a logo and the text "NOTAS Y CALIFICACIONES". The window contains a header "CALIFICACION ESTUDIANTIL" and a toolbar with buttons for "PROCESAR", "OTRA NOTA", "NUEVO", and "SALIR". The main area contains the following data:

		N# Registro	FECHA
MATRICULA	21-EISM-0-005	1	09/10/2024
MATERIA	Redes Inalambricas		
MAESTRO	Marlon Fereraz Perez		
PRIMER PARCIAL	20		
SEGUNDO PARCIAL	17		
PRACTICA	18		
EXAMEN FINAL	36		
NOTA FINAL	91		
CALIFICACION	A		

```
namespace Nota_practica
{
    public partial class FNOTAS : Form
    {
        int NOREGISTRO, PP, SP, P, EF, TN;

        public FNOTAS()
        {
            InitializeComponent();
        }

        private void txt_examenF_TextChanged(object sender, EventArgs e)
        {
            try
            {
                if (Convert.ToInt32(txt_examenF.Text) > 40)
                {
                    MessageBox.Show("ESCRIBA UN NUMERO MENOR O IGUAL A
40");
                }
            }
            catch { }
        }

        private void txt_primerP_TextChanged(object sender, EventArgs e)
```

```

{
    try
    {
        if (Convert.ToInt32(txt_primerP.Text) > 20)
        {
            MessageBox.Show("ESCRIBA UN NUMERO MENOR O IGUAL A
20");
            txt_examenF.Enabled = false;
        }
        else
        {
            txt_examenF.Enabled = true;
        }
    }
    catch { }
}

private void txt_segundoP_TextChanged(object sender, EventArgs e)
{
    try
    {
        if (Convert.ToInt32(txt_segundoP.Text) > 20)
        {
            MessageBox.Show("ESCRIBA EL NUMERO MENOR O IGUAL A
20");
            txt_primerP.Enabled = false;
            txt_examenF.Enabled = false;
        }
        else
        {
            txt_primerP.Enabled = true;
            txt_examenF.Enabled = true;
        }
    }
    catch { }
}

private void txt_pp_TextChanged(object sender, EventArgs e)
{
    try
    {
        if (Convert.ToInt32(txt_pp.Text) > 20)
        {
            MessageBox.Show("ESCRIBA UN NUMERO MENOR O IGUAL A
20");
            txt_segundoP.Enabled = false;
            txt_primerP.Enabled = false;
            txt_examenF.Enabled = false;
        }
        else
        {
            txt_segundoP.Enabled = true;
            txt_primerP.Enabled = true;
            txt_examenF.Enabled = true;
        }
    }
    catch { }
}

private void btn_otraN_Click(object sender, EventArgs e)
{
}

```

```

        LIMPIAR();
    }

    private void btn_procesar_Click(object sender, EventArgs e)
    {
        DECISION();
    }

    private void btn_nuevo_Click(object sender, EventArgs e)
    {
        NUEVO();
    }

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

    private void txt_primerP_KeyPress(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void txt_segundoP_KeyPress(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void txt_pp_KeyPress(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void txt_examenF_KeyPress(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void FNOTAS_Load(object sender, EventArgs e)
    {
        NOREGISTRO = 1;
        txt_registro.Text = Convert.ToString(NOREGISTRO);
        txt_registro.Enabled = false;
        txt_notaF.Enabled = false;
        txt_calificacion.Enabled = false;
        cb_matricula.Text = "Escoja la matrícula del estudiante";
        cb_materia.Text = "Escoja la materia deseada";
        cb_maestro.Text = "Escoja el Maestro";
    }

```

```

}

private void ENTRADAS()
{
    try
    {
        PP = Convert.ToInt32(txt_primerP.Text);
        SP = Convert.ToInt32(txt_segundoP.Text);
        P = Convert.ToInt32(txt_pp.Text);
        EF = Convert.ToInt32(txt_examenF.Text);
    }
    catch
    {
        MessageBox.Show("ENTRADA INVALIDA, INTENTE OTRA VEZ");
        txt_pp.Focus();
    }
}

private void DECISION()
{
    ENTRADAS();
    TN = (PP + SP + P + EF);
    txt_notaF.Text = Convert.ToString(TN);

    if (TN >= 90)
    {
        txt_calificacion.Text = "A";
    }
    else
        if (TN >= 80 & TN < 90)
    {
        txt_calificacion.Text = "B";
    }
    else
        if (TN >= 75 & TN < 80)
    {
        txt_calificacion.Text = "C";
    }
    else
        if (TN >= 70 & TN < 75)
    {
        txt_calificacion.Text = "D";
    }
    else
        if (TN >= 50 & TN < 70)
    {
        txt_calificacion.Text = "FE";
    }
    else
        if (TN < 50)
    {
        txt_calificacion.Text = "F";
    }
}

private void LIMPIAR ()
{
    NOREGISTRO = NOREGISTRO + 1;
    txt_registro.Text = Convert.ToString(NOREGISTRO);
    cb_matricula.Enabled = false;
    cb_materia.Text = "Escoja otra Materia";
    cb_maestro.Text = "Escoja otro Maestro";
    txt_pp.Text = "0";
}

```

```
        txt_segundoP.Text = "0";
        txt_pp.Text = "0";
        txt_examenF.Text = "0";
        txt_notaF.Text = "0";
        txt_calificacion.Text = "0";
        TN = 0;
    }

    private void NUEVO()
    {
        LIMPIAR();
        cb_matricula.Enabled = true;
        cb_matricula.Text = "Escoga otra matricula";
    }
}
```