

Atividade 3

Questão 1 - Detector de Números Primos

Enunciado

Você deve criar um programa em Python que solicite ao usuário um intervalo de números e encontre todos os números primos nesse intervalo. O programa deve contar quantos primos foram encontrados e exibi-los de forma organizada.

Objetivos de Aprendizado

- Praticar loops `for` aninhados
 - Usar o bloco `else` com loops
 - Aplicar lógica condicional (`if`/`elif`/`else`)
 - Trabalhar com entrada de dados do usuário
 - Manipular listas em Python
-

Questão 2 - Verificador de Números Pares e Ímpares

Enunciado

Crie um programa em Python que receba um número inteiro do usuário e determine se ele é par ou ímpar. Em seguida, implemente duas aplicações práticas: um sistema para colorir linhas alternadas de uma tabela e um validador simplificado de números de cartão de crédito usando o conceito de paridade.

Objetivos de Aprendizado

- Compreender e aplicar o operador módulo (%)
- Usar estruturas condicionais if/else
- Implementar loops for com lógica condicional
- Aplicar conceitos de paridade em problemas reais
- Trabalhar com strings e manipulação de caracteres

Observação sobre Estruturas de Repetição:

- Use `for` para colorir linhas alternadas da tabela (número fixo de linhas)
 - Use `for` para iterar pelos dígitos do cartão de crédito na validação
 - Use `while` se desejar repetir o programa até o usuário escolher sair
-

Questão 3 - Calculadora de Média de Notas

Enunciado

Desenvolva um programa que solicite ao usuário quantas notas ele deseja inserir, colete essas notas validando se estão no intervalo de 0 a 10, calcule a média aritmética e determine a situação do estudante (Aprovado ≥ 7.0 , Recuperação ≥ 5.0 , ou Reprovado < 5.0).

Objetivos de Aprendizado

- Trabalhar com listas dinâmicas em Python
- Implementar loops for com entrada de dados
- Aplicar validação de entrada de usuário
- Usar funções built-in como `sum()` e `len()`
- Criar estruturas condicionais aninhadas

Observação sobre Estruturas de Repetição:

- Use **for** para coletar as notas (quantidade definida pelo usuário)
 - Use **while** interno para validar cada nota até ser válida (0-10)
 - Use **while** externo se desejar permitir calcular múltiplas médias
-

Questão 4 - Gerador de Tabuada

Enunciado

Crie um programa que receba um número do usuário e exiba a tabuada completa desse número (de 1 a 10). O programa deve apresentar os resultados de forma organizada e permitir que o usuário escolha gerar tabuadas para outros números.

Objetivos de Aprendizado

- Implementar loops for com range específico
- Usar formatação de strings para saída organizada
- Aplicar operações matemáticas básicas
- Criar programas interativos com loops while
- Trabalhar com entrada e validação de dados

Observação sobre Estruturas de Repetição:

- Use **for** para gerar a tabuada de 1 a 10 (range fixo)
 - Use **while** para o loop principal que permite gerar múltiplas tabuadas até o usuário sair
-

Questão 5 - Calculadora de Soma Sequencial

Enunciado

Desenvolva um programa que receba um número N do usuário e calcule a soma de todos os números inteiros de 1 até N (incluindo N). Por exemplo, se $N=5$, o programa deve calcular $1+2+3+4+5=15$. Implemente tanto a versão com loop quanto discuta a fórmula matemática.

Objetivos de Aprendizado

- Usar loops for com range para acumulação
- Implementar operadores de atribuição ($+=$)
- Compreender padrões de soma aritmética
- Comparar soluções iterativas e matemáticas
- Trabalhar com variáveis acumuladoras

Observação sobre Estruturas de Repetição:

- Use `for` para somar de 1 até N (range conhecido)
 - Use `while` como alternativa, mas `for` é mais direto neste caso
 - Use `while` se desejar repetir o programa para diferentes valores de N
-

Questão 6 - Gerenciador de Times de Futebol Brasileiro

Enunciado

Crie um programa que trabalhe com uma lista de times brasileiros de futebol. O programa deve permitir operações como filtrar times por letra inicial, adicionar novos times, remover times existentes, e realizar buscas. Implemente também operações com fatias (slicing) da lista.

Objetivos de Aprendizado

- Criar e manipular listas homogêneas e heterogêneas
- Usar list comprehensions para filtragem
- Implementar métodos de lista (append, remove, insert, pop)
- Aplicar slicing em diferentes contextos
- Iterar sobre listas com for loops

Observação sobre Estruturas de Repetição:

- Use `for` para percorrer e exibir a lista de times
 - Use `for` para buscar times com características específicas
 - Use `while` para o menu interativo principal (se implementado)
-

Questão 7 - Sistema de Audiência de Novelas Brasileiras

Enunciado

Desenvolva um sistema que gerencie um dicionário com novelas brasileiras e suas respectivas audiências. O programa deve permitir adicionar novas novelas, consultar audiências, filtrar sucessos (audiência > 40 pontos), e calcular estatísticas como média de audiência e ranking das novelas.

Objetivos de Aprendizado

- Criar e manipular dicionários em Python
- Usar métodos de dicionário (keys, values, items, update)
- Implementar dictionary comprehensions
- Trabalhar com estruturas de dados aninhadas
- Aplicar operações de filtragem e ordenação

Observação sobre Estruturas de Repetição:

- Use **for** para iterar sobre o dicionário (.items(), .keys(), .values())
 - Use **for** para calcular estatísticas (média, máximo, mínimo)
 - Use **while** para o menu do sistema (adicionar, consultar, etc.)
-

Questão 8 - Catálogo de Comidas Brasileiras por Região

Enunciado

Implemente um sistema que gerencie informações sobre comidas típicas brasileiras usando tuplas. Cada comida deve ser representada como uma tupla contendo (nome, região, preço). O programa deve permitir consultas por região, cálculo de preços médios, e análises estatísticas sobre a culinária brasileira.

Objetivos de Aprendizado

- Criar e manipular tuplas em Python
- Realizar desempacotamento de tuplas
- Usar tuplas em estruturas de dados complexas
- Implementar operações de consulta e filtragem
- Aplicar conceitos de imutabilidade

Observação sobre Estruturas de Repetição:

- Use `for` para iterar pela lista de tuplas (comidas)
 - Use `for` para filtrar comidas por região ou calcular preços médios
 - Use `while` para o menu interativo de consultas (se implementado)
-

Questão 9 - Calculadora Interativa com Loop While

Enunciado

Desenvolva uma calculadora interativa que permaneça em execução até que o usuário escolha sair. A calculadora deve oferecer as quatro operações básicas (adição, subtração, multiplicação, divisão), tratar erros de entrada, e manter um histórico das operações realizadas na sessão atual.

Objetivos de Aprendizado

- Implementar loops while para programas interativos
- Usar break e continue para controle de fluxo
- Aplicar tratamento de exceções (try/except)
- Criar menus de opções para usuário
- Trabalhar com validação contínua de entrada

Observação sobre Estruturas de Repetição:

- Use `while` como loop principal da calculadora (essencial nesta questão)
 - Use `while` para validação de entrada até ser válida
 - Use `for` para exibir o histórico de operações (se implementado)
-

Questão 10 - Sistema de Loja Virtual Brasileira

Enunciado

Desenvolva um sistema completo de loja virtual que permita gerenciar produtos brasileiros. O sistema deve incluir funcionalidades para adicionar produtos (nome, categoria, preço, estoque), listar produtos por categoria, buscar produtos por nome, calcular valor total de compras, e gerar relatórios de vendas. Use listas, dicionários e funções para organizar o código.

Objetivos de Aprendizado

- Integrar múltiplas estruturas de dados (listas, dicionários, tuplas)
- Implementar funções para modularização do código
- Usar list comprehensions para operações complexas
- Aplicar conceitos de CRUD (Create, Read, Update, Delete)
- Desenvolver um projeto completo com múltiplas funcionalidades

Observação sobre Estruturas de Repetição:

- Use **for** para percorrer lista de produtos (exibir, buscar, filtrar)
- Use **for** para calcular totais de vendas e relatórios
- Use **while** para o menu principal do sistema (loop até sair)
- Use **while** para validação de entrada de dados (preços, quantidades)

Funcionalidades Principais:

1. **Adicionar Produtos:** Como lojista, quero adicionar novos produtos ao catálogo para vendê-los
2. **Listar Produtos:** Como cliente, quero ver todos os produtos disponíveis para escolher o que comprar
3. **Buscar Produtos:** Como cliente, quero encontrar um produto específico pelo nome para localizá-lo rapidamente
4. **Gerenciar Estoque:** Como lojista, quero controlar a quantidade de produtos em estoque

5. **Calcular Vendas:** Como lojista, quero calcular o valor total das vendas realizadas

Estrutura Sugerida de Dados:

```
# Exemplo de produto
produto = {
    "nome": "Camiseta do Flamengo",
    "categoria": "Roupas",
    "preco": 89.90,
    "estoque": 15,
    "origem": "Rio de Janeiro"
}

# Lista de produtos da loja
produtos_loja = []
```

Resumo de Quando Usar Cada Loop:

Use **for** quando:

- Iterar sobre sequências conhecidas (listas, dicionários, ranges)
- Número de iterações é determinado ou previsível
- Processar todos os elementos de uma coleção

Use **while** quando:

- Condição de parada depende da entrada do usuário
- Loops de menu interativo ("continuar até escolher sair")
- Validação que deve repetir até entrada ser válida
- Número de iterações é indeterminado