

PROGRAMACION

GUIA DE

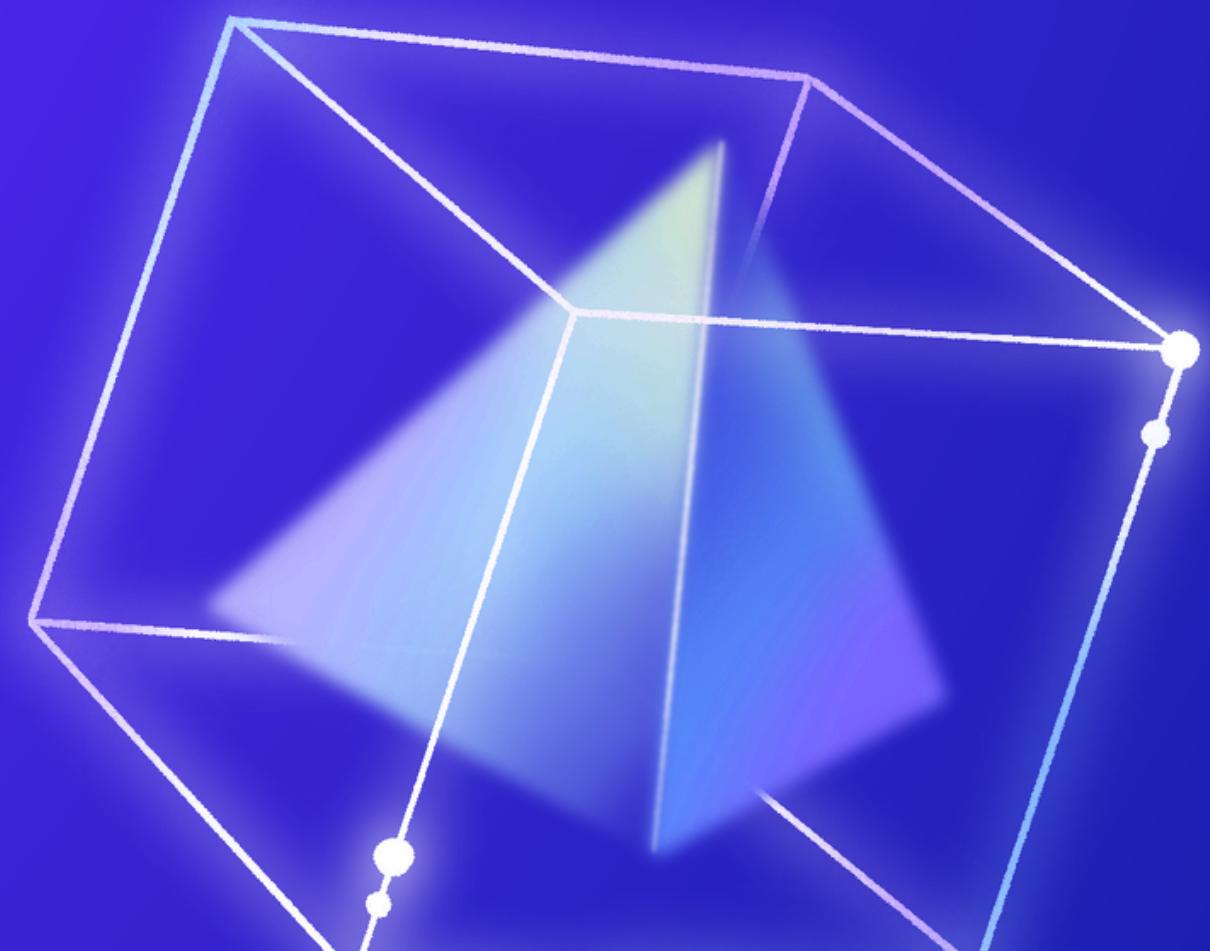
APRENDIZAJE





BUENAS TARDES

- Nombre: Ernesto Carlo Meza García
- Profesor: Christopher Larios.



COMIENZO

Creamos nuestra clase

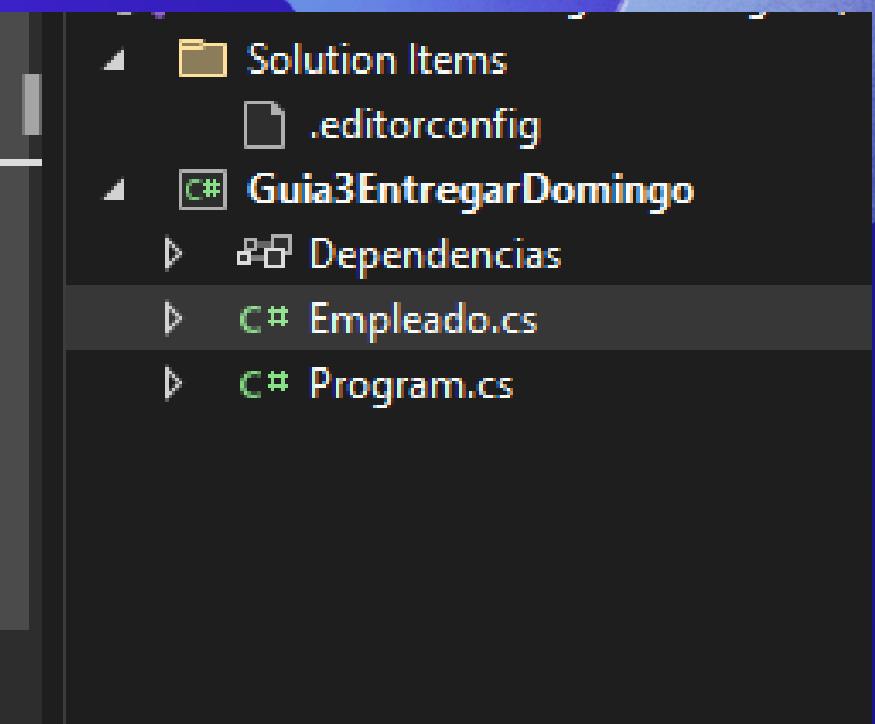
```
namespace Guia3EntregarDomingo
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //creamos una lista de los 3 empleados que tenemos
            List<Empleado> listaEmpleados = new List<Empleado>();

            listaEmpleados.Add(new Empleado("Juan", 101, 30000));
            listaEmpleados.Add(new Empleado("Maria", 102, 35000));
            listaEmpleados.Add(new Empleado("Pedro", 103, 32000));
            // nos ayudamos de un auxiliar
            foreach (var empleado in listaEmpleados)
            {
                empleado.MostrarDetalles(); // llamamos el metodo mostrar detalles que obtiene los nombres, numempleado y
            }
            //cambiamos el estado del empleado si queremos que sea activo o inactivo
            int numeroEmpleadoCambiarEstado = 103;
            Empleado empleadoCambiarEstado = BuscarEmpleado(listaEmpleados, numeroEmpleadoCambiarEstado);

            if (empleadoCambiarEstado != null)
            {
                empleadoCambiarEstado.CambiarEstado(1);
                empleadoCambiarEstado.MostrarDetalles();
            }
            else // si llega a tener otro numero que no sea el de el no lo encontrara
            {
                Console.WriteLine($"Empleado con número {numeroEmpleadoCambiarEstado} no encontrado. ");
            }
        }
    }
}
```

CREAMOS EL PRIMER METODO EMPLEADO

```
9 referencias
internal class Empleado
{
    private string nombre;
    private int numeroEmpleado ;
    private decimal salario;
    private bool activo ;
    //Creamos un metodo que obtendra los private para no tener el error en la clase Empleado
3 referencias
public Empleado(string nombre, int numeroEmpleado, decimal salario) {
    this.nombre = nombre;
    this.numeroEmpleado = numeroEmpleado;
    this.salario = salario;
    this.activo = true;
```



Que obtendra el nombre del empleado, el numero del empleado, el salario del empleado y su estado.

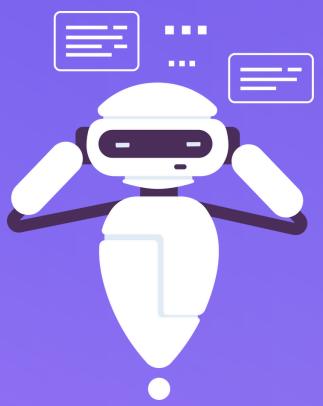
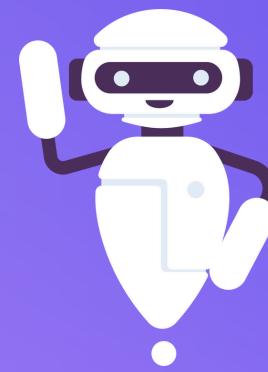
MOSTRAMOS LOS DETALLES DEL EMPLEADO

```
//Creamos el metodo MostrarDetalles obtenemos el nombre, numpleado, salario y su esatdo
2 referencias
public void MostrarDetalles() {

    string estado = (activo) ? "Activo" : "Inactivo";
    Console.WriteLine($"Nombre: {nombre}, Numero de empleado: {numeroEmpleado}, salario: {salario:c}, estado{es
}
```

▷ C# Empleado.cs
▷ C# Program.cs

OBTENEMOS EL NUMERO DEL EMPLEADO



```
//obtenemos el numero de empleado  
1 referencia  
public int ObtenerNumeroEmpleado()  
{  
    return this.numeroEmpleado;  
}
```

CON ESTE METODO LE CAMBIAMOS EL ESTADO DEL EMPLEADO

```
//Creamos un metodo con parametro para cambiar el estado del empleado si esta activado o desactivado activo e inactivo
1 referencia
public void CambiarEstado(int nuevoEstado)
{
    if (nuevoEstado == 1)
    {
        activo = true;
        Console.WriteLine("Empleado activado");
    }
    else if (nuevoEstado == 0)
    {
        activo = false;
        Console.WriteLine("Empleado desactivado");
    }
    else {
        Console.WriteLine("Estado no valido. ");
    }
}
```

0 referencias

```
static void Main(string[] args)
{
    //Creamos una lista de los 3 empleados que tenemos
    List<Empleado> ListaEmpleados = new List<Empleado>();

    ListaEmpleados.Add(new Empleado("Juan", 101, 30000));
    ListaEmpleados.Add(new Empleado("Maria", 102, 35000));
    ListaEmpleados.Add(new Empleado("Pedro", 103, 32000));
```

Creamos una lista para almacenar a los empleados



```
// nos ayudamos de un auxiliar para mostrar los datos de los empleados
foreach (var empleado in listaEmpleados)
{
    empleado.MostrarDetalles(); // llamamos el metodo mostrar detalles que obtiene los nombres, numpleado y
```

MOSTRAMOS LOS DATOS DE LOS EMPLEADOS



CAMBIAMOS EL ESTADO DEL EMPLEADO POR SU NUMERO

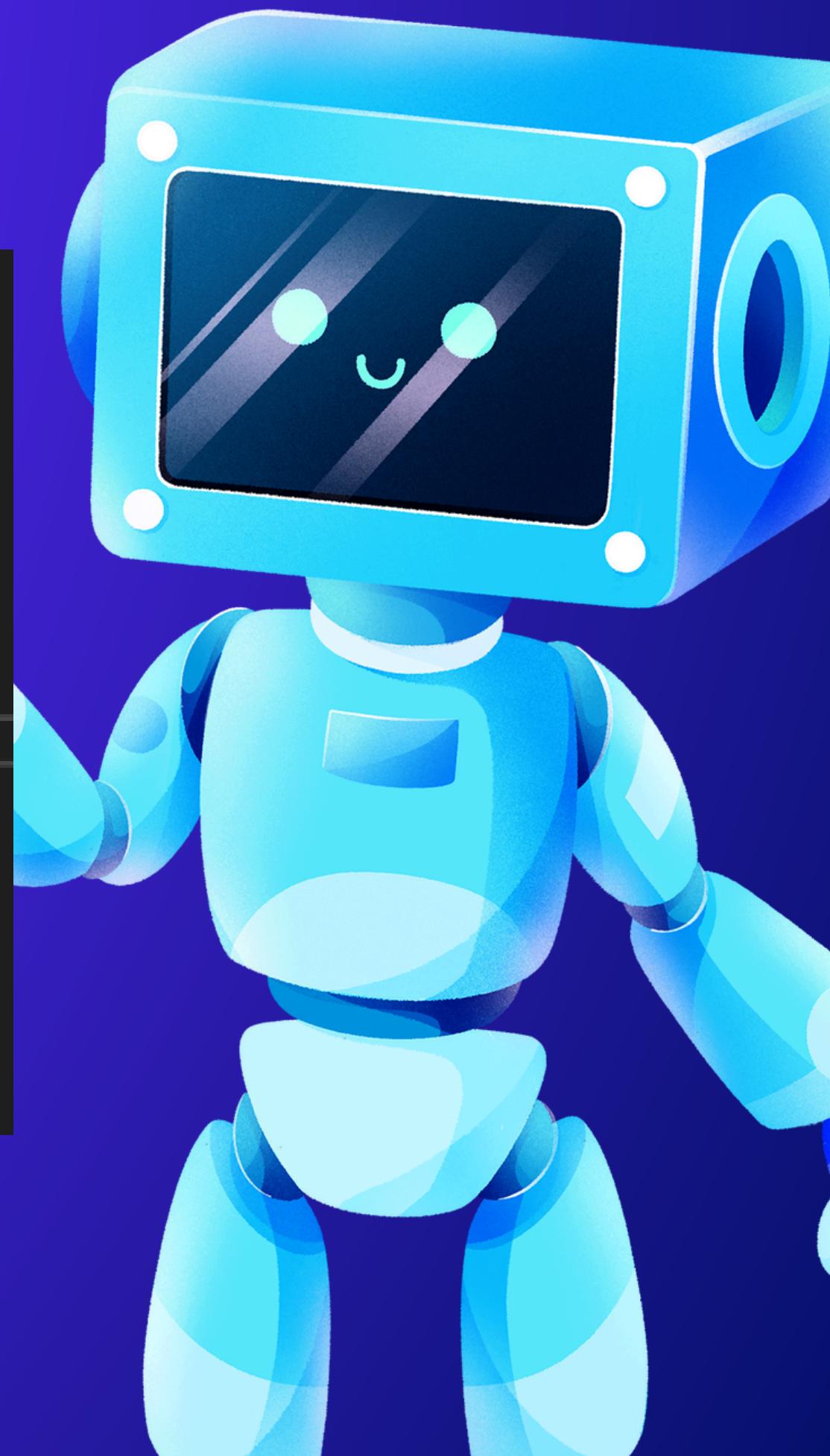
```
    empleado.MostrarDetalles(); // Llamamos el metodo mostrar detalles que obtiene los nombres, num empleados y su estado
}
//cambiamos el estado del empleado si queremos que sea activo o inactivo
int numeroEmpleadoCambiarEstado = 103;
Empleado empleadoCambiarEstado = BuscarEmpleado(ListaEmpleados, numeroEmpleadoCambiarEstado);

if (empleadoCambiarEstado != null)
{
    empleadoCambiarEstado.CambiarEstado(1);
    empleadoCambiarEstado.MostrarDetalles();
}
else // si llega a tener otro numero que no sea el de el no lo encontrara
{
    Console.WriteLine($"Empleado con número {numeroEmpleadoCambiarEstado} no encontrado. ");
}
```

METODO PARA BUSCAR AL EMPLEADO POR SU NUMERO

```
//el metodo nos ayuda para buscar el empleado con el numero del empleado
1 referencia
static Empleado BuscarEmpleado(List<Empleado> empleados, int numeroEmpleado)
{
    foreach (var empleado in empleados)
    {

        if (empleado.ObtenerNumeroEmpleado() == numeroEmpleado)
        {
            return empleado;
        }
    }
    return null;
}
```



PÁGINA DE RECURSOS

