

Introducción a C#

4.1.- CONCEPTOS DE P.O.O.

OBJETO : Bloque dentro de la aplicación O.O.

→ Los objetos se crean de tipos definidos llamados clases

→ Objeto es una instancia de una clase (actúa de plantilla)

PROPIEDADES Y CAMPOS dan acceso a los datos del objeto.

Diferentes objetos instanciados de la misma clase pueden tener distintos valores de sus datos → estado de un objeto

→ Tanto las propiedades como los campos tienen un tipo.

→ Las propiedades se diferencian de los campos que éstas se acceden a través de métodos set y get y no directamente como los campos.

→ En principio es mejor utilizar propiedades a campos para almacenar los datos, ya que proporcionan mayor control y desde el punto de vista del código el acceso es idéntico

Introducción a Visual C#

- Es posible establecer un acceso de sólo lectura y de sólo escritura sobre las propiedades
- Es posible tener distinta accesibilidad para leer o escribir un dato en una propiedad
- Una práctica común es definir campos privados (sólo accesible desde dentro de la clase) y proporcionar acceso controlado fuera de la clase mediante propiedades

MÉTODOS

- Es el nombre que reciben las funciones que proporciona un objeto.
- Los métodos se utilizan para proporcionar acceso a la funcionalidad del objeto

Introducción a Visual C#

En C# y en .NET Framework TODO es un OBJETO.

→ En carácter punto (.) separa el nombre del objeto de los métodos, propiedades o campos que se quiere acceder.

→ Ciclo de vida de un objeto : Construcción , Ejecución y Destrucción.

CONSTRUCCIÓN

→ Ocurre cuando es inicializado y se lleva a cabo por una función que se denomina CONSTRUCTOR

→ La inicialización básica se lleva a cabo de forma automática (posicionar el objeto en memoria)

→ El constructor se encarga de inicializar el estado del objeto.

Introducción a Visual C#

La definición de la clase contiene como mínimo un constructor

- Constructor por defecto

 - Método que tiene el mismo nombre que la clase y no contiene parámetros

 - No es necesario definirlo

- Podemos suministrar tantos constructores no por defecto.

 - Métodos que tienen el mismo nombre que la clase, pero con diferentes parámetros

En C# los constructores se llaman a través de la palabra clave **new**, momento en el que se crea un objeto

Fuera de la clase no se pueden utilizar constructores definidos como **private**

Se puede obligar a utilizar un constructor no por defecto definiendo el constructor por defecto **private**

Introducción a Visual C#

DESTRUCCIÓN

- Ocurre cuando un objeto es eliminado (cuando acaba el ámbito de ese objeto y ya no es accesible por el código)
Proceso : Garbage Collector del .NET runtime
- La inicialización básica se lleva a cabo de forma automática (quitar el objeto de memoria)
- El destructor no es necesario especificarlo
De especificarlo se ejecutaría justo antes de eliminar el objeto.
- método con el mismo nombre de la clase precedido por el carácter tilde (~)

Introducción a Visual C#

MIEMBROS ESTATICOS (static)

→ miembros de instancia de clase

Introducción a Visual C#