



# PROGRAMACION ORIENTADA A OBJETOS C#

**DESARROLLAR METODOS FALTANTES**

**Programación I ITDSC3-003-SDN**

**NOMBRE: Sabrina Amparo Valdez |2025-1320**

**FACILITADOR: Starling Alfredo Germosén Reynoso**

**SECCIÓN: #1 lunes**

**CARRERA: Desarrollo de Software**



# Transformar el proyecto "Contactes" a POO o desarrolle su propia solución que cumpla con el requerimiento 1 de "Agenda Personal".

```
program
4 //sabrina amparo valdez | 20251320
5 Console.WriteLine("Mi Agenda Perrón");
6 Console.WriteLine("Bienvenido a tu lista de contactos");
7
8 AgendaPersonal agenda = new AgendaPersonal();
9 bool running = true;
10
11 while (running)
12 {
13     Console.Write("1. Agregar Contacto    ");
14     Console.Write("2. Ver Contactos      ");
15     Console.Write("3. Buscar Contactos    ");
16     Console.Write("4. Modificar Contacto    ");
17     Console.Write("5. Eliminar Contacto    ");
18     Console.WriteLine("6. Salir");
19     Console.Write("Elige una opción: ");
20
21     int choice = Convert.ToInt32(Console.ReadLine());
22
23     switch (choice)
24     {
25     case 1:
26         agenda.AgregarContacto();
27         break;
28     case 2:
29         agenda.VerContactos();
30         break;
31     case 3:
32         agenda.BuscarContacto();
33         break;
34     case 4:
35         agenda.ModificarContacto();
36         break;
37     case 5:
38         agenda.EliminarContacto();
39         break;
40     case 6:
41         running = false;
42         break;
43     default:
44         Console.WriteLine("Opción no válida");
```

```
program
44 Console.WriteLine("Opción no válida");
45 break;
46 }
47 }
48
49 // Clase que representa un Contacto
50 public class Contacto
51 {
52     public int Id { get; set; }
53     public string Nombre { get; set; }
54     public string Telefono { get; set; }
55     public string Email { get; set; }
56     public string Direccion { get; set; }
57
58     public Contacto(int id, string nombre, string telefono, string email, string direccion)
59     {
60         Id = id;
61         Nombre = nombre;
62         Telefono = telefono;
63         Email = email;
64         Direccion = direccion;
65     }
66
67     public void MostrarInformacion()
68     {
69         Console.WriteLine($"ID: {Id}");
70         Console.WriteLine($"Nombre: {Nombre}");
71         Console.WriteLine($"Teléfono: {Telefono}");
72         Console.WriteLine($"Email: {Email}");
73         Console.WriteLine($"Dirección: {Direccion}");
74     }
75
76     public void MostrarEnLista()
77     {
78         Console.WriteLine($"{Id,-4} {Nombre,-15} {Telefono,-15} {Email,-20} {Direccion,-20}");
```



```
// Clase que gestiona la agenda de contactos
3 referencias
public class AgendaPersonal
{
    private List<Contacto> contactos;
    private int siguienteId;

    1 referencia
    public AgendaPersonal()
    {
        contactos = new List<Contacto>();
        siguienteId = 1;
    }

    1 referencia
    public void AgregarContacto()
    {
        Console.WriteLine("Vamos a agregar ese contacto que te trae loco.");

        Console.Write("Digite el Nombre: ");
        var nombre = Console.ReadLine();

        Console.Write("Digite el Teléfono: ");
        var telefono = Console.ReadLine();

        Console.Write("Digite el Email: ");
        var email = Console.ReadLine();

        Console.Write("Digite la dirección: ");
        var direccion = Console.ReadLine();

        Contacto nuevoContacto = new Contacto(siguienteId, nombre, telefono, email, direccion);
        contactos.Add(nuevoContacto);
        siguienteId++;

        Console.WriteLine("Contacto agregado exitosamente!\n");
    }
}
```

```
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156

3 referencias
public void VerContactos()
{
    if (contactos.Count == 0)
    {
        Console.WriteLine("No hay contactos en la agenda.\n");
        return;
    }

    Console.WriteLine("Id      Nombre      Teléfono      Email      Dirección");
    Console.WriteLine("-----");

    foreach (var contacto in contactos)
    {
        contacto.MostrarEnLista();
    }

    Console.WriteLine();
}

1 referencia
public void BuscarContacto()
{
    if (contactos.Count == 0)
    {
        Console.WriteLine("No hay contactos en la agenda.\n");
        return;
    }

    Console.WriteLine("Digite un Id de Contacto Para Mostrar");
    int idSeleccionado = Convert.ToInt32(Console.ReadLine());

    Contacto contacto = contactos.FirstOrDefault(c => c.Id == idSeleccionado);

    if (contacto != null)
    {
        Console.WriteLine("\nInformación del contacto:");
        Console.WriteLine("-----");
        contacto.MostrarInformacion();
    }
    else
    {
        Console.WriteLine("Contacto no encontrado.");
    }
}
```





```
7      }
8      Console.WriteLine();
9  }
10
11  1 referencia
12  public void ModificarContacto()
13  {
14      if (contactos.Count == 0)
15      {
16          Console.WriteLine("No hay contactos en la agenda.\n");
17          return;
18      }
19
20      VerContactos();
21      Console.WriteLine("Digite un Id de Contacto Para Editar");
22      int idSeleccionado = Convert.ToInt32(Console.ReadLine());
23
24      Contacto contacto = contactos.FirstOrDefault(c => c.Id == idSeleccionado);
25
26      if (contacto != null)
27      {
28          Console.Write($"El nombre es: {contacto.Nombre}, Digite el Nuevo Nombre: ");
29          contacto.Nombre = Console.ReadLine();
30
31          Console.Write($"El Teléfono es: {contacto.Telefono}, Digite el Nuevo Teléfono: ");
32          contacto.Telefono = Console.ReadLine();
33
34          Console.Write($"El Email es: {contacto.Email}, Digite el Nuevo Email: ");
35          contacto.Email = Console.ReadLine();
36
37          Console.Write($"La dirección es: {contacto.Direccion}, Digite la nueva dirección: ");
38          contacto.Direccion = Console.ReadLine();
39
40          Console.WriteLine("Contacto modificado exitosamente!\n");
41      }
42      else
43      {
44          Console.WriteLine("Contacto no encontrado.\n");
45      }
46  }
```

```
197  1 referencia
198  public void EliminarContacto()
199  {
200      if (contactos.Count == 0)
201      {
202          Console.WriteLine("No hay contactos en la agenda.\n");
203          return;
204      }
205
206      VerContactos();
207      Console.WriteLine("Digite un Id de Contacto Para Eliminar");
208      int idSeleccionado = Convert.ToInt32(Console.ReadLine());
209
210      Contacto contacto = contactos.FirstOrDefault(c => c.Id == idSeleccionado);
211
212      if (contacto != null)
213      {
214          Console.WriteLine($"¿Seguro que desea eliminar a {contacto.Nombre}? 1. Si, 2. No");
215          int opcion = Convert.ToInt32(Console.ReadLine());
216
217          if (opcion == 1)
218          {
219              contactos.Remove(contacto);
220              Console.WriteLine("Contacto eliminado exitosamente!\n");
221          }
222          else
223          {
224              Console.WriteLine("Operación cancelada.\n");
225          }
226      }
227      else
228      {
229          Console.WriteLine("Contacto no encontrado.\n");
230      }
231  }
```



```
AgendaPersonal

// Método adicional para búsqueda por nombre (puede ser extendido)
0 referencias
public void BuscarPorNombre(string nombre)
{
    var resultados = contactos.Where(c => c.Nombre.Contains(nombre, StringComparison.OrdinalIgnoreCase)).ToList();

    if (resultados.Count > 0)
    {
        Console.WriteLine($"Resultados para '{nombre}':");
        foreach (var contacto in resultados)
        {
            contacto.MostrarInformacion();
            Console.WriteLine();
        }
    }
    else
    {
        Console.WriteLine($"No se encontraron contactos con el nombre '{nombre}'");
    }
}
```

```
Elige una opción: 2
No hay contactos en la agenda.

1. Agregar Contacto      2. Ver Contactos      3. Buscar Contactos      4. Modificar Contacto      5. Eliminar Contacto      6. Salir
Elige una opción: 1
Vamos a agregar ese contacto que te trae loco.
Digite el Nombre: hayden
Digite el Teléfono: 8097653421
Digite el Email: haydenc@gmail.com
Digite la dirección: anacaona juan pablo II #5 parte atras
Contacto agregado exitosamente!

1. Agregar Contacto      2. Ver Contactos      3. Buscar Contactos      4. Modificar Contacto      5. Eliminar Contacto      6. Salir
Elige una opción: 3
Digite un Id de Contacto Para Mostrar
1

Información del contacto:
-----
ID: 1
Nombre: hayden
Teléfono: 8097653421
Email: haydenc@gmail.com
Dirección: anacaona juan pablo II #5 parte atras

1. Agregar Contacto      2. Ver Contactos      3. Buscar Contactos      4. Modificar Contacto      5. Eliminar Contacto      6. Salir
Elige una opción:
```