100 Exercícios de Python (Do Básico à POO)

Parte 1 - Fundamentos e Lógica (30 exercícios)

1 - Verifique se um número é par ou ímpar.

2 - Calcule a média de três notas.

3 - Descubra se um número é positivo, negativo ou zero.

4 - Determine se um ano é bissexto.

5 - Dado um número, verifique se é primo.

6 - Imprima os 20 primeiros números ímpares.

7 - Faça uma tabuada de multiplicação.

8 - Some todos os números entre 1 e 100.

9 - Conte quantos múltiplos de 3 existem entre 1 e 1000.

10 - Descubra se um número é palíndromo (ex: 121, 333).

11 - Simule um caixa eletrônico com notas de 100, 50, 20, 10, 5 e 1.

12 - Inverta os dígitos de um número (ex: 123 -> 321).

13 - Determine se um triângulo é equilátero, isósceles ou escaleno.

14 - Faça um contador de vogais em uma frase.

15 - Dado um número, exiba sua representação binária.

16 - Adivinhe o número: faça um jogo simples com tentativas.

17 - Crie um conversor de temperatura (Celsius <-> Fahrenheit).

18 - Encontre o maior e menor número entre 10 entradas.

19 - Simule uma calculadora básica com if/elif.

20 - Verifique se duas palavras são anagramas.

21 - Conte quantas palavras há em uma frase.

22 - Remova todos os espaços de uma string.

23 - Gere os n primeiros números da sequência de Fibonacci.

24 - Imprima um triângulo de números (ex: Pascal ou crescente).

25 - Verifique se uma frase é um palíndromo ignorando espaços.

26 - Converta um número romano para inteiro.

27 - Dado um número, descubra se é 'perfeito' (soma dos divisores = ele).

28 - Simule uma senha de 4 dígitos com 3 tentativas.

29 - Calcule o fatorial de um número sem usar recursão.

30 - Implemente o algoritmo da divisão com subtrações sucessivas.

Parte 2 - Estruturas de Dados e Funções (20 exercícios)

31 - Função para contar ocorrências de um elemento em uma lista.

32 - Função que retorna o segundo maior elemento de uma lista.

33 - Remova duplicatas de uma lista sem usar set().

34 - Inverta uma lista sem usar .reverse() ou [::-1].

35 - Faça uma função que receba um dicionário e retorne suas chaves ordenadas.

36 - Simule um sistema de cadastro com dicionário (nome, idade).

37 - Crie uma função que verifica se duas listas têm elementos em comum.

38 - Ordene uma lista de dicionários por uma chave.

39 - Implemente uma fila usando listas.

40 - Implemente uma pilha com listas.

41 - Função que conta palavras únicas em um texto.

42 - Função que encontra o valor mais frequente em uma lista.

43 - Transforme uma lista de tuplas (nome, nota) e retorne a média geral.

44 - Função que junta duas listas intercalando os elementos.

45 - Função que retorna a soma dos números pares de uma lista.

46 - Implemente um sistema de inventário de RPG com dicionário.

47 - Função que identifica se há duplicatas em uma lista.

48 - Função que recebe uma lista e devolve a soma acumulada (ex: [1,2,3] -> [1,3,6]).

49 - Crie uma função zip_manual que faz o que zip() faz.

50 - Função que simula um carrinho de compras (add, remove, total).

Parte 3 - Programação mais avançada (25 exercícios)

51 - Simule o jogo da forca com input do usuário.

52 - Simule o jogo do pedra-papel-tesoura com score.

53 - Implemente o algoritmo de ordenação Bubble Sort.

54 - Implemente o algoritmo de busca binária.

55 - Crie um gerador de senhas seguras.

56 - Implemente um validador de CPF.

57 - Implemente o algoritmo de Euclides para MDC.

58 - Crie um cronômetro com contagem regressiva.

59 - Faça um sistema de quiz de múltipla escolha.

60 - Implemente uma função que calcula o tempo entre duas datas.

61 - Converta números para extenso (ex: 123 -> 'cento e vinte e três').

62 - Implemente o crivo de Eratóstenes para achar primos até n.

63 - Faça uma calculadora polonesa reversa.

64 - Crie um sistema de agenda com arquivos .txt.

65 - Implemente um algoritmo de compressão simples (ex: RLE).

66 - Faça parsing de um arquivo .csv simples sem pandas.

67 - Crie um analisador de texto que gera estatísticas (média de palavras, frases, etc).

68 - Leia e grave dados em um arquivo .json.

69 - Use enumerate() para numerar palavras de uma frase.

70 - Faça uma função que detecta palavras repetidas em um texto.

71 - Implemente uma calculadora de expressões com parênteses.

72 - Crie um sistema de login e senha com armazenamento local.

73 - Valide um número de cartão de crédito com o algoritmo de Luhn.

74 - Implemente um minigame com dados (rolagem aleatória).

75 - Crie um programa que calcula a média dos salários de funcionários de um arquivo.

Parte 4 - Programação Orientada a Objetos (25 exercícios)

76 - Crie uma classe Pessoa com nome, idade e um método falar().

77 - Crie uma classe ContaBancaria com métodos de saque e depósito.

78 - Classe Carro com atributos (marca, modelo, ano) e métodos ligar, desligar.

79 - Classe Aluno com notas e método para calcular média.

80 - Sistema de loja com Produto, Cliente e Pedido.

81 - Classe Retângulo com métodos para calcular área e perímetro.

82 - Classe Triângulo, verifique se é válido e determine o tipo.

83 - Herança: Funcionario -> Gerente, Programador.

84 - Polimorfismo: métodos com mesmo nome mas comportamentos diferentes.

85 - Sobrecarga de operadores (__add__, __str__) para uma classe Vetor.

86 - Crie uma classe Data com validação de data.

87 - Classe ContaCorrente com limite de cheque especial.

88 - Crie uma agenda orientada a objetos (adicionar, remover, buscar contatos).

89 - Classe Livro com método para emprestar e devolver.

90 - Classe Biblioteca que armazena vários livros e usuários.

91 - Classe Calculadora com operações básicas e histórico.

92 - Simule um sistema bancário com POO e persistência em arquivos.

93 - Crie um sistema de batalha RPG entre personagens.

94 - Use propriedades (@property) para controlar acesso a atributos.

95 - Implemente um mini ORM: classes viram objetos armazenáveis em .json.

96 - Classe Animal e subclasses Cachorro, Gato, com métodos próprios.

97 - Sistema de matrícula em curso com Curso e Aluno.

98 - Crie uma classe Timer com start, stop, reset e tempo decorrido.

99 - Crie uma classe Estoque que controle itens, entradas e saídas.

100 - Mini-projeto: Sistema de gerenciamento de tarefas (como um TodoList).