**PROGRAMACION ORRIENTADA A OBJETOS**

|  |  |  |
| --- | --- | --- |
| **FICHA: 2143152** | | |
| **ASIGNATURA** | **NOMBRE ALUMNO** | **FECHA** |
| Desarrollo de App desktop con .Net | María Alejandra Montoya Marín | 26/10/2020 |

**¿Qué es?**

se define como un paradigma de la programación, una manera de programar específica, donde se organiza el código en unidades denominadas clases, de las cuales se crean objetos que se relacionan entre sí para conseguir los objetivos de las aplicaciones.

**Clases en POO: (plantilla)** son declaraciones de objetos, también se podrían definir como abstracciones de objetos. Quiere decir que un objeto es la clase, cuando programamos un Objeto y definimos sus características y su función en realidad lo que hacemos e programar una clase.

**Propiedades:** o conocidas como atributos son las características de los objetos cuan las definimos normalmente se especifica su nombre y tipo es algo así como las variables donde almacenaremos nuestros datos.

**Métodos:** estos son los que están asociados a los objetos es su funcionalidad.

**Objeto:** es una Instancia de una clase, y conjunto de propiedades o atributos (datos) y de comportamiento o funcionalidad (métodos).

**Evento:** Es una interacción del usuario con la máquina, o un mensaje enviado por un objeto.

**Atributos:** Características que tiene la clase.

**Estado interno:** Es una variable que se declara privada, que puede ser únicamente accedida y alterada por un método del objeto.

**Componentes de un objeto:** Atributos, identidad, relaciones y métodos.

Para crear un objeto se tienen que escribir instrucciones especiales que puede depender del tipo del lenguaje que utilices.

**EJEMPLO**

miCoche = new Coche();

Con la palabra **NEW** especificamos que se tiene que crear una instancia de la clase que sigue a continuación. Dentro de los paréntesis podríamos colocar parámetros con los que inicializar el objeto de la clase coche.

El valor concreto de una propiedad de un objeto se llama estado. Para acceder a un estado de un objeto para ver su valor o cambiarlo se utiliza el operador punto. luego colocamos el operador punto y por último el nombre de la propiedad a la que deseamos acceder.

MiCoche.Color = ”rojo”

**CARACTERÍSTICAS DE LA POO**

La **HERENCIA** sirve para crear objetos que incorporen propiedades y métodos de otros objetos. Así podremos construir unos objetos a partir de otros sin tener que reescribirlo todo

El **POLIMORFISMO** sirve para que no tengamos que preocuparnos sobre lo que estamos trabajando, y abstraernos para definir un código que sea compatible con objetos de varios tipos.

La **ABSTRACCIÓN** son las características esenciales de un objeto en donde capturamos sus comportamientos. El proceso de abstracción permite seleccionar las características relevantes dentro de un conjunto e identificar comportamientos comunes para definir nuevos tipos de entidades.

El **ENCAPSULAMIENTO** Significa reunir todos los elementos que pueden considerarse pertenecientes a una misma entidad.

La **MODULARIDAD** es una propiedad que permite subdividir una aplicación en partes más pequeñas (llamadas módulos).

El **PRINCIPIO DE OCULTACIÓN** Cada objeto está aislado del exterior, es un módulo natural, y cada tipo de objeto expone una interfaz a otros objetos que especifica cómo pueden interactuar con los objetos de la clase. El aislamiento protege a las propiedades de un objeto contra su modificación por quien no tenga derecho a acceder a ellas.

**Lenguajes orientada a objetos**

* **C++**
* **C#**
* **JAVA**
* **PHP**
* **visual BASIC.NET**
* **PYTHON**