

# Trilha (6 meses) - Lusciana

Esta trilha de aprendizado foi criada especialmente para iniciantes que desejam explorar os fundamentos de programação e desenvolvimento web por meio da linguagem Python. Ao longo de seis meses, o participante irá explorar desde os conceitos básicos de Python até a criação de aplicações web completas, integrando conhecimentos de front-end e back-end.

## Objetivos

1. **Compreensão de Python:** Ensinar os fundamentos da linguagem Python, incluindo sintaxe, estruturas de controle, funções e manipulação de dados.
2. **Lógica de Programação:** Desenvolver habilidades de raciocínio lógico através da resolução de algoritmos e problemas práticos.
3. **Desenvolvimento Front-End:** Introduzir HTML, CSS e JavaScript, capacitando o participante a criar interfaces interativas e estilizadas.
4. **Integração com Back-End:** Aprender a construir aplicações web utilizando Flask, além de entender como conectar front-end e back-end, trabalhar com bancos de dados e APIs.
5. **Programação Orientada a Objetos e Prática em Projetos Reais:** Proporcionar experiências práticas por meio de projetos completos, permitindo que os alunos apliquem o conhecimento adquirido de forma criativa e prática.
6. **Preparação para o Mercado de Trabalho:** Preparar o participante para ingressar no mercado de trabalho, com foco em habilidades técnicas, construção de portfólio e estratégias de apresentação profissional.

## 01: Fundamentos de Python

- **Semana 1: Introdução ao Python**
  - Instalar Python e configurar um ambiente.
  - Aprender sobre variáveis, tipos de dados (int, float, string, boolean) e operadores.
  - **Exercícios:** Criar um script simples que peça ao usuário para inserir seu nome e idade, e retorne uma mensagem.
  - Lista de exercícios para casa.
- **Semana 2: Estruturas de Controle**
  - Entender condicionais (`if`, `elif`, `else`) e laços de repetição (`for`, `while`).
  - **Exercícios:** Criar um jogo de adivinhação de números.
  - Lista de exercícios para casa.
- **Semana 3: Estruturas de Dados**
  - Aprender sobre listas e dicionários.

- **Exercícios:** Criar uma lista de tarefas e implementar funções para adicionar e remover tarefas.
- Lista de exercícios para casa.
- **Semana 4: Funções**
  - Aprender a definir e chamar funções.
  - **Exercícios:** Criar funções para calcular a soma e a média de uma lista de números.
  - Lista de exercícios para casa.

## 02: Lógica de Programação

- **Semana 5: Algoritmos Básicos**
  - Introdução ao que é um algoritmo e como estruturá-los.
  - **Exercícios:** Resolver problemas simples de lógica, como FizzBuzz.
  - Lista de exercícios para casa.
- **Semana 6: Recursão e Problemas Simples**
  - Compreender o conceito de recursão.
  - **Exercícios:** Implementar a sequência de Fibonacci usando recursão.
  - Lista de exercícios para casa.
- **Semana 7: Algoritmos de Busca**
  - Aprender sobre busca linear e binária.
  - **Exercícios:** Implementar busca linear em uma lista.
  - Lista de exercícios para casa.
- **Semana 8: Revisão e Prática**
  - Revisar os conceitos aprendidos e resolver exercícios adicionais para fixação.
  - Lista de exercícios para casa.

## 03: Introdução ao Front-End

- **Semana 9: HTML e CSS**
  - Aprender os fundamentos de HTML e CSS.
  - **Exercícios:** Criar uma página simples com HTML e estilizar com CSS.
  - Lista de exercícios para casa.
- **Semana 10: Introdução ao JavaScript**
  - Aprender os conceitos básicos de JavaScript.
  - **Exercícios:** Criar um pequeno script para interagir com a página HTML.
  - Lista de exercícios para casa.
- **Semana 11: Integração com Python (Flask)**
  - Introdução ao Flask para criar aplicações web simples.

- **Exercícios:** Criar um app Flask que exiba uma página HTML.
- Lista de exercícios para casa.
- **Semana 12: Projeto Simples**
  - Criar um projeto simples integrando HTML, CSS, JS e Flask.

## 04: Introdução ao Back-End

- **Semana 13: Continuar com Flask**
  - Aprender a criar rotas e passar dados do back-end para o front-end.
  - **Exercícios:** Adicionar rotas para diferentes páginas no seu app Flask.
  - Lista de exercícios para casa.
- **Semana 14: Banco de Dados Simples**
  - Aprender sobre SQLite e como integrá-lo com Flask.
  - **Exercícios:** Criar um CRUD básico (adicionar, listar, editar e deletar) com Flask.
  - Lista de exercícios para casa.
- **Semana 15: APIs Básicas**
  - Introdução à criação de APIs REST com Flask.
  - **Exercícios:** Criar uma API que retorne uma lista de tarefas.
  - Lista de exercícios para casa.
- **Semana 16: Revisão e Melhoria dos Projetos**
  - Melhorar os projetos criados até agora e adicionar novas funcionalidades.
  - Lista de exercícios para casa.

## 05: Programação Orientada a Objetos

- **Semana 17: Introdução à Orientação a Objetos**
  - Compreender os conceitos de classes e objetos.
  - **Exercícios:** Criar uma classe simples (ex: Carro) com atributos e métodos.
  - Lista de exercícios para casa.
- **Semana 18: Herança e Polimorfismo**
  - Aprender sobre herança e polimorfismo em Python.
  - **Exercícios:** Criar uma classe base e subclasses, demonstrando herança.
  - Lista de exercícios para casa.
- **Semana 19: Encapsulamento e Métodos Especiais**
  - Compreender o conceito de encapsulamento e aprender sobre métodos especiais (como `__init__`, `__str__`).
  - **Exercícios:** Implementar métodos especiais em suas classes.
  - Lista de exercícios para casa.

- **Semana 20: Revisão de OOP e Integração com Projetos**
  - Revisar os conceitos de OOP e aplicá-los em projetos já existentes.
  - Lista de exercícios para casa.

## 05: Projetos Práticos

- **Semana 21: Aplicativo de Tarefas**
  - Criar um aplicativo de tarefas completo com front-end e back-end integrados (sugestão).
- **Semana 22: Portfólio**
  - Criar um site pessoal para mostrar seus projetos e habilidades (sugestão).

## 06: Aperfeiçoamento e Preparação

- **Semana 23: Introdução a Testes**
  - Aprender sobre testes básicos em Python.
  - **Exercícios:** Escrever testes para os projetos.
  - Lista de exercícios para casa.
- **Semana 24: Melhorias de Performance**
  - Conceitos básicos de otimização e boas práticas de programação.
  - Lista de exercícios para casa.
- **Semana 25: Deploy de Projetos**
  - Aprender a hospedar um projeto em uma plataforma como Heroku, Vercel ou Github.
  - Lista de exercícios para casa.
- **Semana 26: Preparação para o Mercado**
  - Criar um currículo e perfis em plataformas profissionais.
  - Preparar-se para entrevistas técnicas.