

## **RESUMEN**

### **PROGRAMACIÓN ORIENTADA A OBJETOS EN PHP**

La POO es un paradigma de programación que utiliza objetos e interacciones en el diagrama de un sistema, un paradigma en teoría es cuyo núcleo suministra la base y modelo para resolver problemas.

No nos enseña un método probado y estudiado el cual se basa en las interacciones de objetos, para resolver las necesidades de un sistema informático.

La POO contiene ciertas características conceptuales las cuales debe guardar ciertas y de igual manera identificarlas y diferenciar de otros paradigmas de programación. Dichas características son las siguientes:

- **Abstracción:** es la encargada de definir las características de un objeto
- **Encapsulamiento:** reúne cada uno de los elementos a los cuales se pueden considerar pertenecientes a una misma entidad
- **Modularidad:** es la cual permite dividir una aplicación en varias partes más pequeñas digámoslo así en módulos, independientes unas de otras.
- **Ocultación:** los objetos son aislados del exterior, negando así alguna modificación a la información que contengan y denegando acceder a la misma.
- **Polimorfismo:** permite que varios objetos diferentes puedan contar con un mismo nombre pero sí que uno del otro interfiera en sus propiedades o atributos.
- **Herencia:** Es la relación existente entre dos o más clases, donde una es la principal (padre) y otras son secundarias y dependen (heredan) de ellas (clases "hijas"), donde a la vez, los objetos heredan las características de los objetos de los cuales heredan.
- **Recolección de basura;** Es la técnica que consiste en destruir aquellos objetos cuando ya no son necesarios, liberándonos de la memoria.

### **CLASES O CLASES CONCRETAS**

Una clase es uno de los modelos que se pueden utilizar para la creación de objetos los cuales pueden compartir un mismo comportamiento, estado e identidad.

**Objeto:** Es la entidad provista de métodos o mensajes a los cuales responde (comportamiento); atributos con valores concretos (estado); y propiedades (identidad).

**Método:** Es el algoritmo asociado a un objeto que indica la capacidad de lo que éste puede hacer.

Propiedades y atributo: Las propiedades y atributos, son variables que contienen datos asociados a un objeto.

## **POO BÁSICA**

Herencia de Clases: Los objetos pueden heredar propiedades y métodos de otros objetos. Para ello, PHP permite la “extensión” (herencia) de clases, cuya característica representa la relación existente entre diferentes objetos. Para definir una clase como extensión de una clase “padre” se utiliza la palabra clave `extends`.

La declaración de las clases abstractas son aquellas que no necesitan ser indicadas más sin embargo son heredadas en algún momento y se define anteponiendo la palabra clave `ABSTRACT`, este tipo de clases son las que contienen métodos abstractos, su finalidad es la declaración de clases genéricas las cuales necesitan ser declaradas pero no contienen una definición precisa.

## **OBJETOS E INSTANCIAS**

Una vez que las clases han sido declaradas, será necesario crear los objetos y utilizarlos, aunque hemos visto que algunas clases, como las clases abstractas son solo modelos para otras, y por lo tanto no necesitan instanciar al objeto.

Para instanciar una clase, solo es necesario utilizar la palabra clave `new`. El objeto será creado, asignando esta instancia a una variable (la cual, adoptará la forma de objeto). Lógicamente, la clase debe haber sido declarada antes de ser instanciado.

## **Definición de atributos o propiedades en PHP**

Las propiedades representan ciertas características del objeto en sí mismo. Se definen anteponiendo la palabra clave (`var`) al nombre de la variable (propiedad). No es necesario utilizar la palabra reservada (`var`) para la definición de la variable.

**ACCEDIENDO A LAS PROPIEDADES DE UN OBJETO:** Para acceder a la propiedad de un objeto, existen varias maneras de hacerlo. Todas ellas, dependen del ámbito desde el cual se las invoque así como de su condición y visibilidad.

**CONSTANTES:** Otro tipo de “propiedad” de una clase, son las constantes, aquellas que mantienen su valor de forma permanente y sin cambios. A diferencia de las propiedades estáticas que pueden ser declaradas dentro de una clase, las constantes solo pueden tener una visibilidad pública y no deben ser creadas dentro de las clases. El acceso a constantes es exactamente igual que al de otras propiedades.