

## Clase 7- interfaces

Junes, 17 de abril de 2023 15:51

```
public interface IMiInterface
{
    public void UnMetodo();
}
```

Atención, no usar modificadores si se está utilizando una versión anterior a C# 8.0

Creacion de interfaces.

```
interface IImprimible
{
    void Imprimir();
}
```

```
interface IAgrandable
{
    void Agrandar(double factor);
    double TamañoMaximo { get; set; }
}
```

```
class Rombo : Figura, IImprimible, IAgrandable
{
    public void Imprimir()
    { . . . }

    public void Agrandar(double factor)
    { . . . }

    public double TamañoMaximo
    {
        get { . . . }
        set { . . . }
    }
}
```

Obligado a implementarlo

Si una clase implementa una interfaz debe implementar todos los miembros de la interfaz que no tienen implementación predeterminada.

Si una clase deriva de otra clase y además implementa algunas interfaces, la clase debe ser la primera en la lista después de los dos puntos

```
Rombo r1 = new Rombo();
Figura r2 = new Rombo();
IAgrandable r3 = new Rombo();
IImprimible r4 = new Rombo();
```

Las siguientes son sentencias son válidas

```
r3.TamañoMaximo = 100;
r3.Agrandar(1.2);
r4.Imprimir();
(r3 as IImprimible)?.Imprimir();
```

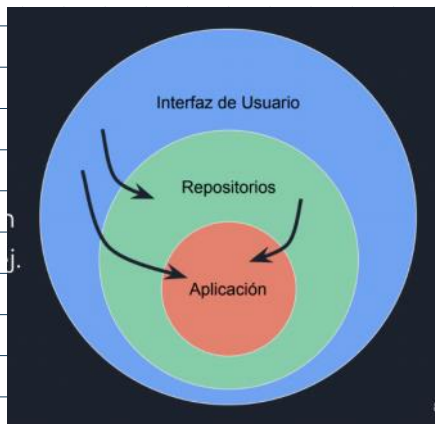
Las interfaces son tipos de referencia, por lo tanto es posible utilizar el operador `as` (ya lo vimos). Es habitual combinar su uso con el del operador `is` de la siguiente manera:

```
. . .
object o;
. . .
if (o is IImprimible)
{
    (o as IImprimible).Imprimir();
}
. . .
```

```
. . .
object o;
. . .
if (o is IImprimible imp)
{
    imp.Imprimir();
}
. . .
```

facilidad incorporada en la versión 7.0 de C#

No se puede instanciar interfaz



Aplicacion: casos de uso (cosas que hace el program ej agregar cliente)