Hito Grupal Primer Trimestre

Tema Propuesto

Programación orientada a objetos en TypeScript vs JavaScript

- Principios de la programación orientada a objetos

La POO es muy potente porque nos permite modelar de manera sencilla datos y comportamientos complejos del mundo real. Al poder manejar los datos y los comportamientos de cada objeto de manera independiente nos evita tener que mantener datos globales y coordinar todo eso.

Conceptos:

Clase

Una clase es una especie de "plantilla" en la que se definen los atributos y métodos predeterminados de un tipo de objeto. Esta plantilla se crea para poder crear objetos fácilmente. Al método de crear nuevos objetos mediante la lectura y recuperación de los atributos y métodos de una clase se le conoce como instanciación.
Las partes de una clase normalmente son:

- **Nombre de la clase.** Sirve para identificar a todos los objetos que tengan unas determinadas características.
- **Conjunto de atributos**. El valor de los atributos representa el estado de cada objeto.
- **Conjunto de métodos**. Permite que los objetos cambien de estado, dependiendo del estado anterior que tuviera el objeto.
- **Niveles de acceso para proteger ciertos miembros de la clase**. Normalmente, se definirán como ocultos (privados) los atributos y visibles (públicos) los métodos.

Herencia

Por ejemplo, herencia de la clase C a la clase D, es la facilidad mediante la cual la clase D hereda en ella cada uno de los atributos y operaciones de C, como si esos atributos y operaciones hubiesen sido definidos por la misma D. Por lo tanto, puede usar los mismos métodos y variables registrados como "públicos" (*public*) en C. Los componentes registrados como "privados" (*private*) también se heredan, pero se mantienen escondidos al programador y solo pueden ser accedidos a través de otros métodos públicos. Para poder acceder a un atributo u operación de una clase en cualquiera de sus subclases, pero mantenerla oculta para otras clases es necesario registrar los componentes como "protegidos" (*protected*), de esta manera serán visibles en C y en D, pero no en otras clases.

Objeto

Instancia de una clase. Entidad provista de un conjunto de propiedades o atributos (datos) y de comportamiento o funcionalidad (métodos), los mismos que consecuentemente reaccionan a eventos. Se corresponden con los objetos reales del mundo que nos rodea, o con objetos internos del sistema (del programa).

Método

Algoritmo asociado a un objeto (o a una clase de objetos), cuya ejecución se desencadena tras la recepción de un "mensaje". Desde el punto de vista del comportamiento, es lo que el objeto puede hacer. Un método puede producir un cambio en las propiedades del objeto, o la generación de un "evento" con un nuevo mensaje para otro objeto del sistema.

- Abstracción

La abstracción son las características específicas de un objeto, aquellas que lo distinguen de los demás tipos de objetos y que logran definir límites conceptuales respecto a quien está haciendo dicha abstracción del objeto.

Se enfoca en la visión externa de un objeto, separa el comportamiento específico de un objeto, a esta división que realiza se le conoce como la barrera de abstracción, la cual se consigue aplicando el principio de mínimo compromiso.

¿Qué es el principio de mínimo compromiso?

Se refiere al proceso por el cuál la interfaz de un objeto muestra su comportamiento específico y nada más.

¿Qué es una Interfaz?

Una interfaz permite crear código con el cuál se específica que métodos serán implementados por una clase sin necesidad de definir qué harán estos métodos, dichos métodos deben ser públicos.

Existe también el principio de mínima sorpresa, en el cuál una abstracción obtiene el comportamiento completo de algún objeto y por ende no ofrece sorpresas o efectos laterales que lleguen más allá del ámbito de la abstracción.

Hay una alta gama de abstracciones que existen desde los objetos que modelan muy cerca de entidades, a objetos que no tienen razón para existir, vamos a hacer una rápida mención de ello.

- Encapsulamiento

Decimos que el encapsulamiento en la programación orientada a objetos es cuando limitamos el acceso o damos un acceso restringido de una propiedad a los elementos que necesita un miembro y no a ninguno más.

El elemento más común de encapsulamiento son las clases, donde encapsulamos y englobamos tanto métodos como propiedades.

Otro ejemplo muy común de encapsulamiento son los getters y setters de las propiedades dentro de una clase. Por defecto nos dan el valor “normal” pero podemos modificarlos para que cambie.

Un caso real sería que los coches suelen marcar un par de kilómetros por hora más de los reales. Por lo que encapsulamos esa lógica dentro del setter, el cual esta oculto para el consumidor que vería en su cuentaquilómetros la velocidad con los dos kilómetros por hora extra.

Podemos decir que encapsulamiento es una forma de ocultación de información entre entidades, mostrándose entre ellas solo la información más necesaria.

- Herencia

Herencia es un concepto de la programación orientada a objetos. El cual es un mecanismo que permite derivar una clase a otra clase.

En otras palabras, tendremos unas clases que serán hijos, y otras clases que serán padres.

Como podemos observar tanto coche como Moto apuntan a Vehículo, eso quiere decir que pueden usar tanto sus propiedades como sus métodos. Por lo que el siguiente código `Moto.VelocidadMaxima` o `Moto.Cilindrada` es igual de válido que `Coche.VelocidadMaxima` o `Coche.Traccion`.

Cuando heredamos de una clase padre únicamente podemos hacerlo de una sola clase. No podemos heredar de dos clases

Una clase hijo no solo hereda los métodos y propiedades de la clase padre. Sino que además también hereda el tipo de la clase padre.

- Polimorfismo

Denominamos polimorfismo al mecanismo que nos permite tener un método en una clase padre como vimos en la herencia y sobrescribirlo en la clase hija.

Esto quiere decir que tendremos el mismo método en ambas clases, pero en la clase hija realizara diferentes acciones. Por lo que el polimorfismo es también denominado sobreescritura de métodos