

## Orientação a Objetos com PHP: Classes, métodos e atributos

- Esse curso capacita a:
  - Entender a diferença entre o paradigma OO e procedural;
  - Aprender a utilizar construtores;
  - Saber como usar métodos estáticos;
  - Reutilizar código através da composição;
  - Entender classes, objetos, atributos e métodos;

Aulas:

### 1. Problemas da programação procedural:

- Programar utilizando procedimentos executados sequencialmente é chamado de programação procedural;
- Alguns problemas da programação procedural, como repetição de código e falta da definição explícita de tipos, em alguns casos;
- **Abstração**, um dos pilares da orientação a objetos, é a prática de abstrair algo do mundo real, trazendo apenas o que for necessário para o nosso sistema;

### 2. Introdução à Orientação a Objetos:

- A palavra **new** é utilizada para criar um objeto, e devolve o endereço dele, esse endereço é armazenado em uma variável, e através desta variável podemos acessar o objeto e seus atributos;

### 3. Definindo os comportamentos:

- Variável **\$this** tem a referência para o objeto que chamou o método atual;

### 4. Protegendo os nossos dados:

- **Encapsulamento**, outro pilar da orientação a objetos, diz que deve-se encapsular os dados dos objetos e expor apenas o necessário;

### 5. Mais sobre métodos:

- Métodos no PHP que são executados em momentos específicos, e estes métodos são conhecidos como métodos mágicos;
- Ao criar uma instância, o método mágico construtor (**\_\_construct**) é executado;
- Quando uma instância deixa de existir, seu método mágico destrutor (**\_\_destruct**) é executado;
- Também pode-se ter membros (atributos e métodos) da própria classe, e não de cada instância, eles são chamados membros estáticos;

## **6. Composição de objetos:**

- Uma classe deve ter apenas uma responsabilidade, deve ser concisa;
- É possível que um objeto tenha outro objeto como valor de um de seus atributos. Isto é chamado de composição;
- A composição pode (e deve) ser utilizada ao definir uma estrutura complexa de classes;