

Abstracción y objetos

Ernesto Rodriguez

Universidad del Itsmo

erodriguez@unis.edu.gt

- Es un proceso que ocurre enteramente dentro de la mente del programador.
- Consiste en construir un *objeto abstracto* a partir de un *objeto real*.
- El objeto abstracto solamente contiene los detalles importantes según el contexto.
- El objeto abstracto ignora los demás detalles del objeto original.
- Existen varios (posiblemente infinitos) objetos abstractos que se pueden construir a partir de un objeto concreto.

¿Para que nos sirve la abstracción?

- No es posible ni practico modelar todos los detalles de un objeto en un sistema de software.
- Nos permite hacer generalizaciones sobre clases de objetos. Ejemplo: desplegar una imagen en un monitor.
- Nos ayuda a simplificar problemas.
- Nos facilita razonar acerca de un sistema de software.
- Nos ayuda a hacer codigo más simple y consiso.

Ejemplo: Un automovil

- ¿Que características tiene un automóvil de verdad?
- ¿Que objetos abstractos se pueden construir a partir de un automóvil de verdad?
- ¿Que aplicaciones tienen cada uno de estos objetos?

Programación orientada a objetos

- Uno de los paradigmas principales de programación junto a *funcional* y *logica*.
- Agrupa atributos junto a operaciones sobre esos atributos en una estructura llamada objeto.
- Los objetos creados mediante abstracción pueden ser representados con objetos de la programación.
- El objeto tiene un estado, el cual es el valor actual de todos sus atributos.
- El estado del objeto se modifica mediante sus operaciones.

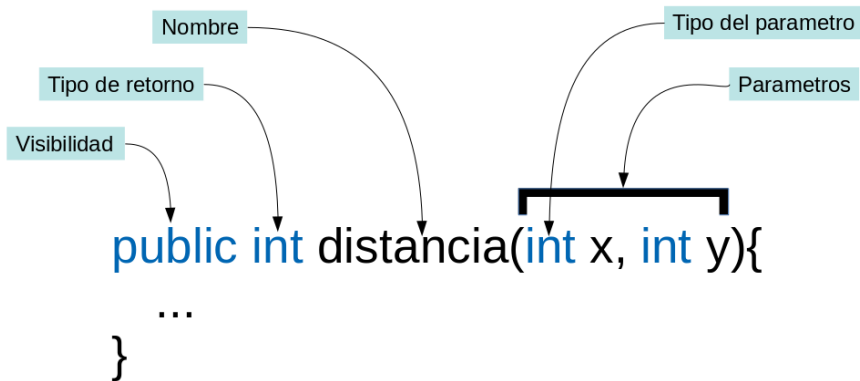
- Describen las propiedades y operaciones de un objeto.
- Se definen mediante un lenguaje de programación como C#.
- Los objetos abstractos pueden representarse mediante clases.
- Una clase permite crear uno o más objetos durante la ejecución de un programa.
- Un objeto creado a partir de una clase se llama *instancia de la clase*.

- También se conocen como objetos.
- Se construyen a partir de una clase.
- Cada objeto existe en su propio espacio de memoria, incluso si fueron creados con la misma clase.
- Al asignar un objeto a una variable, la variable contiene una *referencia* al objeto.
- El estado del objeto se puede modificar mediante *metodos*.

- Corresponden a valores almacenados en la memoria del objeto.
- Sirven como etiquetas que apuntan a una posición particular en memoria.
- Pueden ser *privadas* o *publicas*.
- Las propiedades privadas solo pueden ser accedidas por *metodos* de la clase instanciadora.
- Las propiedades publicas pueden ser accedidas por metodos ajenos a la clase instanciadora.
- Ejemplos: Longitud de un arreglo, posición del *Vehiculo*.

- Los metodos son funciones asociadas a una clase.
- También pueden ser publicos o privados.
- El cuerpo de un metodo puede acceder a todas las propiedades y metodos de la clase instanciadora.
- Consecuentemente, un método puede modificar el *estado* de un objeto.
- La palabra reservada *this* se utiliza para acceder propiedades y metodos del objeto al que le pertenece el metodo.
- El metodo puede recibir parametros que pueda necesitar para realizar su trabajo.

Firma de un Método



Método constructor

- Metodo encargado de colocar a un objeto en su estado inicial.
- Solamente es llamado cuando un objeto nuevo es creado.
- Puede ser publico o privado.
- Se declara mediante el nombre de la clase.
- Se llama mediante la palabra reservada *new*.