

# Apostila

## **PARADIGMAS DE LINGUAGENS DE PROGRAMAÇÃO EM PYTHON**

Profº: Sebastião Rogério



- **O que são paradigmas de programação?**

Imagine que você está construindo uma casa. Assim como um arquiteto precisa de um plano, um programador precisa de paradigmas de programação. Esses paradigmas são como regras que guiam o desenvolvedor na forma de abordar e resolver problemas ao escrever código. Eles moldam a maneira como vemos a estrutura e a execução de um programa.

## **Mas, o que é um programa?**

Pense em um programa como uma receita de bolo. É uma sequência de instruções escritas em uma linguagem de programação que o computador entende e executa. Essas linguagens podem ser divididas em dois grupos principais: interpretadas e compiladas.

### **Linguagens Interpretadas:**

O código é executado diretamente por um interpretador, linha por linha, como se fosse um chef seguindo a receita enquanto cozinha. Exemplos incluem Python, JavaScript e Ruby.

### **Linguagens Compiladas:**

O código é traduzido para uma linguagem que o processador entende antes de ser executado, como se fosse uma receita traduzida para outra língua. Exemplos incluem C, C++ e Rust.

- **Por que aprender paradigmas de programação?**

Dominar diferentes paradigmas é como ter várias ferramentas em sua caixa de ferramentas. Cada uma ajuda a resolver problemas de maneiras diferentes e expande seu conhecimento em tecnologia. Os principais paradigmas se dividem em dois:

### **Programação Imperativa:**

Como um manual detalhado de instruções, o programador fornece o passo a passo para o computador seguir. Exemplos: Cobol, Fortran, Pascal.

### **Programação Declarativa:**

Aqui, o foco é no que deve ser o resultado final, sem se preocupar com os detalhes do processo.

- ❖ **Quais são os principais Paradigmas de Programação?**

### **Procedural:**

Baseado em uma sequência de passos para a execução. 'Como' deve ser feito é o foco.



**Orientado a Objetos:**

Organiza o código em objetos, como peças de Lego, cada uma com suas propriedades e métodos.

**Funcional:**

Foca em funções matemáticas puras, onde uma entrada gera uma saída sem alterar o estado do programa.

**Lógico:**

Baseia-se em avaliações lógico-matemáticas para chegar a um resultado.

**Orientado a Eventos:**

A execução do código depende de eventos, como cliques de mouse ou pressões de teclas.

- **Conceitos Iniciais na Programação**

**Variáveis: O que são e como usá-las**

Pense em variáveis como caixas onde você guarda informações que pode usar mais tarde. Cada caixa tem um nome e um conteúdo que pode variar, como números inteiros, decimais, booleanos ou strings.

**Strings:**

Sequências de caracteres que representam textos. Exemplo: 'Olá, mundo!' é uma string que contém um texto.

**Exemplos de Manipulação de Strings**

Método	Descrição	Exemplo	Resultado
<b>len()</b>	Retorna o tamanho da string	len("Apostila de Python")	18
<b>capitalize()</b>	Retorna a string com a primeira letra maiúscula	"python".capitalize()	'Python'
<b>count()</b>	Conta quantas vezes um caractere aparece na string	"Linguagem Python".count("n")	2
<b>startswith()</b>	Verifica se uma string começa com uma determinada sequência	"Python".startswith("Py")	True

- **Trabalhando com Números**

Os números são como as ferramentas básicas na programação. Eles permitem que façamos cálculos, contagens e muitas outras tarefas. Os principais tipos são:

**Inteiros (int):**

Números sem parte decimal.

**Longos (long):**

Números inteiros maiores que os normais.

**Decimais (float):**

Números com parte decimal.

**Complexos (complex):**

Números com parte real e imaginária.

