
EXERCÍCIOS – VALE 3,5 PONTOS

Variáveis e Funções

*** TIRAR O PRINT DO CÓDIGO FEITO NO VSCODE E DO RESULTADO DESTES CÓDIGOS ***

ENVIAR MODELO VIA WHATSAPP

** TRABALHO INDIVIDUAL **

ENTREGAR ATÉ DIA 27/06

****IMPORTANTE É TENTAR E APRENDER E NÃO NECESSÁRIAMENTE ACERTAR****

Exercício AND: Verificação de Obrigatoriedade de Voto

Escreva um programa em Python que pergunte a idade do usuário e, com base na resposta, informe se a pessoa é obrigada a votar ou não é obrigada a votar, de acordo com as regras eleitorais brasileiras.

Regras para o exercício:

- Pessoas com idade entre 18 e 65 anos (inclusive) são obrigadas a votar.
- As demais não têm obrigatoriedade.

Exercício FOR: Analisando Pares e Múltiplos de 3

Você recebeu uma lista com vários números inteiros e agora sua missão é escrever um programa em Python que analise cada número da lista, verificando as seguintes condições:

Regras para cada número da lista:

- Se o número for par e múltiplo de 3, o programa deve exibir:
🔗 *"X é par e múltiplo de 3."*
- Se o número for par ou múltiplo de 3 (mas não os dois ao mesmo tempo), o programa deve exibir:
🔗 *"X é par ou múltiplo de 3."*
- Se o número não for nenhum dos dois, o programa deve exibir:
🔗 *"X não é par nem múltiplo de 3."*
- Números = [10, 15, 22, 33, 40, 51, 60, 73, 84, 95]

Exercício IF ELSE 01: Batalha de Sobrevivência – O Personagem Vai Morrer?

Imagine que você está desenvolvendo um jogo de RPG (ou pelo menos fingindo que sabe programar um 🤖).

Seu desafio agora é criar um programa em Python que simule o efeito de um ataque em um personagem.

O programa deve:

1. Pedir ao usuário dois valores numéricos:
 - O valor do dano recebido.
 - A quantidade de saúde atual do personagem.
2. Criar uma função chamada `personagem_morreu(dano, saude)` que:
 - Calcule se, após receber o dano, a saúde do personagem fica igual ou menor que zero.

- Se o personagem morrer, retorne a frase:

🔗 "O personagem morreu 🤖"

- Caso contrário, retorne:

🔗 "O personagem sobreviveu 🦸"

3. Exibir o resultado final na tela.

Exercício IF ELSE 02: Soma Surpresa – Passa de 100?

Crie um programa em Python que:

1. Peça ao usuário dois números inteiros:

- O primeiro número.
- O segundo número.

2. Some os dois números.

3. Verifique:

- Se o resultado da soma for maior que 100, o programa deve exibir:

🔗 "O resultado da soma é X, que é maior que 100."

- Caso contrário, exiba:

🔗 "O resultado da soma é X, que não é maior que 100."

Exercício MATCH CASE: Calculadora Simples de Dias da Semana

Faça um programa em Python que funcione como um "tradutor de números para dias da semana".

Regras:

- O programa deve pedir ao usuário para digitar um número de 1 a 7.
- Depois, usando a estrutura match...case (ou switch-case, se você for das antigas 🤖), o programa deve mostrar na tela o dia da semana correspondente, conforme a tabela:

Número Dia da Semana

1	Domingo
2	Segunda-feira
3	Terça-feira
4	Quarta-feira
5	Quinta-feira
6	Sexta-feira
7	Sábado

- Se o usuário digitar qualquer outro número que não esteja entre 1 e 7, o programa deve exibir:

🔗 "Número inválido. Por favor, digite um número de 1 a 7."

Exercício NOT: A Melhor Banda do Mundo 🎸🎤🎧

Crie um programa em Python que teste os conhecimentos (ou o bom gosto musical) do usuário.

Como o programa deve funcionar:

1. Pergunte ao usuário:

🔗 "Qual a melhor banda do mundo?"

2. Depois, verifique a resposta:

- Se a resposta for exatamente "rush" (tudo em minúsculo, sem frescura), o programa deve responder:

🔗 "Correto, é o Rush!"

- Se a resposta não for "rush", o programa deve exibir:

🔗 "Herege!"

Exercício OR: Verificação de Direito à Fila Prioritária 🏃🏃

Crie um programa em Python que simule uma pergunta sobre prioridade de atendimento em filas, como acontece em bancos, mercados ou repartições públicas.

Regras do programa:

1. Exiba as opções ao usuário:

Pergunta sobre Prioridade

Escolha uma opção abaixo:

1. Idoso

2. Gestante

3. Cadeirante

4. Nenhum destes

2. Peça ao usuário para digitar o número correspondente à sua condição.

3. Depois, o programa deve responder:

- Se o usuário escolher as opções 1, 2 ou 3:

🔗 "☑️ Você tem direito à fila prioritária!"

- Se o usuário escolher a opção 4:

🔗 "🕒 Vá pra fila e aguarde"

- Se o usuário digitar qualquer outro número:

🔗 "❌ Opção inválida. Leia de novo e digite um número de 1 a 4."

Exercício WHILE TRUE 01: Contagem Regressiva – Missão Impossível 🚀 ⌚

Crie um programa em Python que faça uma contagem regressiva a partir de um número informado pelo usuário.

Regras do programa:

1. Peça ao usuário um número inteiro positivo (exemplo: 10, 5, 3...).
2. O programa deve exibir a contagem regressiva, número por número, até chegar a zero (ou abaixo).
3. Quando a contagem terminar, exiba a mensagem:

🔗 "Contagem regressiva concluída!"


4. Use um laço de repetição while True com um break para controlar a parada da contagem.

Exercício WHILE TRUE 02: Contagem Crescente – Subindo Degrau por Degrau

Crie um programa em Python que faça uma contagem crescente, começando do zero até o número escolhido pelo usuário.

Regras do programa:

1. Peça ao usuário um número inteiro positivo.

2. O programa deve começar a contagem em 0 e ir aumentando um por um, até chegar no número informado (incluindo ele).
3. Quando a contagem terminar, o programa deve exibir a mensagem:
 *"Contagem crescente concluída!"*
4. Use um laço while True com um break para controlar a saída da repetição.

Exercício WHILE TRUE 03: Menu Interativo com Loop Infinito

Crie um programa em Python que exiba um menu interativo, onde o usuário pode escolher entre algumas opções ou digitar "sair" para encerrar o programa.

Regras do programa:





1. O programa deve ficar em um loop infinito (while True), exibindo o seguinte menu sempre que necessário:

Menu:

1 - Opção A

2 - Opção B

Digite 'sair' para encerrar.

2. Depois de exibir o menu, o programa deve ler a escolha do usuário.
3. Comportamento esperado:
 - Se o usuário digitar 1, exibir:
 *"Você escolheu a Opção A."*
 - Se o usuário digitar 2, exibir:
 *"Você escolheu a Opção B."*
 - Se o usuário digitar "sair" (de qualquer jeito: "sair", "SAIR", "Sair", etc.), o programa deve exibir:
 *"Encerrando o programa... Tchau!"*
E então finalizar o loop (com um break).
 - Qualquer outra entrada deve resultar em:
 *"Opção inválida. Tente de novo."*

DESAFIO 01: Contador de Números — Quem é Quem?

Crie um programa em Python que peça para o usuário digitar 5 números inteiros e, ao final, informe quantos desses números são:

- Positivos (maiores que zero)
- Negativos (menores que zero)
- Zeros (igual a zero)

Regras obrigatórias:

- Use um laço for para pedir os números.
- Use estruturas condicionais (if, elif, else) para classificar cada número e atualizar os contadores correspondentes.

DESAFIO 02: Calculadora Simples com Tratamento de Erros

Crie um programa em Python que funcione como uma calculadora básica. Ela deve permitir que o usuário escolha uma operação e realize o cálculo com dois números.

Funcionalidades:

1. O programa deve mostrar as opções de operações disponíveis:

1 - Soma (+)

2 - Subtração (-)

3 - Multiplicação (*)

4 - Divisão (/)

2. O usuário pode escolher a operação digitando 1, 2, 3, 4 ou digitar 'sair' para encerrar o programa.
3. Se o usuário digitar uma operação inválida (qualquer coisa diferente de 1, 2, 3, 4 ou 'sair'), o programa deve mostrar uma mensagem de erro e pedir para tentar novamente.
4. Depois que a operação for escolhida corretamente, o programa deve pedir para o usuário digitar dois números (aceitando números decimais).
5. O programa deve tratar erros de entrada, como quando o usuário digita texto no lugar de número.
6. No caso de divisão, o programa deve impedir a divisão por zero, mostrando uma mensagem de erro apropriada.
7. Por fim, o programa deve mostrar o resultado da operação no formato:

num1 operador num2 = resultado