



UNIVERSIDADE DE  
**VASSOURAS**



UNIVERSIDADE DE VASSOURAS  
Curso de Graduação em Engenharia Software

## Aula 2

# Laboratório de Programação Orientada à Objeto



Prof. Diego Ramos Inácio

Geógrafo

Mestrando em Engenharia de Biossistemas

Especialista em Topografia e Sensoriamento Remoto

Specialist in GIS and Data Modeling em Digimap



## Bibliografia

- Fowler, Martin. UML essencial: um breve guia para a linguagem-padrão de modelagem de objetos / Martin Fowler; trad. João Tortello. - 3.ed. - Porto Alegre: Bookman, 2005. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788560031382>.
- Banin, Sérgio Luiz Python 3: conceitos e aplicações: uma abordagem didática / Sérgio Luiz Banin. --São Paulo: Érica, 2018. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788536530253>.
- Larman, Craig. Utilizando UML e padrões: Uma introdução á análise e ao projeto orientados a objetos e ao desenvolvimento iterativo / Craig Larman ; tradução Rosana T. Vaccare Braga [et al.]. - 3. ed. - Dados eletrônicos. - Porto Alegre : Bookman, 2007. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788577800476>.
- Wazlawick, Raul Sidnei. Introdução aos algoritmos e programação com python: uma abordagem dirigida por testes / Raul Sidnei Wazlawick. - 1. ed. - Rio de Janeiro: Elsevier, 2018. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788595156968>.
- Gamma, Erich. Padrões de projeto: soluções reutilizáveis de software orientado a objetos / Erich Gamma [et al.]; tradução Luiz A. Meirelles 166 Salgado. - Dados eletrônicos. - Porto Alegre: Bookman, 2007. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788577800469>.

## Tópicos

### Ementa da Disciplina

- Conceitos do paradigma da Programação Orientada à Objetos - POO;
- Abstração;
- Introdução a classes e objetos;
- Agregação e Composição de objetos;
- Encapsulamento;
- Herança;
- Polimorfismo;
- Tratamento de Exceções;
- Projeto de POO;
- Linguagem de programação orientada a objetos;
- Realização de teste de software;

## Introdução a Classe e Objetos

### Método Construtor;

Método especial usado para dar o start em um objeto instanciado.

```
def __init__(self, param):  
    lógica atribuída ao método construtor
```

O método `__init__()`  
É automaticamente invocado quando ela é instanciada

```
1 class Pessoa:  
2     def __init__(self, nome, sexo, cpf):  
3         self.nome = nome  
4         self.sexo = sexo  
5         self.cpf = cpf  
6  
7     if __name__ == "__main__":  
8         pessoal = Pessoa("João", "M", "123456")  
9         print(pessoal.nome)
```

João

## Introdução a Classe e Objetos

### Identificador 'self';

Responsável por identificar a instância em que o método é invocado, dessa forma pode se dizer que é uma referencia a instância atual da classe (objeto).

`def __init__(self, param):`

lógica atribuída ao método construtor

Sempre o primeiro parâmetro usado no método, é usado para acessar os membros que pertencem à classe em si

```
1 class Pessoa:
2     def __init__(self, nome, sexo, cpf):
3         self.nome = nome
4         self.sexo = sexo
5         self.cpf = cpf
6
7     if __name__ == "__main__":
8         pessoal = Pessoa("João", "M", "123456")
9         print(pessoal.nome)
```

João

**Contato**



**Professor:**

Diego Ramos Inácio

**E-mail:**

[diego.inacio@univassouras.edu.br](mailto:diego.inacio@univassouras.edu.br)