

Elaboración de diagramas de clases (Entornos de desarrollo)

Clases. Atributos, métodos y visibilidad

Una clase es un modelo o prototipo a partir del cual se crean los objetos. Se enfoca intencionalmente en lo básico, mostrando cómo incluso una clase simple puede modelar limpiamente el estado y el comportamiento.

En el mundo real, a menudo encontrará muchos objetos individuales del mismo tipo. Puede haber miles de otras bicicletas, todas de la misma marca y modelo. Cada bicicleta se construyó a partir del mismo conjunto de planos y, por lo tanto, contiene los mismos componentes. En términos de orientación a objetos, decimos que su bicicleta es una instancia de la clase de objetos conocida como bicicletas. Una clase es el plano a partir del cual se crean los objetos individuales.

Las clases contienen atributos que definen cómo es un objeto.

Los métodos operan en el estado interno de un objeto y sirven como el mecanismo principal para la comunicación de objeto a objeto. Ocultar el estado interno y requerir que toda la interacción se realice a través de los métodos de un objeto se conoce como encapsulación de datos, un principio fundamental de la programación orientada a objetos.

Más información

- <https://www.jesusninoc.com/03/07/trabajar-con-clases-en-powershell-5/>
-

Objetos. Instanciación

Un objeto es un paquete de software de estado y comportamiento relacionados. Los objetos de software se utilizan a menudo para modelar los objetos del mundo real que se encuentran en la vida diaria.

Los objetos del mundo real comparten dos características: todos tienen estado y comportamiento . Los perros tienen estado (nombre, color, raza, hambre) y comportamiento (ladrar, traer, menear la cola). Las bicicletas también tienen estado (marcha actual, cadencia actual del pedal, velocidad actual) y comportamiento (cambio de marcha, cambio de cadencia del pedal, aplicación de frenos). Identificar el estado y el comportamiento de los objetos del mundo real es una excelente manera de comenzar a pensar en términos de programación orientada a objetos.

Los objetos de software son conceptualmente similares a los objetos del mundo real: también consisten en estados y comportamientos relacionados. Un objeto almacena su estado en campos (variables en algunos lenguajes de programación) y expone su comportamiento a través de métodos (funciones en algunos lenguajes de programación).

Más información

- <https://www.jesusninoc.com/02/13/trabajar-con-objetos-en-powershell/>

- <https://www.jesusninoc.com/02/14/trabajar-con-objetos-en-powershell-5/>
 - <https://www.jesusninoc.com/09/04/crear-nuevos-objetos/>
-

Ejercicios

- <https://www.jesusninoc.com/12/11/ejercicios-de-powershell-realizar-un-inventario-de-tu-equipo-a-nivel-hardware-y-software-ten-en-cuenta-como-clasificar-la-informacion-y-no-olvides-temas-importantes-como-por-ejemplo-controladores/>
-

Relaciones. Herencia, composición, agregación

La herencia proporciona un mecanismo potente y natural para organizar y estructurar su software. Esta sección explica cómo las clases heredan el estado y el comportamiento de sus superclases, y explica cómo derivar una clase de otra usando la sintaxis simple proporcionada por el lenguaje de programación Java.

Los diferentes tipos de objetos a menudo tienen una cierta cantidad en común entre sí. Las bicicletas de montaña, las bicicletas de carretera y las bicicletas tándem, por ejemplo, comparten las características de las bicicletas (velocidad actual, cadencia actual del pedal, marcha actual). Sin embargo, cada uno también define características adicionales que los hacen diferentes: las bicicletas tándem tienen dos asientos y dos juegos de manillares; las bicicletas de carretera tienen manillar abatible; Algunas bicicletas de montaña tienen un anillo de cadena adicional, lo que les da una relación de transmisión más baja.

La programación orientada a objetos permite que las clases hereden el estado y el comportamiento de uso común de otras clases. En el lenguaje de programación Java, cada clase puede tener una superclase directa, y cada superclase tiene el potencial para un número ilimitado de subclases.

Notación de los diagramas de clases

En ingeniería de software, un diagrama de clases en Lenguaje Unificado de Modelado (UML) es un tipo de diagrama de estructura estática que describe la estructura de un sistema mostrando las clases del sistema, sus atributos, operaciones (o métodos), y las relaciones entre los objetos.