

# Lista de Exercícios de Funções em Python

## Nível Básico (Foco em def e return)

1. **Saudação Simples:** Crie uma função chamada saudar que imprima "Olá, mundo!".
2. **Saudação Personalizada:** Crie uma função que receba um nome como parâmetro e imprima "Olá, [nome]!".
3. **Soma de Dois Números:** Crie uma função que receba dois números e retorne a soma deles.
4. **Verificador de Número Par:** Crie uma função que receba um número e retorne True se ele for par e False se for ímpar.
5. **Área de um Retângulo:** Crie uma função que receba a base e a altura de um retângulo e retorne sua área.
6. **Cálculo de Média:** Crie uma função que receba três notas e retorne a média delas.
7. **Conversor de Temperatura:** Crie uma função que receba uma temperatura em Celsius e a converta para Fahrenheit. A fórmula é:  $F = (C * 9/5) + 32$ .

## Nível Intermediário (Foco em return, loops e if com tipagem)

8. **Verificador de Login:** Crie uma função que receba um nome de usuário e uma senha. Se o nome for "admin" e a senha for "12345", retorne "Acesso concedido". Caso contrário, retorne "Acesso negado". Use tipagem para os parâmetros e o retorno.
9. **Encontrar o Maior Número:** Crie uma função que receba uma lista de números e retorne o maior número da lista. Use um loop. Se a lista estiver vazia, retorne None.
10. **Contador de Vogais:** Crie uma função que receba uma string e retorne o número de vogais (a, e, i, o, u) que ela contém.
11. **Calculadora de IMC:** Crie uma função que receba o peso (em kg) e a altura (em metros) e retorne o Índice de Massa Corporal (IMC). A fórmula é:  $IMC = peso / (altura ** 2)$ .
12. **Classificador de IMC:** Baseado no exercício anterior, crie uma função que receba o IMC e retorne a classificação: "Abaixo do peso", "Peso normal", "Sobrepeso" ou "Obesidade". Use múltiplos if/elif/else.
13. **Validador de Senha:** Crie uma função que receba uma senha (string) e retorne True se ela tiver pelo menos 8 caracteres e False caso contrário.
14. **Gerador de Fibonacci:** Crie uma função que receba um número inteiro n e retorne uma lista com a sequência de Fibonacci até o n-ésimo termo.

## Nível Avançado (Foco em \*args, \*\*kwargs e lambda)

15. **Soma de Números Variáveis:** Crie uma função que use \*args para somar uma quantidade qualquer de números.
16. **Detalhes do Produto:** Crie uma função que receba o nome de um produto e use \*\*kwargs para exibir detalhes adicionais como preço, marca e estoque.
17. **Combinador de Argumentos:** Crie uma função que receba um argumento obrigatório (nome) e use \*args e \*\*kwargs para exibir todos os dados de forma clara.
18. **Filtro de Lista com lambda:** Crie uma função que receba uma lista de números e use um loop para retornar uma nova lista apenas com os números pares. Use uma lambda dentro do loop.
19. **Ordenador de Dicionários com lambda:** Crie uma função que receba uma lista de dicionários (cada um com "nome" e "idade"). Use uma lambda e um loop para ordenar a lista pela idade e retornar a lista ordenada.
20. **Retorno de Múltiplos Valores:** Crie uma função que receba um raio de um círculo e retorne a área e a circunferência. Desempacote os valores do retorno em duas variáveis separadas.

## Nível Expert (Foco em try/except e Lógica Complexa)

21. **Validador de Entrada Numérica:** Crie uma função que solicite ao usuário que digite um número inteiro. Use um loop e try/except para garantir que a entrada seja válida. A função deve retornar o número inteiro.
22. **Calculadora Segura:** Crie uma função dividir(a, b) que retorne o resultado da divisão. Use try/except para capturar o erro de divisão por zero e retorne uma mensagem amigável nesse caso.
23. **Autenticador de Usuário:** Crie uma função autenticar(usuario, senha) que verifica se o usuário e a senha existem em um dicionário pré-definido. A função deve retornar True se a autenticação for bem-sucedida e False caso contrário.
24. **Gerenciador de Tarefas Interativo:** Crie um programa com um menu que usa funções para adicionar, remover e listar tarefas em uma lista global. Adicione tratamento de erros para entradas inválidas do usuário.
25. **Analizador de Texto:** Crie uma função que receba um texto e retorne um dicionário contendo a contagem de palavras, a contagem de vogais e a contagem de consoantes.