

## Introdução ao Python

Python é uma linguagem de programação de alto nível, interpretada e de propósito geral. Criada por Guido van Rossum e lançada em 1991, ela é conhecida por sua sintaxe simples e legibilidade. Python é amplamente usada em áreas como ciência de dados, automação, desenvolvimento web e inteligência artificial.

## Estruturas de Controle

As estruturas de controle permitem que um programa tome decisões e repita ações. Entre elas estão o if, for e while. Por exemplo: `if x > 0: print("Positivo")` `for i in range(5): print(i)`

## Funções em Python

Funções são blocos de código reutilizáveis. São definidas com a palavra-chave `def`. Exemplo: `def saudacao(nome): print(f"Olá, {nome}!")` `saudacao("João")` Elas podem receber parâmetros e retornar valores usando `return`.

## Programação Orientada a Objetos

Python suporta POO (Programação Orientada a Objetos). Usa-se classes e objetos para modelar entidades do mundo real. Exemplo: `class Pessoa: def __init__(self, nome, idade): self.nome = nome self.idade = idade def falar(self): print(f"Olá, meu nome é {self.nome}")` `p1 = Pessoa("Maria", 25)` `p1.falar()`

## Manipulação de Arquivos

Python permite ler e escrever arquivos facilmente: `with open("arquivo.txt", "w") as f: f.write("Olá, mundo!")` `with open("arquivo.txt", "r") as f: conteudo = f.read() print(conteudo)` O uso do `'with'` garante que o arquivo será fechado corretamente.

## Módulos e Pacotes

Um módulo é um arquivo Python contendo definições e instruções. Podemos importar módulos com `'import math'` ou funções específicas com `'from math import sqrt'`. Pacotes são coleções de módulos organizados em diretórios.

## Expressões Lambda e List Comprehension

Expressões lambda são funções anônimas, usadas para operações simples: `quadrado = lambda x: x ** 2` `print(quadrado(4))` List comprehension permite criar listas de forma concisa: `numeros = [x for x in range(10) if x % 2 == 0]` `print(numeros)`

## Bibliotecas Populares

Python possui uma vasta coleção de bibliotecas: - NumPy: cálculos numéricos - Pandas: manipulação de dados - Matplotlib: visualização de dados - Flask e Django: desenvolvimento web - TensorFlow e PyTorch: aprendizado de máquina

## Tratamento de Exceções

Exceções tratam erros que ocorrem durante a execução do programa. `try: x = int(input("Digite um número: ")) except ValueError: print("Valor inválido!") finally: print("Execução finalizada.")` O bloco 'finally' sempre é executado, ocorrendo ou não erro.

## Conclusão

Python é uma linguagem poderosa, flexível e ideal tanto para iniciantes quanto para profissionais. Sua comunidade ativa e vasta gama de bibliotecas tornam-na uma das linguagens mais usadas no mundo. Este PDF pode ser usado para treinar um chatbot que responda perguntas sobre seus conteúdos.