

Extends

La palabra clave extends es usada en la declaración o expresión de clases, para crear una clase hija de otra.

Descripción

La palabra clave extends se puede usar para crear una subclase a partir de clases personalizadas, así como sus objetos incorporados.

La propiedad .prototype de la nueva subclase debe ser un Object o null.

Qué es la extensión de clases

Para comprender los mecanismos o las ventajas de la extensión de clases podemos verlo con un ejemplo típico. Por ejemplo, pensemos en los animales. Si tuviéramos que modelizar en programación orientada a objetos la clase animal, le asignaríamos unas propiedades y métodos cualesquiera que necesitásemos.

Otro día podríamos querer definir la clase mamífero (es un animal que tiene unas características específicas). Entonces tendríamos la posibilidad de reescribir todo el comportamiento definido para los animales y añadirle todo el comportamiento o características de los mamíferos. Pero eso no sería totalmente óptimo, porque con ello estaríamos reescribiendo el mismo código que se había programado ya para los animales. Estaríamos haciendo el trabajo dos veces y eso es algo que la programación orientada a objetos trata de evitar.

La programación orientada a objetos permite la herencia o extensión de clases que en estos casos nos viene como anillo al dedo. Debido a que un mamífero es un animal que tiene unas particularidades especiales, lo interesante es definir los mamíferos como una extensión de los animales. Es decir, hacer una clase mamífero que herede las propiedades y métodos de los animales. Con ello sólo tendríamos que definir los comportamientos

y características propias de los mamíferos, apoyándonos en el trabajo realizado anteriormente para la modelización de los animales.

Así mismo, si definiésemos la clase reptil, pues también podríamos ayudarnos extendiendo la clase animal. Es decir, modelizaríamos el reptil a partir del animal (extendiendo la clase animal) y añadiendo los métodos y propiedades de los reptiles.

El esquema de herencia se puede llevar hasta donde sea necesario. Si algún día necesitamos definir la clase perro, pues podríamos hacerlo extendiendo la clase mamífero. O si definimos la clase Dogo Alemán podríamos hacerlo a través de extender la clase perro.

Esto tiene varias ventajas. Una es la reutilización de código, como ya hemos dicho. Otra ventaja sería la encapsulación. Dado que sabemos que la clase perro funciona tal cual queremos, pues podemos crear todas clases con distintas razas de perros sin preocuparnos de cómo funcionan los perros, ni cómo funcionan los mamíferos, ni los animales, porque todas esas clases fueron creadas como un paquete que se comprobó que funcionaba correctamente. Pero si algún día detectamos que una de nuestras clases funciona mal, o se le desea añadir una funcionalidad, simplemente tenemos que editarla y automáticamente ese cambio se trasladará a todas las clases que la extiendan.

Con la herencia, como se puede entender, ganamos en reusabilidad, facilidad de mantenimiento y creación del código.