

```
using BussinessLayer;
using Entities;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Security.Cryptography;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Gym
{
    public partial class RegistroPrimeraVez : Form
    {
        #region Instancias
        //entidades
        private Entities.Empleados _empleados;
        private readonly Personas _personas;

        //clases internas
        private readonly Restricciones _restricciones;
        private readonly MetodosGenerales _metodosGenerales;

        //Negocio
        private readonly BussinessEmpleados _bussinessEmpleados;
        private readonly BussinessPersonas _bussinessPersonas;
        #endregion

        #region Load
        public RegistroPrimeraVez()
        {
            InitializeComponent();
            _restricciones = new Restricciones();
            _metodosGenerales = new MetodosGenerales();
            _bussinessEmpleados = new BussinessEmpleados();
            _empleados = new Entities.Empleados();
            _bussinessPersonas = new BussinessPersonas();
            _personas = new Personas();
        }
        private void RegistroPrimeraVez_Load(object sender, EventArgs e)
        {
            Tipos_Documentos();
            Tipos_Sexos();
        }

        #endregion

        #region Variables

        private bool camposVacíos = true;
        private string clave = string.Empty;

        #endregion

        #region Métodos encapsulados

        private void VerificarCamposVacios()
```

```
{
    //Recorro todos los textbox para saber si están vacíos
    //Y si hay alguno vacío, lo notifico, porque no pueden
    //haber campos vacíos en este formulario.
    foreach (Control txt in this.Controls)
    {
        if (txt is TextBox box)
        {
            TextBox t;
            t = box;

            //verificamos si algún txt está vacío
            if (string.IsNullOrEmpty(t.Text))
            {
                camposVacíos = true;
                break;
            }
            else
            {
                camposVacíos = false;
            }
        }
    }
}

private void AltaPersona()
{
    //Pasamos los datos a la entidad Personas
    _personas.Nombre = txtNombre.Text.ToString();
    _personas.Apellido = txtApellido.Text.ToString();
    _personas.Tipo_Documento_ID = cmbTipoDocumento.SelectedIndex;
    _personas.Nro_documento = txtNroDocumento.Text.ToString();
    _personas.Tipo_Sexo_ID = cmbSexo.SelectedIndex;
    _personas.Nro_Telefono = txtTelefono.Text.ToString();
    _personas.Fecha_Alta = DateTime.Now;
    _personas.Mail = txtMail.Text.ToString();

    //Y la enviamos a la capa de negocio para llevarla a la bdd.
    _bussinessPersonas.AltaJefe(_personas);
}

private void AltaEmpleado()
{
    //Primero obtenemos el último ID de la persona registrada
    _metodosGenerales.GetLastID();
    //luego ese ID lo asignamos a la entidad Empleados
    _empleados.Persona_ID = _metodosGenerales.persona_ID;
    _empleados.Tipo_Empleado_ID = 1;
    _empleados.Estado_Empleado_ID = 0;
    _empleados.Usuario = txtUsuario.Text.ToString();

    //Encriptamos la clave con un HASH256
    clave = Convert.ToString(txtClave.Text);
    //Y la asignamos al campo de la entidad correspondiente
    _empleados.Clave = EncriptamientoSHA256.GetSHA256(clave);

    //Envío toda la data a la capa de negocio para ser mandada a la bdd.
    _bussinessEmpleados.AltaEmpleado(_empleados);
}
```

```

private void Tipos_Documentos()
{
    //llamo al método para traer los tipos de documentos y los asigno al combobox
    _metodosGenerales.Bring_Tipos_Documentos();
    cmbTipoDocumento.DataSource = _metodosGenerales.DtTipos_Documentos;
    cmbTipoDocumento.DisplayMember = "Tipo";
    cmbTipoDocumento.ValueMember = "Tipo_documento_ID";
}

private void Tipos_Sexos()
{
    //llamo al método para traer los tipos de sexos y los asigno al combobox
    _metodosGenerales.Bring_Tipos_Sexos();
    cmbSexo.DataSource = _metodosGenerales.DtTipos_Sexos;
    cmbSexo.DisplayMember = "Sexo"; //Pero solo esta columna es la que muestro
    cmbSexo.ValueMember = "Tipo_Sexo_ID";
}

#endregion

#region Eventos KeyPress

//Tengo un método encapsulado en la clase Restricciones para los campos
//que quiero que sean numéricos.
//Así que creo una variable, le asigno la tecla presionada
//y si no es numérico, lo bloquea.
//Es una clase general, xq todos los controles de los forms, que son numéricos
//utilizan el mismo método.
//Para no crear nuevamente el método en cada formulario.
private void txtNroDocumento_KeyPress(object sender, KeyPressEventArgs e)
{
    string strTexto = txtNroDocumento.Text;
    _restricciones.SoloNumeros(e, strTexto);
}
private void txtTelefono_KeyPress(object sender, KeyPressEventArgs e)
{
    string strTexto = txtTelefono.Text;
    _restricciones.SoloNumeros(e, strTexto);
}

#endregion

#region Eventos Botones

private void lblCancelar_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    VerificarCamposVacios();
    if (!camposVacios)
    {
        DialogResult cancelar = MessageBox.Show("Faltan campos sin rellenar. ¿Seguro
que quiere cancelar la operación? s" +
                                                "En caso de seguir con el formulario,
presione Cancelar. Caso contrario, presione Aceptar",
                                                "Datos incompletos",
                                                MessageBoxButtons.OKCancel);
        if (cancelar == DialogResult.OK)
        {
            this.Close();
        }
    }
    else
    {

```

```
        this.Close();
    }
}
private void btnAceptar_Click(object sender, EventArgs e)
{
    //Verificamos campos vacíos
    VerificarCamposVacios();
    if (camposVacios)
    {
        //Se solicita que llene todos los campos
        MessageBox.Show("Para poder dar de alta el registro, deben completarse todos
los campos.",
                        "Campos incompletos",
                        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    }
    else
    {
        //Se da el alta del empleado
        AltaPersona();
        AltaEmpleado();
        DialogResult result = MessageBox.Show("Se generó su perfil de manera correcta.
Desde ahora puede ingresar al sistema con su usuario y clave.", "Registro
exitoso", MessageBoxButtons.OK, MessageBoxIcon.Information);
        if (result == DialogResult.OK)
        {
            this.Close();
        }
    }
}

#endregion

}
}
```