

Educação Profissional Paulista

Técnico em
**Desenvolvimento
de Sistemas**

Pilares da Programação Orientada a Objetos

**Implementação de classes e objetos e
conceituação de encapsulamento**

Aula 2

Código da aula: [SIS]ANO1C3B2S10A2

Exposição



Objetivo da aula

Praticar a atribuição de classes dentro do contexto de orientação e seus objetos, alinhados à conceituação de encapsulamento.



Competências da unidade (técnicas e socioemocionais)

- Conhecer frameworks de desenvolvimento ágeis, utilizando tecnologias de CI e CD que trabalham em conjunto com a segurança do ambiente funcional e as entregas divididas em partes, agregando valor ao negócio de forma rápida;
- Praticar a curiosidade e a resiliência de sistemas computacionais.



Recursos didáticos

- Recurso audiovisual para exibição de vídeos e imagens;
- Caderno para anotações.



Duração da aula

50 minutos.

Encapsulamento em Python: protegendo os dados

Encapsulamento é crucial para a **segurança dos dados** e a **integridade da aplicação**, permitindo que os objetos controlem o acesso a seus atributos e métodos internos.

- ✓ Praticar o encapsulamento dentro dos projetos de software.
- ✓ Conhecer a utilização de atributos e métodos ao definir classes em um projeto.
- ✓ Praticar o desenvolvimento de software utilizando a linguagem de programação Python.

Exposição



© Getty Images

O que é encapsulamento?

Encapsulamento é um dos **princípios fundamentais** da Programação Orientada a Objetos (POO).

Ele se refere à prática de **ocultar os detalhes internos** de como uma classe funciona, expondo apenas o necessário para o uso externo. Isso significa que o estado interno de um objeto (seus atributos) é **mantido privado**, e o acesso a esses dados é feito por meio de métodos públicos.

Por que é importante?

Encapsulamento é importante por várias razões:



Segurança

Previne que o estado interno do objeto seja alterado de forma inesperada ou inadequada.



Manutenção

Facilita a manutenção do código, pois as mudanças internas em uma classe não afetam outras partes do código que a utilizam.



Abstração

Permite focar em como usar um objeto, sem a necessidade de entender os detalhes complexos de sua implementação interna.

Exposição

Métodos para acessar e modificar dados

Getters e Setters são métodos que permitem **ler** e **alterar o valor** de atributos privados de uma classe de maneira controlada.

- **Getters (acessores):** são métodos usados para acessar o valor de um atributo. Eles permitem que você obtenha o estado de um objeto sem expor diretamente seus atributos internos.
- **Setters (modificadores):** são métodos usados para definir ou alterar o valor de um atributo. Eles permitem impor regras para a alteração de dados, como validar dados antes de alterar um atributo.



Importante

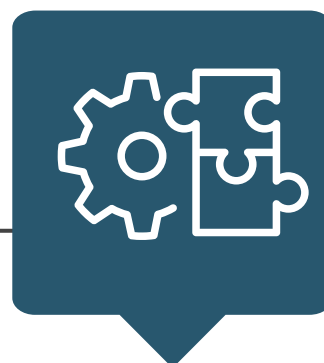
Esses métodos são fundamentais para **manter o princípio de encapsulamento**, garantindo que os dados internos de um objeto só possam ser acessados ou modificados de maneira segura e controlada.

Vamos
fazer uma
atividade

Aplicação prática
dos conceitos de
POO em Python.

 **40 minutos**

Definição de classes e objetos em Python



Materiais necessários

Para realização desta atividade em laboratório, deverão ser utilizados os materiais:

- Caderno ou folhas de sulfite;
- Lápis ou caneta para anotação;
- Recursos computacionais (computador ou notebook para prática).



Passo a passo

1. Leia com atenção cada uma das atividades propostas no documento.
2. Antes de iniciar as atividades práticas, realize esboços no papel para definir sua intenção e resultado.
3. Utilize as ferramentas de desenvolvimento em Python para prática.



Faça download da explicação da atividade

Hoje desenvolvemos:

- 1 Revisão dos conceitos de **classes e objetos** dentro do ambiente de desenvolvimento de software.
- 2 Habilidade prática com o **desenvolvimento de classes** para agrupar funções dentro do código.
- 3 Habilidade prática com o desenvolvimento de **códigos na linguagem Python**, aplicando os conceitos de Programação Orientada a Objetos.

O que nós
**aprendemos
hoje?**

© Getty Images



Saiba mais

Estamos avançando no conceito de **Programação Orientada a Objetos**. Que tal conhecermos um pouco mais a fundo?

HASHTAG PROGRAMAÇÃO. *Como sair do zero em classes no Python* – Self e Init explicados. Disponível em:

<https://www.youtube.com/watch?v=gomDSZaay3E>.
Acesso em: 23 fev. 2024.

Referências da aula

HENRIQUE, J. ALURA. *POO: o que é programação orientada a objetos?* Alura, 18 set. 2023.
Disponível em: <https://www.alura.com.br/artigos/poo-programacao-orientada-a-objetos>.
Acesso em: 23 fev. 2024.

HASHTAG PROGRAMAÇÃO. *Como sair do zero em classes no Python – Self e Init explicados.*
Disponível em: <https://www.youtube.com/watch?v=gomDSZaay3E>. Acesso em: 23 fev. 2024.

Identidade visual: imagens © Getty Images

Educação Profissional Paulista

Técnico em
**Desenvolvimento
de Sistemas**