

## Desenvolvimento de um projeto em Python

Você deverá desenvolver um **sistema em Python**. O importante é que o software **simule um pequeno gerenciamento de informações** e que obrigatoriamente utilize os conceitos abaixo.

### Requisitos do sistema

#### 1. Menu interativo

- ✓ O programa deve exibir um menu com opções, permitindo ao usuário navegar entre funcionalidades usando while e if/else.
- ✓ Exemplo: cadastrar, listar, editar, excluir, consultar, sair.

#### 2. Estruturas de dados

- ✓ Utilize **listas** para armazenar os objetos criados (clientes, produtos, veículos, etc.).
- ✓ Utilize **tuplas** para representar dados fixos (ex.: categorias, status, níveis de acesso).

#### 3. Funções

- ✓ Cada ação do sistema deve ser implementada em uma função específica (ex.: cadastrar(), listar(), buscar()).

#### 4. Exceções

- ✓ O sistema deve tratar erros com try/except, como:
  - Consultar um item que não existe.
  - Inserir valores inválidos.

#### 5. Classes e objetos

- ✓ Crie uma classe principal (ex.: Produto, Aluno, Paciente, Veículo), com atributos privados.
- ✓ Use @property e @setter para controlar pelo menos um atributo (ex.: preço, nota, saldo, idade).

#### 6. Herança

- ✓ Crie uma classe base (ex.: Pessoa, Item, Usuário) e pelo menos duas subclasses que herdem dela, com atributos adicionais específicos.

## 7. Requisitos extras (opcional)

- ✓ Permitir que cada objeto relacionado a um usuário ou cliente tenha uma lista de registros associados (ex.: compras realizadas, livros emprestados, consultas médicas).

---

### Temas:

Loja, clínica, consultório médico, Jogo, oficina, restaurante, estacionamento, transporte, etc.

### Equipe

O trabalho deverá ser feito em no máximo três estudantes...

### Observações

1 - Este trabalho tem como objetivo integrar diversos conceitos fundamentais de programação em Python:

- ✓ Estruturas de controle (if, for, while)
- ✓ Estruturas de dados (listas, tuplas e dicionários)
- ✓ Funções
- ✓ Tratamento de exceções
- ✓ Programação orientada a objetos (atributos, property, setter)
- ✓ **Herança entre classes.**

2 – Após a entrega do trabalho, deverá ser explicado para o professor as estruturas básicas, implementação de funções, estruturas de dados, classes. Cada um da equipe deverá explicar.

### Critério avaliativo:

- Funcionamento do software: **4.00**
- Estruturas de código, nomeação de variáveis, funções, métodos, classes, objetos – **2.00**
- Apresentação do projeto e explicação do código: **4.00**