

job_v5.1

July 7, 2025

```
[1]: import csv
import sqlite3
import pandas as pd

# --- FUNÇÕES DE LIMPEZA E TRATAMENTO ---

def remove_ponto(valor):
    """Remove a pontuação de um valor numérico e o converte para inteiro."""
    try:
        # Garante que o valor seja tratado como string antes do replace
        return int(round(float(str(valor).replace('.', '')), 0))
    except (ValueError, TypeError):
        # Retorna 0 ou outro valor padrão se a conversão falhar
        return 0

def corrigir_caracteres_produto(nome_produto):
    """Corrige problemas de codificação comuns em nomes de produtos."""
    if not isinstance(nome_produto, str):
        return nome_produto # Retorna o valor original se não for uma string

    correcoes = {
        'Ã&#o': 'ãõ', # Resolve FeijÃ&#o -> Feijão, PimentÃ&#o -> Pimentão
        'Ã&#c': 'â',
        'Ã&#e': 'é',
        'Ã&#i': 'á',
        'Ã&#3': 'ó',
        'Ã&#u': 'õ', # Resolve MelÃ&#u -> Melão (caso comum também)
    }
    # Adicione outras correções se necessário

    for erro, acerto in correcoes.items():
        nome_produto = nome_produto.replace(erro, acerto)

    return nome_produto

# --- SCRIPT PRINCIPAL ---
```

```

DB_FILE = 'produtos.db'
CSV_ENTRADA = 'producao_revisada.csv'
CSV_SAIDA = 'producao_final_corrigido.csv'

try:
    # Abre o arquivo CSV com os dados da produção de alimentos
    # Adicionado encoding='utf-8' para melhor compatibilidade com caracteres_
    ↪especiais
    with open(CSV_ENTRADA, 'r', encoding='utf-8') as file:

        reader = csv.reader(file)
        next(reader) # Pula o cabeçalho

        conn = sqlite3.connect(DB_FILE)
        cursor = conn.cursor()

        cursor.execute('DROP TABLE IF EXISTS producao')
        cursor.execute('''CREATE TABLE producao (
                            produto TEXT,
                            quantidade INTEGER,
                            preco_medio REAL,
                            receita_total INTEGER,
                            margem_lucro REAL
                        )''')

        # Itera sobre as linhas do CSV
        for row in reader:
            # Filtra linhas com quantidade maior que 10
            if int(row[1]) > 10:

                # --- APLICAÇÃO DAS CORREÇÕES ---
                produto_corrigido = corrigir_caracteres_produto(row[0])
                receita_tratada = remove_ponto(row[3])

                # Calcula a margem de lucro
                custo_total = int(row[1]) * float(row[2])
                margem_lucro = round((receita_tratada - custo_total) /_
    ↪receita_tratada, 4) if receita_tratada != 0 else 0

                # Insere a linha com os dados corrigidos na tabela
                cursor.execute('INSERT INTO producao VALUES (?, ?, ?, ?, ?)',
                                (produto_corrigido, row[1], row[2],_
    ↪receita_tratada, margem_lucro))

        conn.commit()
        conn.close()

```

```

    print(f"Job de processamento e carga no banco '{DB_FILE}' concluído com
    ↳Sucesso!")

except FileNotFoundError:
    print(f"ERRO: O arquivo '{CSV_ENTRADA}' não foi encontrado. Verifique o
    ↳nome e o local do arquivo.")
except Exception as e:
    print(f"Ocorreu um erro durante o processamento: {e}")

# --- EXPORTAÇÃO DO RESULTADO PARA CSV ---

print(f"\nIniciando a exportação dos dados tratados para '{CSV_SAIDA}'...")

try:
    conn = sqlite3.connect(DB_FILE)
    query = "SELECT * FROM producao"
    df_final = pd.read_sql_query(query, conn)
    conn.close()

    df_final.to_csv(CSV_SAIDA, index=False, sep=';', decimal=',',
    ↳encoding='utf-8-sig')

    print(f"Arquivo '{CSV_SAIDA}' gerado com sucesso!")
    print(f"Total de {len(df_final)} linhas exportadas.")

except Exception as e:
    print(f"Ocorreu um erro durante a exportação para CSV: {e}")

print("Job Concluído com Sucesso!")

```

Job de processamento e carga no banco 'produtos.db' concluído com Sucesso!

Iniciando a exportação dos dados tratados para 'producao_final_corrigido.csv'...

Arquivo 'producao_final_corrigido.csv' gerado com sucesso!

Total de 12 linhas exportadas.

Job Concluído com Sucesso!