

Tabalho prático - IBD >>>>

Compras Públicas do Governo Federal

Daniel Augusto Castro de Paula
Diego Roberto das Chagas
Gabriel Vilhena Menetryer

Compras Públicas do Governo Federal

Duas planilhas csv importadas:

comprasGOV-mensal-VW_FT_PNCP_COMPRA-latest.csv

comprasGOV-mensal-VW_FT_PNCP_COMPRA_ITEM-latest.csv

Descrição e processamento dos dados

Definições de funções em python utilizando pandas, sqlite3 e matplotlib

```
[2] # consultas para ajudar a visualizar os dados

def fetch(query, conn, formatted=True):
    # executa a consulta e recupera todas as linhas
    cur = conn.cursor()
    cur.execute(query)
    rs = cur.fetchall()

    # extrai os nomes das colunas da descrição do cursor
    columns = [desc[0] for desc in cur.description]

    # retorna um DataFrame com os nomes das colunas
    return pd.DataFrame(rs, columns=columns) if formatted else rs

def show_tables(conn):
    return [x[0] for x in fetch('SELECT tbl_name FROM sqlite_master WHERE type="table"', conn, False)]


def shape(table, conn):
    nrows = fetch(f'SELECT COUNT(*) FROM {table}', conn, False)[0][0]
    ncols = fetch(f'SELECT COUNT(*) FROM pragma_table_info("{table}")', conn, False)[0][0]

    return (nrows, ncols)


def desc(table, conn):
    cur = conn.cursor()
    cur.execute(f'PRAGMA table_info("{table}")')
    columns = [row[1] for row in cur.fetchall()]
```

Descrição e processamento dos dados

Ingestão dos dados em duas tabelas grandes com todo conteúdo:
dados_brutos_compras
dados_brutos_itens

```
# ingestão dos dados

# Caminhos dos arquivos CSV no Drive
source_compras = '/content/drive/MyDrive/comprasGOV-mensal-VW_FT_PNPC_COMPRA-latest.csv'
source_itens   = '/content/drive/MyDrive/comprasGOV-mensal-VW_FT_PNPC_COMPRA_ITEM-latest.csv'

# Conexão com o banco SQLite
conn = sqlite3.connect('tp2.db')
cur = conn.cursor()

# Desativa validação de chaves estrangeiras e ajusta PRAGMA para acelerar ingestão
cur.execute('PRAGMA foreign_keys = 0')
cur.execute('PRAGMA synchronous = OFF')
cur.execute('PRAGMA journal_mode = MEMORY')

# Tamanho de lote para processamento em massa
BATCH_SIZE = 1000

# Função genérica para ingestão de CSV em tabela SQLite
def ingest_csv(source, table_name):
    cur.execute('BEGIN TRANSACTION')
    with open(source, 'r', encoding='utf-8') as f:
        reader = csv.reader(f)
        header = next(reader)
        columns = [f'"{col}" TEXT' for col in header]

        # Cria tabela se não existir
        cur.execute(f'CREATE TABLE IF NOT EXISTS {table_name} ({", ".join(columns)})')

        # Prepara comando de inserção com placeholders
        placeholders = ','.join(['?' * len(header)])
        insert_sql = f'INSERT INTO {table_name} VALUES ({placeholders})'

        # Processa linhas em lotes
        for i in range(0, len(reader), BATCH_SIZE):
            batch = [row for row in reader[i:i+BATCH_SIZE]]
            cur.executemany(insert_sql, batch)

    cur.execute('COMMIT')  
}
```

Limpeza dados

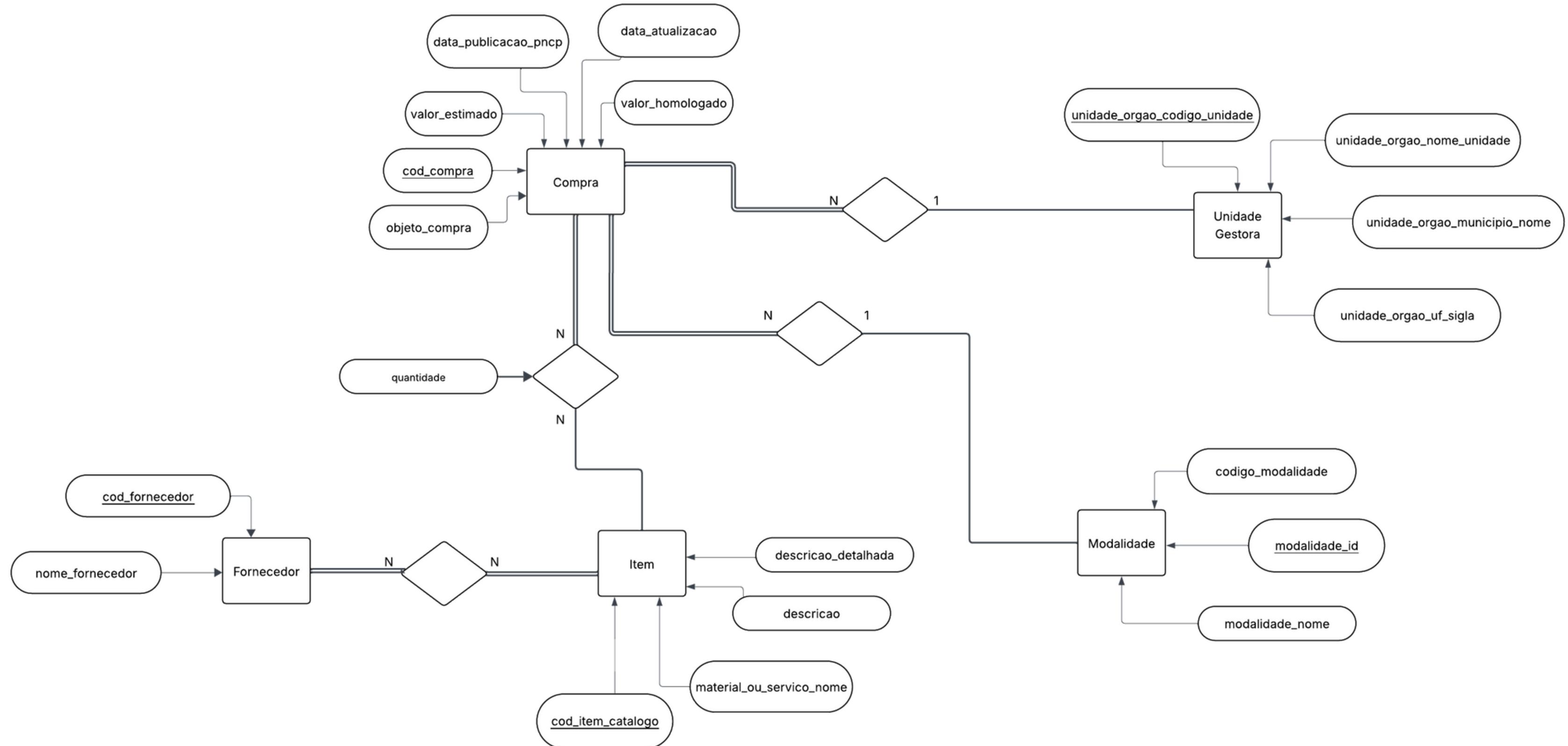
Objetivo dessa limpeza foi impedir que erros de lançamentos manuais ou falhas de sincronia nos csvs prejudicasse nossas análise sobre as compras

- 1- Exclusão de itens que não tinham o código do item preenchido (apagamos os registros das compras também);**
- 2- Deleção de compras sem itens associados;**
- 3- Deleção de itens que não tinham compras correspondentes.**

Modelagem ER

Diagrama ER - TP 2

Compras do Governo Federal



Modelagem relacional

1- Unidade_gestora (unidade_orgao_codigo_unidade, unidade_orgao_nome_unidade, unidade_orgao_uf_sigla, unidade_orgao_municipio_nome)

2- Modalidade (modalidade_id, codigo_modalidade, modalidade_nome)

3- Compra (cod_compra, data_publicacao_pnccp, data_atualizacao, valor_total_estimado, valor_total_homologado, objeto_compra, modalidade_id, unidade_orgao_codigo_unidade)

- modalidade_id referencia Modalidade(modalidade_id) , unidade_orgao_codigo_unidade referencia Unidade_gestora(unidade_orgao_codigo_unidade)

4- Item (cod_item_catalogo, material_ou_servico_nome, descricao, descricao_detalhada)

5- Itens_compra (cod_compra, cod_item_catalogo, quantidade)

- cod_compra referencia Compra(cod_compra), cod_item_catalogo referencia Item(cod_item_catalogo)

6- Fornecedor (cod_fornecedor, nome_fornecedor)

7- Fornecedor_item (cod_fornecedor, cod_item_catalogo)

- cod_fornecedor referencia Fornecedor(cod_fornecedor), cod_item_catalogo referencia Item(cod_item_catalogo)

Consultas e análises

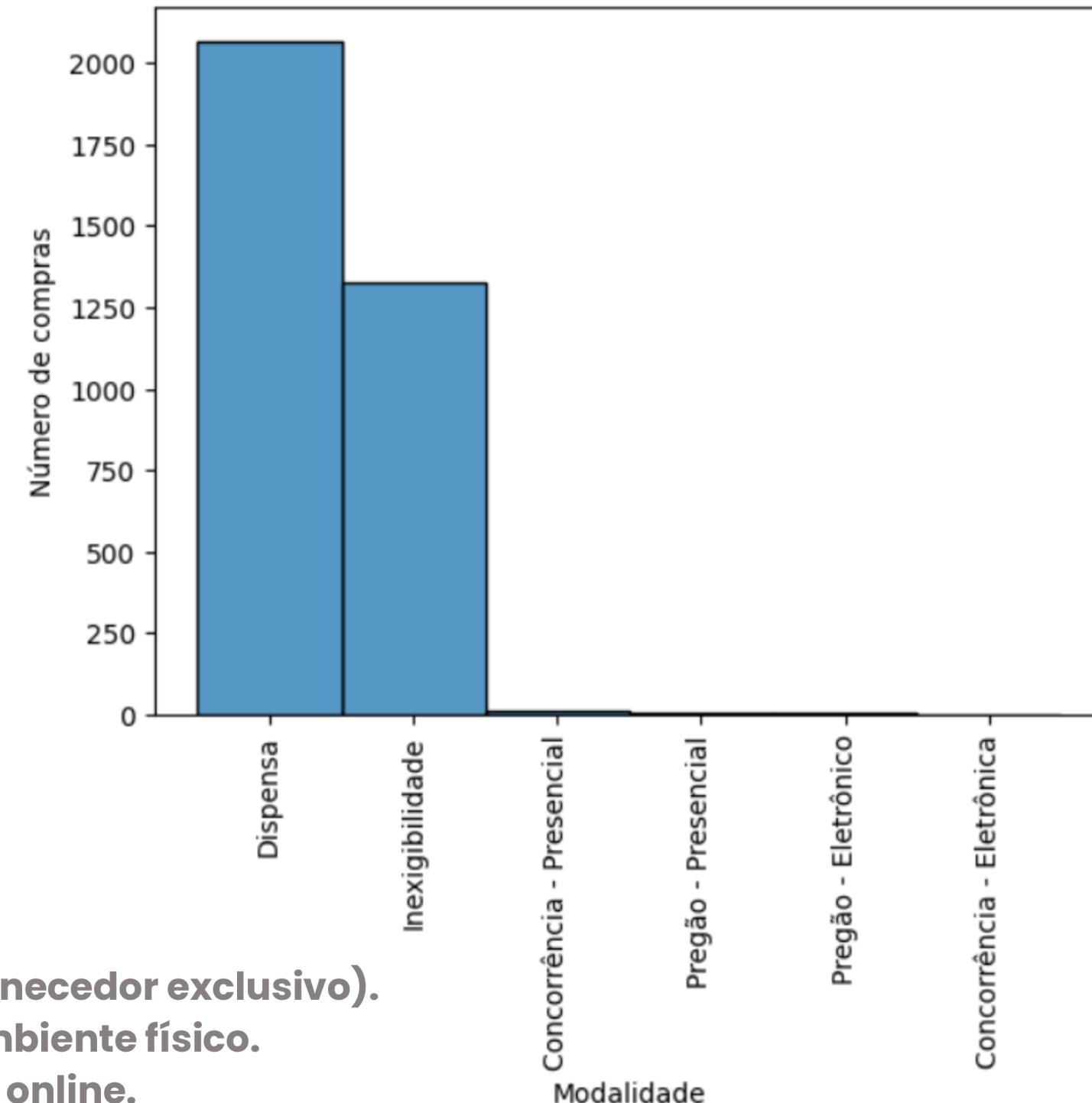
```
[26] # Compras com valor estimado acima de 1 milhão de reais
0s   fetch("""
      SELECT
          cod_compra,
          valor_total_estimado,
          data_atualizacao
      FROM Compra
      WHERE valor_total_estimado > 1000000;
      "", conn)
```

	cod_compra	valor_total_estimado	data_atualizacao
0	6476796	1.200000e+08	2025-06-02 17:47:23
1	6733742	2.827999e+06	2025-06-04 13:03:44
2	6741546	1.442426e+09	2025-06-04 11:48:38
3	7017419	1.717200e+06	2025-06-03 08:13:27
4	7351411	2.196160e+06	2025-06-02 10:37:05
...
161	8726572	1.031656e+06	2025-06-06 16:41:57
162	8727014	3.909000e+06	2025-06-06 17:29:05
163	8727197	3.760800e+06	2025-06-06 18:07:58
164	8727223	4.166863e+06	2025-06-06 18:15:34
165	8727424	1.814400e+06	2025-06-06 19:27:32

166 rows × 3 columns

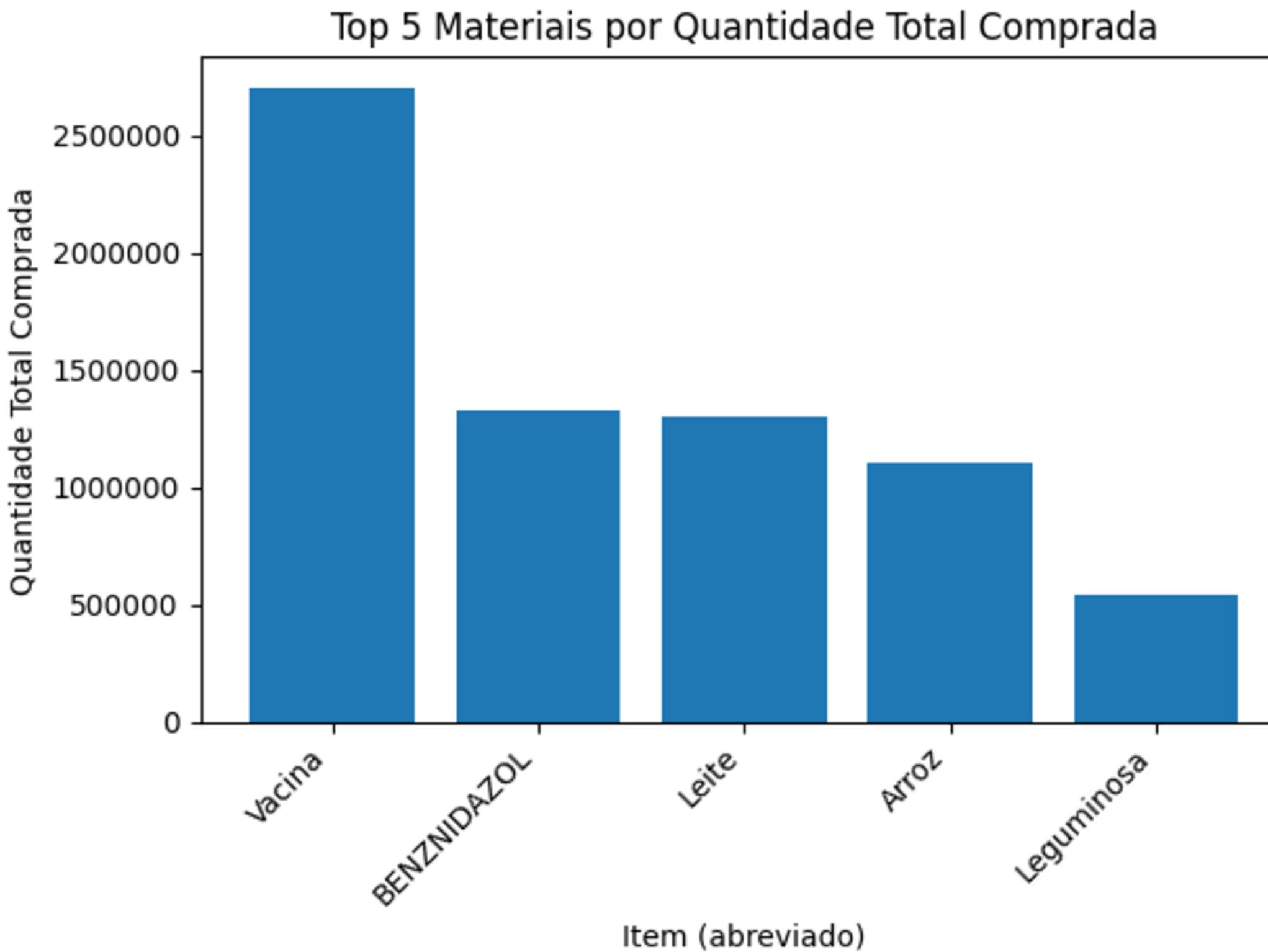
Consultas e análises

```
# Distribuição de compras por modalidade
distro_modalidade = fetch("""
SELECT
    m.modalidade_nome AS modalidade,
    COUNT(*) AS frequencia
FROM Compra c
INNER JOIN Modalidade m
    ON c.modalidade_id = m.modalidade_id
GROUP BY m.modalidade_nome
ORDER BY frequencia DESC;
""", conn, False)
```



- **Dispensa:** contratação direta para compras de pequeno valor, sem disputa formal.
- **Inexigibilidade:** contratação direta quando não há possibilidade de competição (fornecedor exclusivo).
- **Concorrência – Presencial:** licitação aberta a todos, com sessões e propostas em ambiente físico.
- **Concorrência – Eletrônica:** mesma concorrência aberta, mas todo o processo é feito online.
- **Pregão – Presencial:** leilão reverso em sessão pública presencial para aquisição de bens/serviços comuns.
- **Pregão – Eletrônico:** leilão reverso via plataforma online, mais ágil e transparente.
- **Credenciamento:** chamamento público para formar cadastro de fornecedores; contratações posteriores ocorrem por chamada direta.
- **Concurso:** seleção de propostas técnicas, científicas ou artísticas, vencendo o melhor trabalho.

Consultas e análises

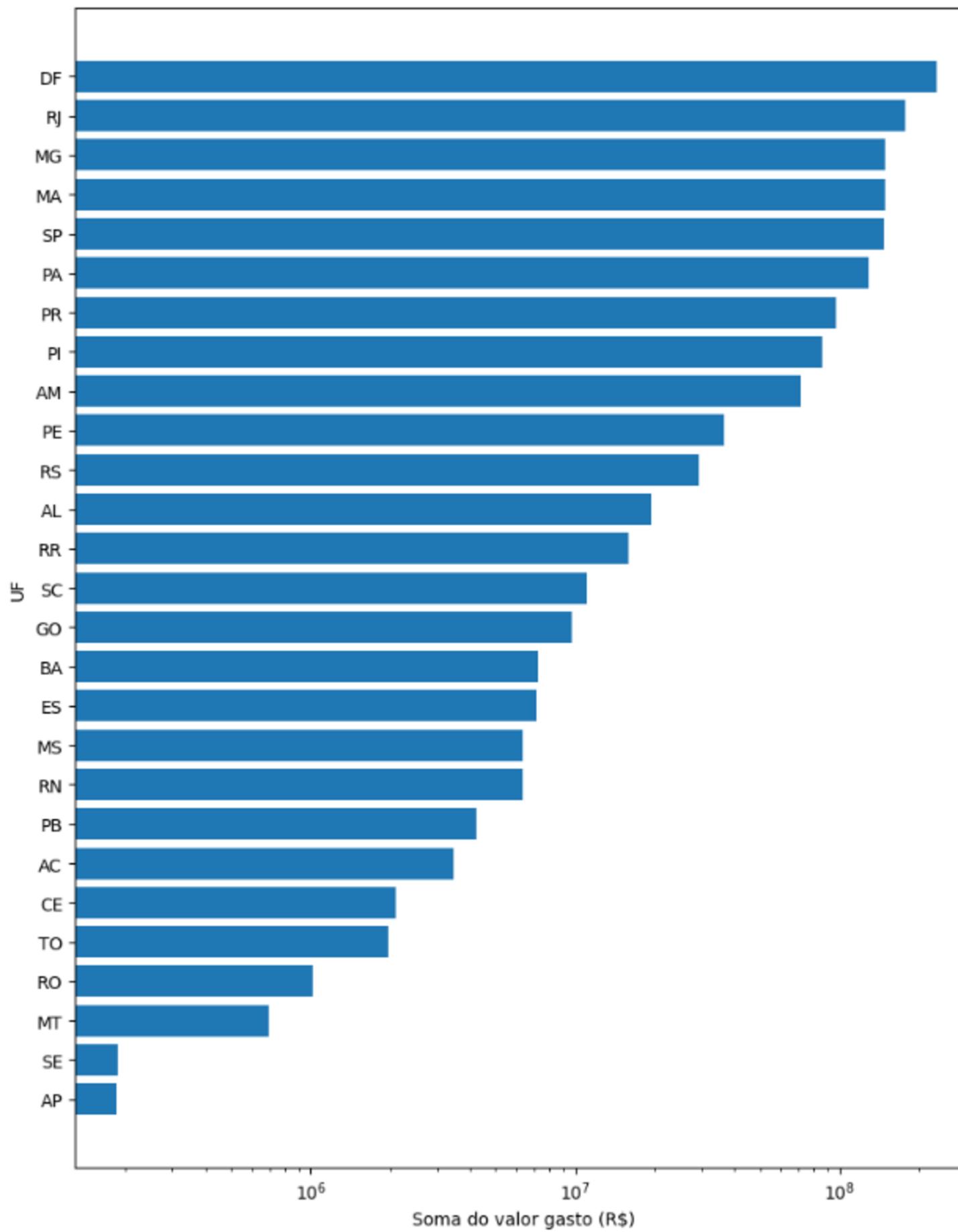


Consultas e análises

Top 5 Fornecedores por Itens Distintos Fornecidos



Consultas e análises



FIM