Clases abstractas:

* Pueden contener miembros virtuales y abstractos.
* Pueden contener miembros de instancia.

Si se define un método abstracto:

* La clase derivada debe utilizar la palabra reservada *override* para implementar el método abstracto.
* Por definición los métodos abstractos son virtuales, pero no pueden estar marcados con *virtual* de forma explícita.

Una clase sellada que hereda de una clase abstracta:

* Sólo debe implementar los métodos abstractos.

Concepto de Polimorfismo:

* Tiende a existir en la relaciones de herencia.
* Es la propiedad que tienen los objetos de invocar genéricamente un comportamiento.

Programación orientada a objetos (POO):

* Es una manera de construir sofware basada en un nuevo paradigma.
* Propone resolver problemas de la realidad a través de identificar objetos y relaciones de colaboración entre ellos.
* El Objeto y el Mensaje son sus elementos fundamentales.

Un método virtual NO puede crearse cuando:

* La clase está definida como *sealed*.

Un constructor no puede ser declarado como *virtual*.

Una propiedad definida como virtual en una clase:

* Debe tener implementación en la clase que se definió.

Interfaces:

* Por definición todos los miembros son abstractos, pero no pueden llevar la palabra *abstract*.

Métodos de extensión:

* Son métodos estáticos que se usan como métodos de instancia en las clases donde se agregan.
* Se definen como estáticos en clase estáticas.

Palabra *new*:

* Lo único que hace es adquirir memoria binaria sin inicializar.
* No participa de ninguna manera en la inicialización de la memoria.

Firma del método:

* Nombre método, tipo y cantidad de parámetros, modificador de parámetro (out o ref).
* NO: nombre de parámetros, tipo de retorno del método.

Arrays:

* Implementa IEnumerable, se puede usar *foreach*.
* Son todos dinámicos.
* Int[] unidimensional. Int[ , ] multidimensional.