# Clases

* Definición de un tipo de objeto que indica sus atributos y métodos.
* Molde o plantilla a partir de la cual se crean objetos.
* Diseñadas con una responsabilidad en particular (cohesión).
* Pueden depender en cierto grado de otras clases (acoplamiento).

Una clase debe tener un único objetivo, si tiene múltiples objetivos va a perder cohesión y va a ser un programa más difícil de controlar.

Mientras menos acoplamiento tengan las clases, un mejor software vamos a tener.

Para crear una clase, utilice la palabra reservada **class** y enseguida el nombre de la clase comenzando con la primera letra en mayúscula.

namespace Clases

{

class Rectangulo

{

// propiedades o atributos

public int Base { get; set; }

public int Altura { get; set; }

// métodos

public int CalcularArea()

{

return Base \* Altura;

}

}

}

## Modificadores de acceso

La palabra reservada public es un modificador de acceso, que se utiliza para establecer el nivel de acceso/visibilidad para clases, campos, métodos y propiedades.

C# tiene los siguientes modificadores de acceso:

**public:** El código es accesible para todas las clases, no tiene restricciones.

**private:** El código solo es accesible dentro de la misma clase.

**protected:** El código es accesible dentro de la misma clase, o en una clase que se hereda de esa clase.

**internal:** El código solo es accesible dentro de su propio assembly, pero no desde otro assembly.

También hay dos combinaciones**: protected internal** y **private protected**.

**protected internal:** El acceso está limitado al assembly o tipos actuales derivados de la clase contenedora.

**private protected:** El acceso está limitado a la clase o tipos que contienen derivados de la clase contenedora dentro del conjunto actual. Disponible desde C# 7.2.

Solo se permite un modificador de acceso para un miembro o tipo, excepto cuando usa las combinaciones protegidas internas o privadas protegidas.

Los modificadores de acceso no están permitidos en los espacios de nombres (namespaces). Los espacios de nombres no tienen restricciones de acceso.

## Constructores

## Atributos

## Métodos

# Objetos

* Entidad con estado y comportamiento.
* Estructura de datos (color, modelo, marca) junto con sus operaciones (acelerar, frenar, girar).
* Instancia de una clase.

Su comportamiento son las acciones que puede realizar.

Su estado son las propiedades o entidades que lo hacen único.

## Instancia de clases

// Crear un objeto de tipo Rectangulo

Rectangulo rect = new Rectangulo();

Podemos instanciar e inicializar variables de una manera más sencilla pasándole los parámetros entre llaves.

// Sintaxis de inicialización

Rectangulo rect2 = new Rectangulo { Base = 3, Altura = 8 };

## Invocar propiedades o métodos de objetos

Bibliografía

<https://www.w3schools.com/cs/cs_properties.php>

<https://www.udemy.com/course/c-sharp-para-principiantes/learn/lecture/4801520>

<https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/keywords/accessibility-levels>

<https://www.w3schools.com/cs/cs_classes.php>

<https://www.w3schools.com/cs/cs_access_modifiers.php>