

C# Properties Cheat Sheet

1. Propiedades Automáticas

```
public class Persona
{
    // Propiedad automática
    public string Nombre { get; set; }
}
```

2. Propiedades con Atributos Privados

```
public class Persona
{
    // Atributo privado
    private string nombre;

    // Propiedad pública
    public string Nombre
    {
        get { return nombre; }
        set { nombre = value; }
    }
}
```

3. Propiedades con Validación en el Setter

```
public class Persona
{
    // Atributo privado
    private int edad;

    // Propiedad con validación
    public int Edad
    {
        get { return edad; }
        set
        {
            if (value >= 0)
            {
                edad = value;
            }
            else
            {
                throw new ArgumentOutOfRangeException("La edad no puede ser negativa.");
            }
        }
    }
}
```

4. Propiedades de Solo Lectura

```

public class Persona
{
    // Atributo privado
    private string id;

    // Propiedad de solo lectura
    public string Id
    {
        get { return id; }
    }

    // Constructor
    public Persona(string id)
    {
        this.id = id;
    }
}

```

5. Propiedades de Solo Escritura

```

public class Persona
{
    // Atributo privado
    private string password;

    // Propiedad de solo escritura
    public string Password
    {
        set { password = value; }
    }
}

```

6. Propiedades con Lógica en el Getter y Setter

```

public class Persona
{
    // Atributo privado
    private string nombreCompleto;

    // Propiedad con lógica
    public string NombreCompleto
    {
        get
        {
            return nombreCompleto.ToUpper();
        }
        set
        {
            if (!string.IsNullOrEmpty(value))
            {
                nombreCompleto = value;
            }
            else
            {
                throw new ArgumentException("El nombre no puede estar vacío.");
            }
        }
    }
}

```

7. Propiedades con Setter Privado

```
public class Persona
{
    // Propiedad con setter privado
    public string Id { get; private set; }

    // Constructor
    public Persona(string id)
    {
        Id = id;
    }
}
```

Ejemplo Completo

```
public class Persona
{
    // Atributos privados
    private string nombre;
    private int edad;

    // Propiedad para nombre
    public string Nombre
    {
        get { return nombre; }
        set { nombre = value; }
    }

    // Propiedad para edad con validación
    public int Edad
    {
        get { return edad; }
        set
        {
            if (value >= 0)
            {
                edad = value;
            }
            else
            {
                throw new ArgumentOutOfRangeException("La edad no puede ser negativa.");
            }
        }
    }

    // Propiedad con setter privado
    public string Id { get; private set; }

    // Constructor
    public Persona(string nombre, int edad, string id)
    {
        Nombre = nombre;
        Edad = edad;
        Id = id;
    }

    // Método para mostrar información
    public void MostrarInformacion()
    {
        Console.WriteLine($"Nombre: {Nombre}, Edad: {Edad}, Id: {Id}");
    }
}

// Uso de la clase
class Program
```

```

{
    static void Main(string[] args)
    {
        Persona persona = new Persona("Juan", 30, "12345");
        persona.MostrarInformacion();

        persona.Nombre = "María";
        persona.Edad = 25;

        persona.MostrarInformacion();

        try
        {
            persona.Edad = -5;
        }
        catch (ArgumentOutOfRangeException ex)
        {
            Console.WriteLine(ex.Message);
        }
    }
}

```

Resumen

- **Propiedades Automáticas:** `public string Nombre { get; set; }`
- **Propiedades con Atributos Privados:** Encapsulan campos privados con getters y setters.
- **Validación en el Setter:** Asegura que los valores asignados sean válidos.
- **Propiedades de Solo Lectura:** `public string Id { get; }`
- **Propiedades de Solo Escritura:** `public string Password { set; }`
- **Lógica en Getter y Setter:** Realizan operaciones adicionales al obtener o establecer un valor.
- **Propiedades con Setter Privado:** `public string Id { get; private set; }`

Esta cheat sheet cubre los conceptos básicos y algunas prácticas comunes para trabajar con propiedades en C#.