

# Laboratório Prático: Extrair, Transformar e Carregar Dados do PIB



**Skills  
Network**

**Esforço Estimado:** 60 mins

## Introdução

Neste projeto prático, você colocará em prática as habilidades adquiridas ao longo do curso e criará um pipeline ETL completo para acessar dados de um site e processá-los para atender aos requisitos.

## Cenário do Projeto:

Uma empresa internacional que está procurando expandir seus negócios em diferentes países ao redor do mundo contratou você. Você foi contratado como um engenheiro de dados júnior e tem a tarefa de criar um script automatizado que possa extrair a lista de todos os países em ordem de seus PIBs em bilhões de USD (arredondados para 2 casas decimais), conforme registrado pelo Fundo Monetário Internacional (FMI). Como o FMI libera essa avaliação duas vezes por ano, esse código será usado pela organização para extrair as informações à medida que forem atualizadas.

Os dados necessários parecem estar disponíveis na URL mencionada abaixo:

URL

```
'https://web.archive.org/web/20230902185326/https://en.wikipedia.org/wiki/List_of_countries_by_GDP_%28nominal%29'
```

As informações necessárias devem ser disponibilizadas como um arquivo CSV `Countries_by_GDP.csv`, bem como uma tabela `Countries_by_GDP` em um arquivo de banco de dados `World_Economies.db` com os atributos `Country` e `GDP_USD_billion`.

Seu chefe quer que você demonstre o sucesso deste código executando uma consulta na tabela do banco de dados para exibir apenas as entradas com uma economia superior a 100 bilhões de USD. Além disso, você deve registrar em um arquivo todo o processo de execução chamado `etl_project_log.txt`.

Você deve criar um código Python `'etl_project_gdp.py'` que realize todas as tarefas necessárias.

## Objetivos

Você deve completar as seguintes tarefas para este projeto:

1. Escrever uma função de extração de dados para recuperar as informações relevantes da URL necessária.
2. Transformar as informações de PIB disponíveis de 'Milhões de USD' para 'Bilhões de USD'.
3. Carregar as informações transformadas no arquivo CSV necessário e como um arquivo de banco de dados.
4. Executar a consulta necessária no banco de dados.
5. Registrar o progresso do código com timestamps apropriados.

## Configuração inicial

Antes de começar a construir o código, você precisa instalar as bibliotecas necessárias para isso.

As bibliotecas necessárias para o código são as seguintes:

1. `requests` - A biblioteca usada para acessar as informações a partir da URL.
2. `bs4` - A biblioteca que contém a função `BeautifulSoup` usada para web scraping.
3. `pandas` - A biblioteca usada para processar os dados extraídos, armazená-los nos formatos necessários e comunicar-se com os bancos de dados.
4. `sqlite3` - A biblioteca necessária para criar uma conexão com o servidor de banco de dados.
5. `numpy` - A biblioteca necessária para a operação de arredondamento matemático conforme exigido nos objetivos.
6. `datetime` - A biblioteca que contém a função `datetime` usada para extrair o timestamp para fins de registro.

Como discutido anteriormente, use o seguinte formato de comando em uma janela de terminal para instalar as bibliotecas.

```
python3.11 -m pip install <library_name>
```

Enquanto requests, sqlite3 e datetime vêm incluídos com python, as outras bibliotecas terão que ser instaladas.

#### ► Instalando Bibliotecas

Uma vez que as bibliotecas necessárias estejam instaladas, crie um arquivo etl\_project\_gdp.py no caminho \home\project\.

## Estrutura do código

O código deve ser criado de maneira organizada, de modo que você possa realizar cada tarefa com uma função dedicada. Para referência, você pode copiar e colar a estrutura conforme mostrado abaixo em etl\_project\_gdp.py.

```
# Code for ETL operations on Country-GDP data
# Importing the required libraries
def extract(url, table_attribs):
    ''' This function extracts the required
    information from the website and saves it to a dataframe. The
    function returns the dataframe for further processing. '''
    return df
def transform(df):
    ''' This function converts the GDP information from Currency
    format to float value, transforms the information of GDP from
    USD (Millions) to USD (Billions) rounding to 2 decimal places.
    The function returns the transformed dataframe.'''
    return df
def load_to_csv(df, csv_path):
    ''' This function saves the final dataframe as a `CSV` file
    in the provided path. Function returns nothing.'''
def load_to_db(df, sql_connection, table_name):
    ''' This function saves the final dataframe as a database table
    with the provided name. Function returns nothing.'''
def run_query(query_statement, sql_connection):
    ''' This function runs the stated query on the database table and
    prints the output on the terminal. Function returns nothing. '''
def log_progress(message):
    ''' This function logs the mentioned message at a given stage of the code execution to a log file. Function returns nothing'''
''' Here, you define the required entities and call the relevant
functions in the correct order to complete the project. Note that this
portion is not inside any function.'''
```

## Preliminar: Importando bibliotecas e definindo valores conhecidos

De acordo com o requisito, escreva os comandos em etl\_project\_gdp.py na posição especificada na estrutura do código, para importar as bibliotecas relevantes.

#### ► Clique aqui para a solução

Além disso, você precisa inicializar todas as entidades conhecidas. Estas estão mencionadas abaixo:

##### 1. URL:

```
'https://web.archive.org/web/20230902185326/https://en.wikipedia.org/wiki/List_of_countries_by_GDP_%28nominal%29'
```

##### 2. table\_attribs: Os atributos ou nomes das colunas para o dataframe armazenados como uma lista. Como os dados disponíveis no site estão em milhões de USD, os atributos devem inicialmente ser 'Country' e 'GDP\_USD\_millions'. Isso será modificado na função de transformação mais tarde.

##### 3. db\_name: Como mencionado no cenário do Projeto, 'World\_Economies.db'

##### 4. table\_name: Como mencionado no cenário do Projeto, 'Countries\_by\_GDP'

##### 5. csv\_path: Como mencionado no cenário do Projeto, 'Countries\_by\_GDP.csv'

Você deve registrar o processo de inicialização

#### ► Clique aqui para a solução

## Tarefa 1: Extraíndo informações

A extração de informações de uma página da web é feita usando o processo de web scraping. Para isso, você precisará analisar o link e elaborar a estratégia de como obter as informações necessárias. Os seguintes pontos são importantes para esta tarefa.

1. Inspecione a URL e observe a posição da tabela. Note que até mesmo as imagens com legendas são armazenadas em formato tabular. Assim, na página da web fornecida, nossa tabela está na terceira posição, ou índice 2. Dentre isso, precisamos das entradas sob 'País/Território' e 'FMI -> Estimativa'.
2. Note que há algumas entradas em que a estimativa do FMI é mostrada como '—'. Além disso, há uma entrada no topo chamada 'Mundo', que não precisamos. Separe essa entrada das outras porque esta entrada não tem um hyperlink e todas as outras na tabela têm. Portanto, você pode aproveitar isso e acessar apenas as linhas para as quais a entrada sob 'País/Território' tem um hyperlink associado.

*Note que '—' é um caractere especial e não um hífen geral, '-'. Copie o caractere das instruções aqui para usar no código.*

Assumindo que a função recebe os parâmetros URL e table\_attrbs como argumentos, complete a função extract() no código seguindo os passos abaixo.

1. Extraia a página web como texto.
  - ▶ [Clique aqui para dica](#)
2. Analise o texto em um objeto HTML.
  - ▶ [Clique aqui para dica](#)
3. Crie um DataFrame pandas vazio chamado df com colunas como table\_attrbs.
  - ▶ [Clique aqui para dica](#)
4. Extraia todos os atributos 'tbody' do objeto HTML e, em seguida, extraia todas as linhas da tabela de índice 2 usando o atributo 'tr'.
  - ▶ [Clique aqui para dica](#)
5. Verifique o conteúdo de cada linha, tendo o atributo 'td', para as seguintes condições.
  - a. A linha não deve estar vazia.
  - b. A primeira coluna deve conter um hyperlink.
  - c. A terceira coluna não deve ser '—'.
  - ▶ [Clique aqui para dica](#)
6. Armazene todas as entradas que atendem às condições no passo 5 em um dicionário com chaves iguais às entradas de table\_attrbs. Anexe todos esses dicionários um a um ao dataframe.
  - ▶ [Clique aqui para dica](#)

▶ [Clique aqui para solução](#)

## Tarefa 2: Transformar informações

A função de transformação precisa modificar o 'GDP\_USD\_millions'. Você precisa cobrir os seguintes pontos como parte do processo de transformação.

1. Converta o conteúdo da coluna 'GDP\_USD\_millions' do dataframe df de formato de moeda para números flutuantes.
  - ▶ [Clique aqui para dica](#)
2. Divida todos esses valores por 1000 e arredonde para 2 casas decimais.
  - ▶ [Clique aqui para dica](#)
3. Modifique o nome da coluna de 'GDP\_USD\_millions' para 'GDP\_USD\_billions'.
  - ▶ [Clique aqui para dica](#)

▶ [Clique aqui para solução](#)

## Tarefa 3: Carregando informações

O processo de carregamento para este projeto é bifásico.

1. Você deve salvar o dataframe transformado em um arquivo CSV. Para isso, passe o dataframe df e o caminho do arquivo CSV para a função load\_to\_csv() e adicione as instruções necessárias lá.

▶ [Clique aqui para dica](#)  
▶ [Clique aqui para solução](#)

2. Você precisa salvar o dataframe transformado como uma tabela no banco de dados. Isso precisa ser implementado na função load\_to\_db(), que aceita o dataframe df, o objeto de conexão com o banco de dados SQL conn e a variável de nome da tabela table\_name a ser utilizada.

▶ [Clique aqui para dica](#)  
▶ [Clique aqui para solução](#)

## Tarefa 4: Consultando a tabela do banco de dados

Assumindo que a consulta apropriada foi iniciada e a instrução da consulta foi passada para a função run\_query(), juntamente com o objeto de conexão SQL sql\_connection e a variável do nome da tabela table\_name, esta função deve executar a instrução da consulta na tabela e recuperar a saída como um dataframe filtrado. Este dataframe pode ser simplesmente impresso.

▶ [Clique aqui para dica](#)  
▶ [Clique aqui para solução](#)

## Tarefa 5: Registro de progresso

O registro deve ser feito usando a função log\_progress(). Esta função será chamada várias vezes durante a execução deste código e será solicitada a adicionar uma entrada de log em um arquivo .txt, etl\_project\_log.txt. A entrada deve estar no seguinte formato:

```
'<Carimbo_de_tempo> : <texto_da_mensagem>'
```

Aqui, o texto da mensagem é passado para a função como um argumento. Cada entrada deve estar em uma linha separada.

▶ [Clique aqui para dica](#)  
▶ [Clique aqui para solução](#)

# Chamadas de função

Agora, você deve configurar a sequência de chamadas de função para as tarefas atribuídas. Siga a sequência abaixo.

Tarefa	Mensagem de log na conclusão
Declarar valores conhecidos	Preliminares concluídas. Iniciando o processo ETL.
Chamar função extract()	Extração de dados concluída. Iniciando o processo de transformação.
Chamar função transform()	Transformação de dados concluída. Iniciando o processo de carregamento.
Chamar load_to_csv()	Dados salvos no arquivo CSV.
Iniciar conexão SQLite3	Conexão SQL iniciada.
Chamar load_to_db()	Dados carregados no banco de dados como tabela. Executando a consulta.
Chamar run_query() *	Processo completo.
Fechar conexão SQLite3	-

Nota: A instrução da consulta a ser executada aqui é

```
f"SELECT * from {table_name} WHERE GDP_USD_billions >= 100"
```

► [Clique aqui para solução](#)

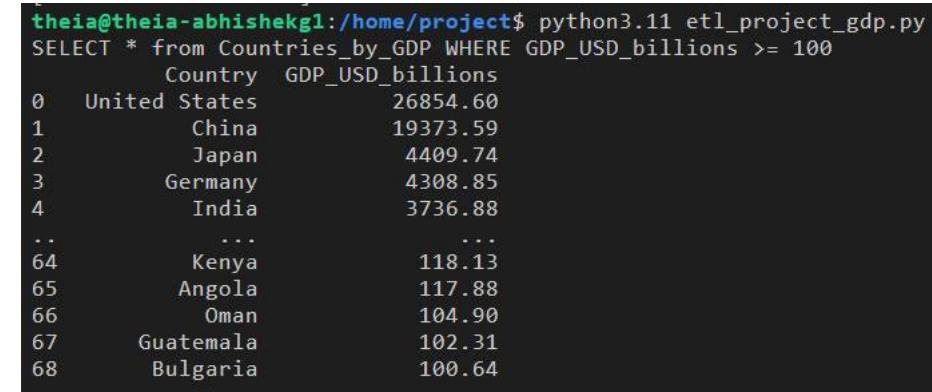
## Execução de Código e Saída Esperada

Uma vez que o código esteja completo, execute-o através do terminal usando o seguinte comando:

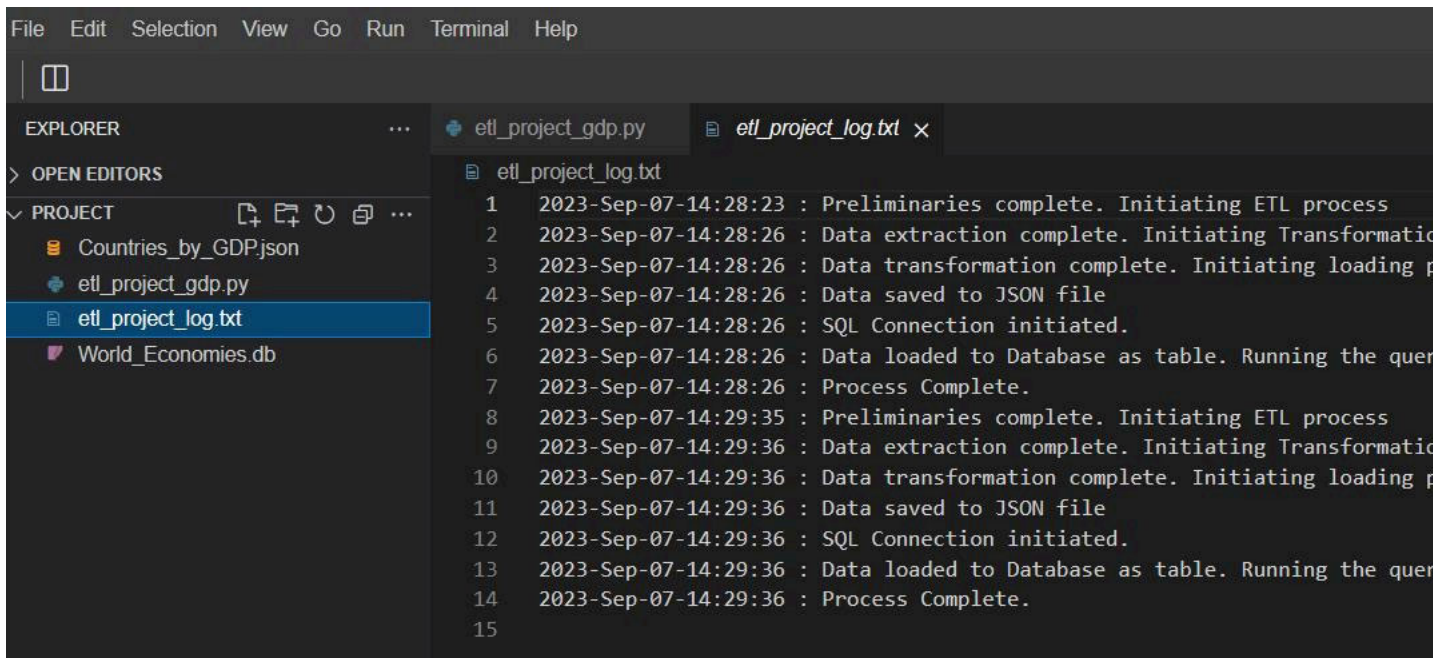
```
python3.11 etl_project_gdp.py
```

Você deve ser capaz de fazer as seguintes observações.

- 1. Saída do terminal



- 2. Arquivos criados e o conteúdo do log



The screenshot shows a VS Code interface. The Explorer panel on the left shows a project structure with files like 'Countries\_by\_GDP.json', 'etl\_project\_gdp.py', 'etl\_project\_log.txt' (selected), and 'World\_Economies.db'. The main editor area shows the content of 'etl\_project\_log.txt', which contains a log of ETL process steps with timestamps. The terminal panel at the bottom shows the same log content.

```
File Edit Selection View Go Run Terminal Help

EXPLORER
OPEN EDITORS
PROJECT
  Countries_by_GDP.json
  etl_project_gdp.py
  etl_project_log.txt
  World_Economies.db

etl_project_log.txt
1 2023-Sep-07-14:28:23 : Preliminaries complete. Initiating ETL process
2 2023-Sep-07-14:28:26 : Data extraction complete. Initiating Transformatio
3 2023-Sep-07-14:28:26 : Data transformation complete. Initiating loading p
4 2023-Sep-07-14:28:26 : Data saved to JSON file
5 2023-Sep-07-14:28:26 : SQL Connection initiated.
6 2023-Sep-07-14:28:26 : Data loaded to Database as table. Running the quer
7 2023-Sep-07-14:28:26 : Process Complete.
8 2023-Sep-07-14:29:35 : Preliminaries complete. Initiating ETL process
9 2023-Sep-07-14:29:36 : Data extraction complete. Initiating Transformatio
10 2023-Sep-07-14:29:36 : Data transformation complete. Initiating loading p
11 2023-Sep-07-14:29:36 : Data saved to JSON file
12 2023-Sep-07-14:29:36 : SQL Connection initiated.
13 2023-Sep-07-14:29:36 : Data loaded to Database as table. Running the quer
14 2023-Sep-07-14:29:36 : Process Complete.
15
```

#### Nota Importante:

Mantendo a consistência da estrutura do laboratório, a página da web acessada é roteada através de um banco de dados de arquivos. Muitas vezes, caso o servidor de arquivos esteja ocupado, os usuários podem encontrar uma execução atrasada e/ou um erro como: `requests.exceptions.ConnectionError: HTTPSConnectionPool(host='web.archive.org', port=443): Max retries exceeded with url.` Nessa situação, tente executar o código novamente. Caso o problema persista, você pode mudar a URL para a versão ao vivo, como: [https://en.wikipedia.org/wiki/List\\_of\\_countries\\_by\\_GDP\\_%28nominal%29](https://en.wikipedia.org/wiki/List_of_countries_by_GDP_%28nominal%29)

## Solução do projeto de prática

Caso você não consiga obter a saída necessária do código ou esteja enfrentando alguns erros, o arquivo final para `etl_project_gdp.py` está compartilhado abaixo. Por favor, note que isso é para sua ajuda, e incentivamos você a tentar resolver os erros por conta própria primeiro.

► `etl_project_gdp.py`