

Ejercicio: Implementar una aplicación de control personal con los siguientes datos: nombre, apellido, fecha de nacimiento, fecha de contratación. Usar ArrayList.

- El código personal es de la forma "P002" automático
- Botón "Nuevo", nuevo código.
- Botón "Registrar"
- Modificar
- Eliminar → Al hacer doble clic en el control de ListView

Clase: Personal

```
class Personal
{
    O references
    public string codigo { get; set; }

    O references
    public string paterno { get; set; }

    O references
    public string materno { get; set; }

    O references
    public string nombres { get; set; }

    O references
    public DateTime fechaNac { get; set; }

    O references
    public DateTime fechaCon { get; set; }
}
```

Formulario

CÓDIGO	PATERNO	METERNO	NOMBRES	FECHA NAC.	FECHA CONTR...
--------	---------	---------	---------	------------	----------------

4 references

```
public partial class Form1 : Form
{
    // ArrayList para almacenar el personal
    ArrayList aPersonal = new ArrayList();

    1 reference
    public Form1()
    {
        InitializeComponent();
    }
}
```

```
public string generaCodigo()
{
    string codigo = "";
    if (aPersonal.Count < 1) codigo = "P001";
    else
    {
        // Para hallar el último código
        foreach (Personal p in aPersonal)
        {
            codigo = p.codigo.ToString(); // P004
        }
    }
    return "P" + (int.Parse(codigo.Substring(1, 3)) + 1).ToString("000");
}
```

// Método para listado

1 reference

```
private void listado()
{
    lvPersonal.Items.Clear(); // Simpre limpiar antes de insertar
    foreach (Personal p in aPersonal)
    {
        ListViewItem fila = new ListViewItem(p.codigo);
        fila.SubItems.Add(p.paterno);
        fila.SubItems.Add(p.materno);
        fila.SubItems.Add(p.nombres);
        fila.SubItems.Add(p.fechaNac.ToShortDateString());
        fila.SubItems.Add(p.fechaCon.ToShortDateString());
        lvPersonal.Items.Add(fila);
    }
}
```

```
// Método para limpiar los controles
0 references
public void limpiarControles()
{
    txtPaterno.Clear();
    txtMaterno.Clear();
    txtNombres.Clear();
}
```

```
// Método para llenar los campos al hacer doble clic al ListView
1 reference
private void lvPersonal_MouseDoubleClick(object sender, MouseEventArgs e)
{
    // Obtener la posición del elemento seleccionado (doble en el código)
    ListViewItem elemento = lvPersonal.GetItemAt(e.X, e.Y); // (X, Y)

    // Verificación si existe elemento
    if (elemento != null)
    {
        lblCodigo.Text = elemento.Text;
        txtPaterno.Text = elemento.SubItems[1].Text;
        txtMaterno.Text = elemento.SubItems[2].Text;
        txtNombres.Text = elemento.SubItems[3].Text;
        txtFechaNacimiento.Text = elemento.SubItems[4].Text;
        txtFechaContrato.Text = elemento.SubItems[5].Text;
    }
}
```

Generar el primer código al abrir la aplicación

```
1 reference
private void Form1_Load(object sender, EventArgs e)
{
    lblCodigo.Text = generaCodigo();
}
```

```
// Generar nuevo código al hacer clic en el botón "Nuevo"
1 reference
private void btnNuevo_Click(object sender, EventArgs e)
{
    lblCodigo.Text = generaCodigo();
}
```

```

private void btnModificar_Click(object sender, EventArgs e)
{
    try
    {
        foreach (Personal p in aPersonal)
        {
            if (p.codigo == lblCodigo.Text)
            {
                p.paterno = txtPaterno.Text;
                p.materno = txtMaterno.Text;
                p.nombres = txtNombres.Text;
                p.fechaNac = DateTime.Parse(txtFechaNacimiento.Text);
                p.fechaCon = DateTime.Parse(txtFechaContrato.Text);
                break;
            }
        }
        listado();
        generaCodigo();
        limpiarControles();
    }
}

```

```

        limpiarControles();
    }
    catch (Exception)
    {
        MessageBox.Show("Error al modificar registro");
    }
}

```

```

1 reference
private void btnEliminar_Click(object sender, EventArgs e)
{
    try
    {
        foreach (Personal p in aPersonal)
        {
            if (p.codigo == lblCodigo.Text)
            {
                DialogResult dialog = MessageBox.Show("¿Desea eliminar?",
                    "Eliminar Registro", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
                if (dialog == DialogResult.Yes)
                {
                    aPersonal.Remove(p);
                    break;
                }
            }
        }
        listado();
    }
}

```

```
        }
    }
    catch (Exception)
    {
        MessageBox.Show("Error al eliminar");
    }
}
```

1 reference

```
private void btnSalir_Click(object sender, EventArgs e)
{
    DialogResult dialog = MessageBox.Show("¿Desea salir?",
        "Salir", MessageBoxButtons.YesNo, MessageBoxIcon.Error);
    if (dialog == DialogResult.Yes)
    {
        this.Close();
    }
}
```