

Introdução a Programação Orientação a Objetos I

Who i am?

Sarah Soares, M.Sc

Bacharel em Sistemas de Informação (UFAC)

Mestre em Ciência da Computação (UFAC)

Engenheira de Dados - Instituto Talanoa

Engenheira de Software - Tag Dev

Professora Universitária/Mentora em Dados

Who u am?

O QUE TEREMOS NA AULA DE HOJE!



Objetivo da disciplina

- A programação orientada a objetos é bem requisitada no contexto das aplicações atuais.

Atenção

Entre os motivos dessa preferência, estão a possibilidade de **fácil leitura e manutenção de código**

Objetivos específicos da disciplina

- Entender e saber aplicar conceitos de programação algorítmica no paradigmas.
- Saber utilizar os principais tipos de dados.
- Entender o conceito de estruturas de controle e saber utilizar as estruturas de controle.

O que esperar da disciplina

- Conceitos de Orientação a Objetos
- Mensagens
- Métodos, estados
- Tipos e classes
- Polimorfismo
- Identificação de objetos
- Abstrações
- Generalizações

O que esperar da disciplina

- Características, sintaxe, recursos de uma linguagem orientada a objetos
- Tratamento de exceções
- Componentes de interface gráfica (AWT/Swing)
- Acesso a sistema de arquivos

Fonte: “Vozes da Minha Cabeça”

- FURGERI. S. **Programação orientada a objetivos: conceitos e técnicas**. São Paulo: Saraiva: Érica, 2015.
- SARAIVA JR, Orlando. **Introdução à Orientação a Objetos com C++ e Python: Uma abordagem prática**. Novatec. 2017.
- LUTZ, Mark. **Learning Python: Powerful Object-Oriented Programming (English Edition)**. 5ª edição. O'Reilly Media. 2013.
- OBERLEITNER, Allen et. al. **Programação orientada a objetos (Série Universitária)**. Editora Senac São Paulo, 2021.

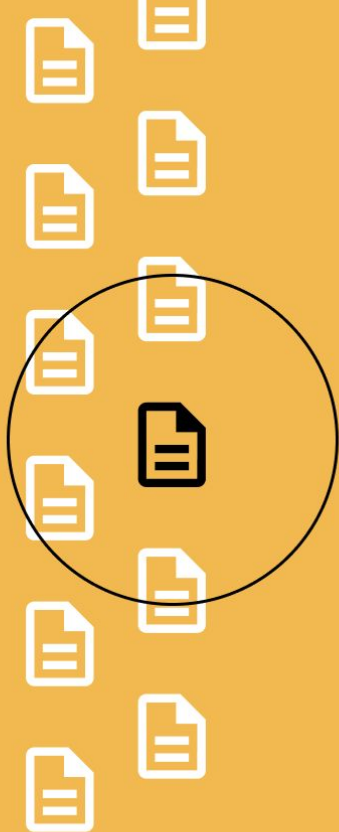
Fonte: “Vozes da Minha Cabeça”

- ASCENCIO, Ana Fernanda Gomes; CAMPOS, Edilene Aparecida Veneruchi. **Fundamentos da Programação de Computadores**. 3ª Edição. São Paulo. Pearson, 2012.
- SEBESTA, R. **Conceitos de linguagens de programação**. 9ª Edição. Porto Alegre: Bookman, 2011.
- MENDES, Douglas Rocha. **Programação Java Com Ênfase em Orientação a Objetos**. São Paulo: Novatec, 2009.
- KATHY, Sierra; BATES, Bert. **Use a Cabeça! Python**. 2ª Edição. Rio de Janeiro: Alta Books, 2018.

Metodologia

- Aulas expositivas e práticas (recursos multimídia)
- Trabalhos individuais e em duplas
- Exercícios individuais em sala e no Laboratório
- Avaliações (Prova)

AVALIAÇÃO



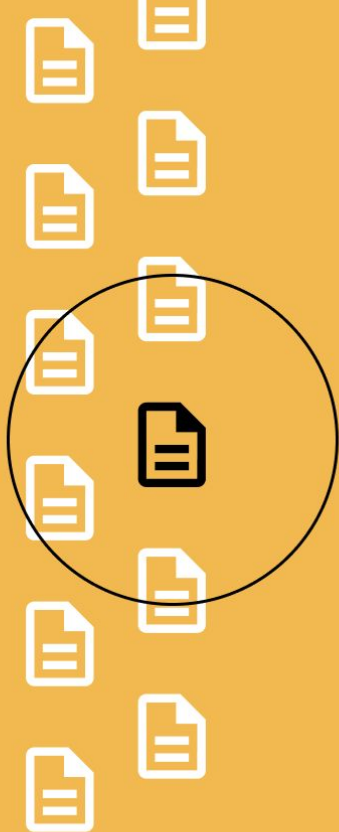
70% - Prova
30% - Atividades

Atingiu 70%?



Não atingiu 70%?

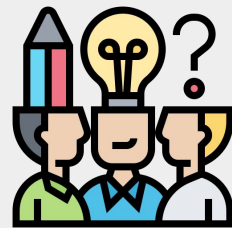




REGRAS DO ROLÊ

- ~~Não ser preso~~
- Celular no silencioso
- Participação
- Respeitar o coleguinha
- Duro nos assuntos e leve nas pessoas
- Utilizar os recursos corretamente
- **CONVERSAR COM A PROF**
- Sem comida no laboratório
- Dar crédito apropriado quando usar trabalhos de terceiros

Discutindo a Relação



- Expectativas para o curso?
- O que quero aprender?
- Sugestões?!
- Preocupações?!

Orientação a objetos

Motivação

- Cada uma das diversas linguagens de programação segue diferentes **paradigmas**.
- Um deles é justamente o da **orientação a objetos**, que possui um padrão em constante evolução, principalmente no que se refere à segurança da informação e ao reaproveitamento de código.

Important!

Segurança

Praticidade

Reuso

Manutenibilidade

Motivação

- A programação orientada a objetos tem o propósito principal de aproximar o mundo lógico da programação e o mundo em que vivemos.
- **Objetos, classes, atributos e métodos** são o que definem o paradigma de programação orientada a objetos.

Objeto



Classe



Objeto

- Objetos são como componentes de um sistema.
- Por exemplo: um objeto pode representar uma casa com suas propriedades, como quantidades de cômodos, metragem e acessos.

Important!

É um modelo de forma concreta coisas do mundo real

Classe

- As classes de programação são projetos de um objeto, aonde têm características e comportamentos, ou seja, permite armazenar propriedades e métodos dentro dela.

Important!

O nome de uma classe é sempre iniciado com letras maiúsculas por convenção.

Atributos

- São as características individuais das classes.
- Exemplo: Uma casa azul com 200m²
 - Atributo: `.cor = azul`
 - Atributo: `.metragem = 200m2`

Métodos

- Identificam os **comportamentos e ações** de um objeto da classe que vai fazer algo com esses dados.
- Exemplo: método chamado `__init__()`
 - Toda vez que um objeto “Casa” for criado, o `__init__()` define o estado inicial desse objeto.



Bora praticar?

Objeto

- Classe
 - Atributos
 - Métodos

Outras infos

- Os **objetos** criados **trabalham juntos** para atingir os **objetivos definidos** no domínio do problema.
- Um P00 é composto por muitos objetos que **interagem entre si**, num processo de **colaboração**, com um objetivo definido.
- Assim os objetos **criam outros objetos** e **chamam métodos** de outros objetos que realizam algum processamento e retornam valores para os outros objetos **(Ficou confuso? Depois piora)**.

Vantagens de usar POO

Confiável

A geração de código baseado no conceito de objetos e classes fornece uma grande independência ao programa.

Oportuno

A criação paralela de código torna todo o processo bastante ágil.

Ajustável

Ao atualizar uma parte pequena, o conceito de herança garante que, automaticamente, todas as partes que utilizarem tal método sejam beneficiadas.

Vantagens de usar POO

Extensível

Não é preciso “reinventar a roda”, reescrevendo o código. Isso confere maior capacidade de estender um sistema já existente.

Reutilizável

Um mesmo objeto pode ser utilizado em aplicações diferentes, desde que sejam compatíveis.

Natural

O conceito de POO que traz para a programação o mundo concreto, tal qual vemos no dia-a-dia, faz com que se ganhe naturalidade.

Linguagens que utilizam POO

- TODAS!!!
- Os exemplos mais emblemáticos de linguagens puramente POO, certamente, são o Java e JavaScript.
- Linguagens orientadas a objeto significativas incluem C++, C#, Python, Lua, PHP, Ruby, Swift

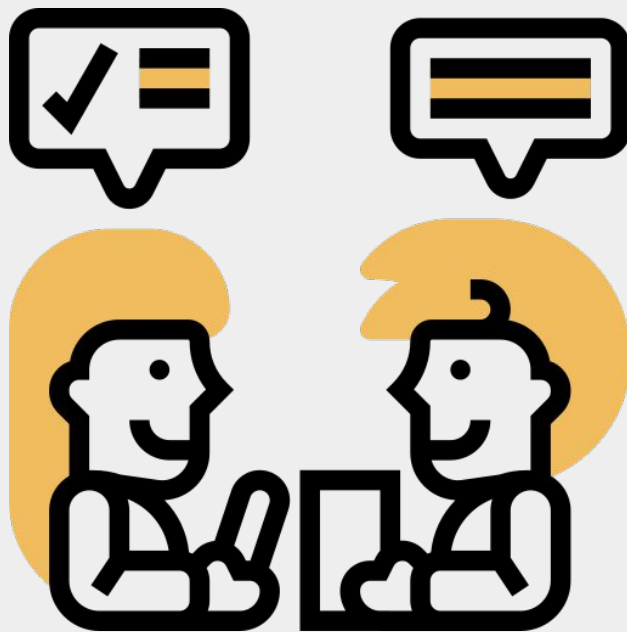
Prática

— Introdução a Programação Orientação a Objetos I

Prática

Individual

- **Acumulativo:** você aplicará o conceito de cada prática em seu projeto pessoal, livre
- **Salvo por mérito:** a entrega do projeto final poderá te ajudar a passar na disciplina.
- **Entregas:** cronograma no class.
- **Objetivo:** praticar o paradigma e montar repositório.



Para aprofundar!

Escolher 1 artigo relacionado ao tema e realizar leitura para a próxima aula

2018 > Atual

Pesquisa no
Google Scholar

Trazer
**informações
relevantes e
impressões**

Introdução a Programação Orientação a Objetos I