Teste de Analista de Dados

Pulse Client Experts

Alexandre F B Ruas

1. Introdução

Este manual técnico tem como objetivo apresentar, de forma estruturada e detalhada, a arquitetura desenvolvida para o projeto de análise dos microdados do Exame Nacional do Ensino Médio (ENEM) 2020. O documento descreve os componentes essenciais da solução, incluindo os softwares e linguagens de programação utilizadas, o processo de modelagem dos dados, o banco de dados empregado e os fluxos de trabalho adotados.

Além de fornecer uma visão abrangente da infraestrutura técnica, este manual também tem a finalidade de permitir que outros profissionais possam reproduzir integralmente o ambiente, seja para fins de estudo, validação ou expansão da análise. Para isso, são disponibilizadas instruções sobre a configuração dos recursos, a preparação dos dados e a execução dos painéis analíticos.

A abordagem adotada prioriza a transparência metodológica e a reprodutibilidade, contribuindo para o fortalecimento de práticas abertas e colaborativas na análise de dados educacionais.

2. Critérios e Recomendações

2.1. Critérios avaliados

- Docker
- SQL
- Python
- Organização do Código
- Documentação
- ETL
- Modelagem dos dados

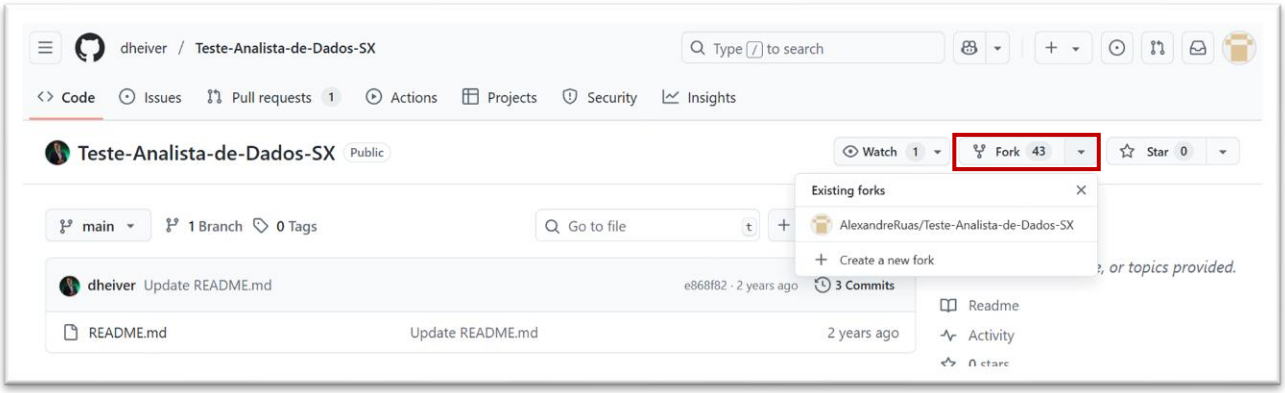
2.2. Recomendações

- PySpark
- Esquema Estrela

3. Repositório

3.1. Acesso ao repositório e criação de fork

O repositório original do projeto no GitHub pode ser acessado através do link: https://github.com/dheiver/Teste-Analista-de-Dados-SX/tree/main.



A partir dele, foi criado um novo fork para desenvolvimento do projeto, disponível em: **AlexandreRuas/Teste-Analista-de-Dados-SX**.

Link para o GitHub na Web:
https://github.com/AlexandreRuas/Teste-Analista-de-Dados-SX

3.2. Dados do projeto

Os dados necessários para o desenvolvimento do projeto estão disponíveis para baixar no seguinte link:

https://download.inep.gov.br/microdados/microdados_enem_2020.zip

Após a extração do arquivo, a base de dados principal pode ser encontrada no diretório: **\microdados_enem_2020\DADOS\MICRODADOS_ENEM_2020.csv**

3.3. Estrutura de dados

A nova estrutura de dados foi organizada no diretório **Teste-Analista-de-Dados-SX**, conforme descrito a seguir:

dados	~~ITENS_PROVA_2020.csv~~ ~~MICRODADOS_ENEM_2020.csv~~
dicionario	Dicionario_Microdados_Enem_2020.ods Dicionario_Microdados_Enem_2020.xlsx
documentos enem	Edital_Enem_2020_Digital.pdf Edital_Enem_2020_Impresso.pdf Leia_Me_Enem_2020.pdf manual_de_redacao_do_enem_2020.pdf matriz_referencia_enem.pdf
documentos tecnicos	manual-tecnico.docx manual-tecnico.pdf
inputs	INPUT_R_ITENS_PROVA_2020.R INPUT_R_MICRODADOS_ENEM_2020.R INPUT_SAS_ITENS_PROVA_2020.sas

	INPUT_SAS_MICRODADOS_ENEM_2020.sas
	INPUT_SPSS_ITENS_PROVA_2020.sps
	INPUT_SPSS_MICRODADOS_ENEM_2020.sps
provas e gabaritos	ENEM_2020_P1_CAD_01_DIA_1_AZUL.pdf
	ENEM_2020_P1_CAD_02_DIA_1_AMARELO.pdf
	ENEM_2020_P1_CAD_03_DIA_1_BRANCO.pdf
	ENEM_2020_P1_CAD_04_DIA_1_ROSA.pdf
	ENEM_2020_P1_CAD_04_DIA_1_ROSA_AMPLIADA.pdf
	ENEM_2020_P1_CAD_04_DIA_1_ROSA_SUPERAMPLIADA.pdf
	ENEM_2020_P1_CAD_09_DIA_1_LARANJA_LEDOR.pdf
	ENEM_2020_P1_CAD_10_DIA_1_VERDE_LIBRAS.pdf
	ENEM_2020_P1_CAD_05_DIA_2_AMARELO.pdf
	ENEM_2020_P1_CAD_06_DIA_2_CINZA.pdf
	ENEM_2020_P1_CAD_07_DIA_2_AZUL.pdf
	ENEM_2020_P1_CAD_08_DIA_2_ROSA.pdf
	ENEM_2020_P1_CAD_08_DIA_2_ROSA_AMPLIADA.pdf
	ENEM_2020_P1_CAD_08_DIA_2_ROSA_SUPERAMPLIADA.pdf
	ENEM_2020_P1_CAD_11_DIA_2_LARANJA_LEDOR.pdf
	ENEM_2020_P1_CAD_12_DIA_2_VERDE_LIBRAS.pdf
	ENEM_2020_P1_GAB_01_DIA_1_AZUL.pdf
	ENEM_2020_P1_GAB_02_DIA_1_AMARELO.pdf
	ENEM_2020_P1_GAB_03_DIA_1_BRANCO.pdf
	ENEM_2020_P1_GAB_04_DIA_1_ROSA.pdf
	ENEM_2020_P1_GAB_04_DIA_1_ROSA_AMPLIADA.pdf
	ENEM_2020_P1_GAB_04_DIA_1_ROSA_SUPERAMPLIADA.pdf
	ENEM_2020_P1_GAB_09_DIA_1_LARANJA_LEDOR.pdf
	ENEM_2020_P1_GAB_10_DIA_1_VERDE_LIBRAS.pdf
	ENEM_2020_P1_GAB_05_DIA_2_AMARELO.pdf
	ENEM_2020_P1_GAB_06_DIA_2_CINZA.pdf
	ENEM_2020_P1_GAB_07_DIA_2_AZUL.pdf
	ENEM_2020_P1_GAB_08_DIA_2_ROSA.pdf
	ENEM_2020_P1_GAB_08_DIA_2_ROSA_AMPLIADA.pdf
	ENEM_2020_P1_GAB_08_DIA_2_ROSA_SUPERAMPLIADA.pdf
	ENEM_2020_P1_GAB_11_DIA_2_LARANJA_LEDOR.pdf
	ENEM_2020_P1_GAB_12_DIA_2_VERDE_LIBRAS.pdf
	ENEM_2020_P2_CAD_01_DIA_1_AZUL.pdf
	ENEM_2020_P2_CAD_02_DIA_1_AMARELO.pdf
	ENEM_2020_P2_CAD_03_DIA_1_BRANCO.pdf
	ENEM_2020_P2_CAD_04_DIA_1_ROSA.pdf
	ENEM_2020_P2_CAD_04_DIA_1_ROSA_AMPLIADA.pdf
	ENEM_2020_P2_CAD_04_DIA_1_ROSA_SUPERAMPLIADA.pdf
	ENEM_2020_P2_CAD_09_DIA_1_LARANJA_LEDOR.pdf
	ENEM_2020_P2_CAD_05_DIA_2_AMARELO.pdf
	ENEM_2020_P2_CAD_06_DIA_2_CINZA.pdf
	ENEM_2020_P2_CAD_07_DIA_2_AZUL.pdf
	ENEM_2020_P2_CAD_08_DIA_2_ROSA.pdf
	ENEM_2020_P2_CAD_08_DIA_2_ROSA_AMPLIADA.pdf
	ENEM_2020_P2_CAD_08_DIA_2_ROSA_SUPERAMPLIADA.pdf
	ENEM_2020_P2_CAD_11_DIA_2_LARANJA_LEDOR.pdf
	ENEM_2020_P2_GAB_01_DIA_1_AZUL.pdf
	ENEM_2020_P2_GAB_02_DIA_1_AMARELO.pdf
	ENEM_2020_P2_GAB_03_DIA_1_BRANCO.pdf
	ENEM_2020_P2_GAB_04_DIA_1_ROSA.pdf
	ENEM_2020_P2_GAB_04_DIA_1_ROSA_AMPLIADA.pdf
	ENEM_2020_P2_GAB_04_DIA_1_ROSA_SUPERAMPLIADA.pdf
	ENEM_2020_P2_GAB_09_DIA_1_LARANJA_LEDOR.pdf
	ENEM_2020_P2_GAB_05_DIA_2_AMARELO.pdf
	ENEM_2020_P2_GAB_06_DIA_2_CINZA.pdf
	ENEM_2020_P2_GAB_07_DIA_2_AZUL.pdf
	ENEM_2020_P2_GAB_08_DIA_2_ROSA.pdf
	ENEM_2020_P2_GAB_08_DIA_2_ROSA_AMPLIADA.pdf
	ENEM_2020_P2_GAB_08_DIA_2_ROSA_SUPERAMPLIADA.pdf
	ENEM_2020_P2_GAB_11_DIA_2_LARANJA_LEDOR.pdf
	ENEM_2020_DIGITAL_CAD_01_DIA_1_AZUL_ESPANHOL.pdf
	ENEM_2020_DIGITAL_CAD_01_DIA_1_AZUL_INGLES.pdf
	ENEM_2020_DIGITAL_CAD_02_DIA_1_AMARELO_ESPANHOL.pdf

	ENEM_2020_DIGITAL_CAD_02_DIA_1_AMARELO_INGLES.pdf ENEM_2020_DIGITAL_CAD_03_DIA_1_BRANCO_ESPANHOL.pdf ENEM_2020_DIGITAL_CAD_03_DIA_1_BRANCO_INGLES.pdf ENEM_2020_DIGITAL_CAD_04_DIA_1_ROSA_ESPANHOL.pdf ENEM_2020_DIGITAL_CAD_04_DIA_1_ROSA_INGLES.pdf ENEM_2020_DIGITAL_CAD_05_DIA_2_AMARELO.pdf ENEM_2020_DIGITAL_CAD_06_DIA_2_CINZA.pdf ENEM_2020_DIGITAL_CAD_07_DIA_2_AZUL.pdf ENEM_2020_DIGITAL_CAD_08_DIA_2_ROSA.pdf ENEM_2020_DIGITAL_GAB_01_DIA_1_AZUL_ESPANHOL.pdf ENEM_2020_DIGITAL_GAB_01_DIA_1_AZUL_INGLES.pdf ENEM_2020_DIGITAL_GAB_02_DIA_1_AMARELO_ESPANHOL.pdf ENEM_2020_DIGITAL_GAB_02_DIA_1_AMARELO_INGLES.pdf ENEM_2020_DIGITAL_GAB_03_DIA_1_BRANCO_ESPANHOL.pdf ENEM_2020_DIGITAL_GAB_03_DIA_1_BRANCO_INGLES.pdf ENEM_2020_DIGITAL_GAB_04_DIA_1_ROSA_ESPANHOL.pdf ENEM_2020_DIGITAL_GAB_04_DIA_1_ROSA_INGLES.pdf ENEM_2020_DIGITAL_GAB_05_DIA_2_AMARELO.pdf ENEM_2020_DIGITAL_GAB_06_DIA_2_CINZA.pdf ENEM_2020_DIGITAL_GAB_07_DIA_2_AZUL.pdf ENEM_2020_DIGITAL_GAB_08_DIA_2_ROSA.pdf	
python	ETL-PySpark.ipynb	
sql	insights	dist_notas.sql media_geral.sql media_por_disciplina.sql media_por_estado.sql media_por_estado_pivot.sql media_por_etnia.sql media_por_genero.sql media_por_inscrito.sql media_por_municio_escola.sql media_por_tipo_escola.sql media_por_tipo_escola_pivot.sql microdados_rj.sql microdados_sp.sql percentual_ausentes.sql respostas_desafio.sql total_inscritos.sql
	tabelas dimensionais	dim_ano_conclusao.sql dim_area_conhecimento.sql dim_bens_e_infraestrutura.sql dim_cor_prova.sql dim_cor_raca.sql dim_dep_adm_escola.sql dim_estado_civil.sql dim_faixa_etaria.sql dim_funcionamento_escola.sql dim_grau_escolaridade.sql dim_grupo_ocupacao.sql dim_lingua_estrangeira.sql dim_localizacao_escola.sql dim_motivo_abandono.sql dim_municipio.sql dim_nacionalidade.sql dim_presenca_prova_objetiva.sql dim_renda_mensal.sql dim_servico_domiciliar.sql dim_sexo.sql dim_sim_ou_nao.sql dim_situacao_conclusao.sql dim_status_redacao.sql dim_tipo_ensino.sql dim_tipo_escola.sql dim_tipo_prova_objetiva.sql dim_uf.sql

	tabelas fato	fato_escola.sql fato_itens_prova.sql fato_local_prova.sql fato_participante.sql fato_prova_objetiva.sql fato_questionario.sql fato_redacao.sql
	views	vw_escola.sql vw_itens_prova.sql vw_local_prova.sql vw_microdados.sql vw_participante.sql vw_prova_objetiva.sql vw_questionario.sql vw_redacao.sql
	esquema.mwb	
	esquema.pdf	
tableau	dados	viz_dist_notas.csv viz_dist_notas_ch.csv viz_dist_notas_cn.csv viz_dist_notas_lc.csv viz_dist_notas_mt.csv viz_dist_notas_por_inscrito.csv viz_dist_notas_rd.csv viz_media_por_estado.csv viz_media_por_estado_pivot.csv viz_media_por_tipo_escola.csv viz_media_por_tipo_escola_pivot.csv viz_microdados_rj.csv viz_microdados_sp.csv
	enem_2020.twb	
README.md		

A seguir, uma breve descrição dos arquivos que compõem a estrutura de dados do projeto:

3.3.1. Dados

O arquivo **MICRODADOS_ENEM_2020.csv** contém informações gerais sobre a realização das provas, a caracterização do participante e da escola que ele declarou ter frequentado, e as notas das provas objetivas e da redação e um questionário socioeconômico.

O arquivo **ITENS_PROVA_2020.csv** contém informações gerais sobre os itens das provas.

O diretório **dados** não foi incluído no GitHub devido ao seu tamanho elevado, sobretudo por conta do arquivo **MICRODADOS_ENEM_2020.csv**, que possui aproximadamente 2GB.

3.3.2. Dicionário

Os arquivos **Dicionário_Microdados_Enem_2020.ods** e **Dicionário_Microdados_Enem_2020.xlsx** possuem o mesmo conteúdo: informações gerais sobre as variáveis contidas nas bases e sobre as perguntas e alternativas do questionário aplicado.

3.3.3. Documentos ENEM

Conteúdo do diretório **documentos enem**:

- **Leia-me_Enem_2020.pdf**: breve descrição do Enem, bem como das informações sobre as bases e os arquivos disponibilizados nos Microdados.
- **Matriz_referencia_enem.pdf**: apresentação da Matriz de Referência, que compreende os eixos cognitivos, as competências e as habilidades avaliadas em cada área de conhecimento do Ensino Médio.
- **Edital_Enem_2020_Impresso.pdf**: edital de publicação do ENEM 2020, versão tradicional do Enem, com aplicação em papel, em locais convencionais, com uso de cadernos físicos.
- **Edital_Enem_2020_Digital.pdf**: edital de publicação do ENEM 2020, versão digital, aplicada pela primeira vez como projeto piloto (2020), aplicada em laboratórios de informática, com prova realizada em computador, com mesmo conteúdo e estrutura de prova impressa.
- **Manual_de_redacao_do_enem_2020.pdf**: expõem a metodologia de avaliação da redação, bem como o que se espera do participante em cada uma das competências avaliadas.

3.3.4. Documentos Técnicos

O diretório **documento tecnicos** reúne os documentos técnicos relacionados ao projeto, com o propósito de facilitar a compreensão das etapas envolvidas, das configurações adotadas, dos softwares utilizados e de outros aspectos relevantes para a reprodução e o entendimento da solução desenvolvida.

3.3.5. Inputs

No diretório **inputs** são encontrados arquivos com extensões **.r**, **.sas** e **.sps**. Estes arquivos são utilizados em ambientes de análise estatística e ciência de dados. Eles armazenam scripts ou conjuntos de dados que permitem a execução de procedimentos analíticos, manipulação de informações, aplicação de modelos estatísticos e geração de relatórios. Cada formato está associado a uma plataforma específica e desempenha papel fundamental no desenvolvimento e na documentação de projetos quantitativos.

3.3.6. Provas e Gabaritos

No diretório **provas e gabaritos** estão disponíveis os cadernos de prova (impressos e digitais) do ENEM 2020, juntamente com seus respectivos gabaritos, referentes à primeira e à segunda aplicações do exame.

3.3.7. Python

Arquivo **ETL-PySpark.ipynb**, contendo o código do processo ETL desenvolvido em PySpark para leitura e carregamento dos arquivos **MICRODADOS_ENEM_2020.csv** e **ITENS_PROVA_2020.csv** no banco de dados **enem**, no container **mysql-enem** do Docker.

3.3.8. SQL

O diretório **sql** reúne todos os scripts SQL empregados na criação das tabelas fato e dimensionais, além das views e demais tabelas utilizadas ao longo do desenvolvimento do projeto. Também estão incluídos dois arquivos complementares:

- **esquema.pdf**: apresenta o diagrama entidade-relacionamento (ER) do banco de dados Enem.
- **esquema.mwb**: contém o modelo do banco de dados elaborado no MySQL Workbench.

3.3.9. Tableau

O diretório **tableau** armazena o arquivo principal **enem_2020.twb**, com as visualizações do Tableau. Além disso, contém os arquivos **.csv** exportados do MySQL, que representam os dados das tabelas utilizadas na elaboração dos dashboards e análises visuais.

3.3.10. README

Arquivo **README.md**, localizado na raiz do diretório **\Teste-Analista-de-Dados-SX**, contendo instruções para o desenvolvimento do projeto, foi editado. Todas as respostas requeridas, após a conclusão do projeto, foram adicionadas neste arquivo.

4. Softwares

Nesta seção, são apresentados os softwares empregados no desenvolvimento do projeto.

O objetivo é garantir que qualquer usuário possa reproduzir o ambiente de trabalho utilizado, facilitando a execução dos scripts, a análise dos dados e a replicação dos resultados.

4.1. Git Desktop

Git é um sistema de controle de versão distribuído, amplamente utilizado para gerenciar o histórico de alterações em projetos de software. Ele permite que múltiplos desenvolvedores colaborem simultaneamente, mantendo rastreabilidade, segurança e organização no código-fonte.

No Windows, o Git pode ser instalado com suporte a comandos Unix por meio do Git Bash, um terminal que acompanha o instalador oficial e oferece uma experiência semelhante à de sistemas Linux.

4.1.1. Onde baixar

O Git para Windows está disponível gratuitamente no site oficial:

https://git-scm.com/downloads

A instalação inclui:

- Git Bash (terminal com comandos Unix)
- Git GUI (interface gráfica opcional)
- Integração com o terminal padrão do Windows (PowerShell ou CMD)

4.1.2. Por que utilizar o Git Desktop

A adoção do Git neste projeto tem como objetivos:

- Visualização e controle de alterações.
- Commits, push/pull e clonagem do repositório.
- Manter histórico completo de alterações.
- Integrar com plataformas como GitHub para hospedagem e controle remoto.
- Aproveitar a integração nativa com o Visual Studio Code (VS Code).
- Atualizar o GitHub na Web com o conteúdo da pasta local: **C:\tmp\Teste-Analista-de-Dados-SX**.

4.2. Docker

Docker Desktop é uma aplicação que permite criar, gerenciar e executar containers de forma simples em ambientes Windows, macOS e Linux. Ele fornece uma interface gráfica

intuitiva e integra ferramentas de linha de comando para facilitar o desenvolvimento de aplicações isoladas e portáveis.

No contexto deste projeto, o Docker é utilizado para criar um ambiente de banco de dados MySQL encapsulado em um container, garantindo consistência, reprodutibilidade e facilidade de configuração.

4.2.1. Onde baixar

O Docker Desktop pode ser baixado gratuitamente no site oficial:

https://www.docker.com/products/docker-desktop/

Após baixar o instalador, siga as instruções de instalação específicas para seu sistema operacional. É necessário habilitar a virtualização no BIOS e, em alguns casos, instalar o WSL2 no Windows. Nesse projeto, após concluir a instalação do Docker no Windows, foi necessário atualizar o subsistema **WSL (Windows Subsystem for Linux)** para garantir o funcionamento adequado do software.

Essa atualização foi realizada executando o seguinte comando em uma janela do Prompt de Comando (CMD) com privilégios administrativos:

```
wsl -- update
```

4.2.2. Criação do container

Neste projeto, foi criado um container que tem como objetivo:

- Isolar o banco de dados do ambiente local, evitando conflitos com outras instâncias de MySQL instaladas.
- Facilitar a replicação do ambiente por outros usuários ou membros da equipe.
- Automatizar a carga e manipulação dos dados do ENEM, utilizando scripts SQL e arquivos CSV.
- Controlar versões e configurações específicas do MySQL sem afetar o sistema principal.

Comando executado para criação do container:

```
docker run --name mysql-enem -e MYSQL_ROOT_PASSWORD=senha123 -e
MYSQL_DATABASE=enem2020 -p 3306:3306 -d mysql:8.0
```

Esse comando cria um container MySQL com:

- Nome: **mysql-enem**
- Senha do usuário root: **root**
- Banco de dados inicial: **enem**
- Porta exposta: **3306**

4.2.3. Arquivo de configuração (opcional)

O arquivo **.wslconfig** permite controlar o uso de recursos pelo WSL2, como memória RAM, número de núcleos de CPU e espaço de swap. Essa configuração é especialmente útil para evitar que o computador fique lento ou trave, principalmente quando o Docker Desktop está em uso, já que ele opera sobre o WSL2 e pode consumir uma quantidade significativa de memória.

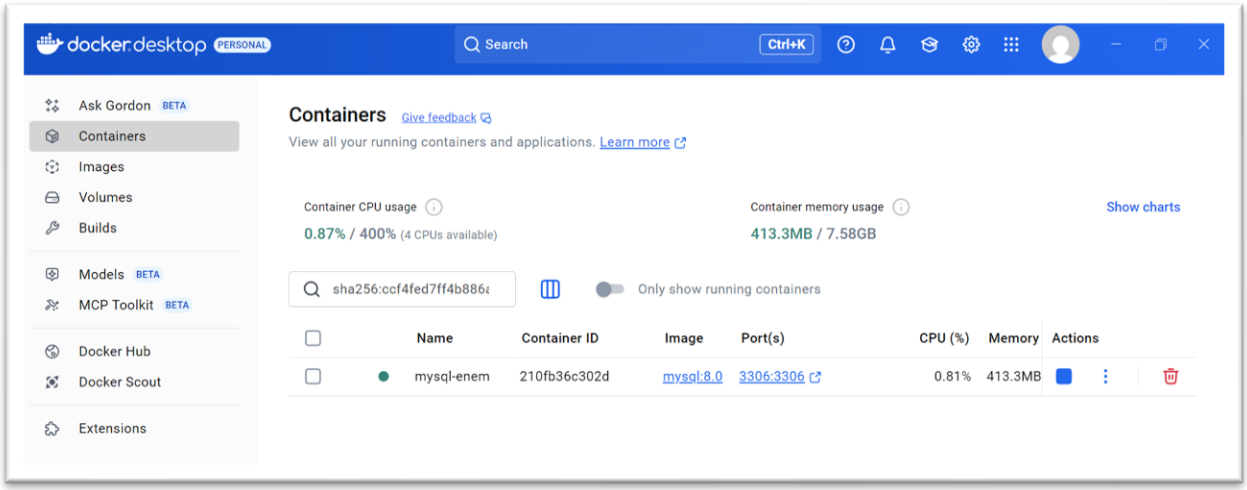
O arquivo de configuração pode ser criado em **C:\Users\<>**, com o seguinte conteúdo:

[wsl2]
memory=8GB
processors=4

Para que o WSL2 e o Docker Desktop respeitem essas configurações, você precisa reiniciar o ambiente WSL, fechar o Docker Desktop e executar o comando:

wsl – shutdwon

Esse comando encerra **todas as distribuições WSL2**, incluindo **docker-desktop** e **docker-desktop-data**. Após isso, ao abrir o Docker Desktop ou qualquer terminal WSL novamente, as configurações serão aplicadas.



4.3. Anaconda

Anaconda é uma distribuição Python que facilita o gerenciamento de ambientes e pacotes. Ele inclui:

- Jupyter Notebook (interface interativa para código Python)
- Conda (gerenciador de pacotes e ambientes)
- Suporte a bibliotecas como NumPy, Pandas, Matplotlib, PySpark etc.

4.3.1. Onde baixar

O Anaconda pode ser baixado gratuitamente no site oficial:
https://www.anaconda.com/download

4.3.2. Usando Jupyter com PySpark no Anaconda

Instalar PySpark no Anaconda:

```
conda create -n pyspark_env python=3.10
conda activate pyspark_env
conda install openjdk
conda install pyspark
conda install -c conda-forge findspark
```

Para criar o kernel:

```
conda install ipykernel
python -m ipykernel install --user --name=pyspark_env --display-name "Python (PySpark)"
```

Isso vai registrar seu ambiente como uma opção de kernel, com nome Python (PySpark), no Jupyter.

Para iniciar o Jupyter Notebook via terminal:

```
jupyter notebook
```

No notebook, escrever o script PySpark para carregar o conteúdo do arquivo **MICRODADOS_ENEM_2020.csv** no container **mysql-enem** no Docker.

4.3.3. Script PySpark

O script (**ETL-PySpark.ipynb**) pode ser encontrado em:
\Teste-Analista-de-Dados-SX\python

```python
import os

# Configuração das variáveis de ambiente
os.environ['SPARK_HOME'] = 'C:\\tmp\\spark-3.5.6-bin-hadoop3'
```

Define a variável de ambiente **SPARK_HOME**, apontando para o diretório onde o Spark está instalado. Isso é necessário para que o PySpark saiba onde encontrar os binários do Spark.

No Windows, é comum que o Spark exija a variável de ambiente JAVA_HOME corretamente configurada, especialmente quando o Java está instalado em um diretório não padrão ou não é automaticamente reconhecido pelo sistema. Uma configuração incorreta pode impedir o Spark de inicializar ou causar falhas silenciosas.

```python
from pyspark.sql import SparkSession

# Inicialização da sessão spark
spark = SparkSession.builder.appName("ENEM ETL").getOrCreate()
```

Cria uma SparkSession, que é o ponto de entrada para usar o Spark com DataFrames.

O nome da aplicação é **ENEM ETL**, útil para monitoramento em interfaces como Spark UI.

```
# Localização dos arquivos
path_microdados = "C:/tmp/Teste-Analista-de-Dados-SX/dados/MICRODADOS_ENEM_2020.csv"
path_itens_prova = "C:/tmp/Teste-Analista-de-Dados-SX/dados/ITENS_PROVA_2020.csv"

# Leitura dos arquivos
df_microdados = spark.read.csv(path_microdados, sep=";", header=True, inferSchema=True, encoding="ISO-8859-1")
df_itens_prova = spark.read.csv(path_itens_prova, sep=";", header=True, inferSchema=True, encoding="ISO-8859-1")
```

Define os caminhos dos arquivos CSV com os dados do ENEM 2020.

Lê os arquivos CSV como DataFrames Spark.

- **sep=";"**: separador usado nos arquivos.
- **header=True**: indica que a primeira linha contém os nomes das colunas.
- **inferSchema=True**: tenta inferir automaticamente os tipos de dados.
- **encoding="ISO-8859-1"**: usado para lidar com acentuação e caracteres especiais do português.

```
from sqlalchemy import create_engine

# Conexão com MySQL no Docker
engine = create_engine("mysql+mysqlconnector://root:root@localhost:3306/enem")
```

Cria um engine **SQLAlchemy** para se conectar ao MySQL.

```
# Configurações JDBC para MySQL no Docker
jdbc_url = "jdbc:mysql://localhost:3306/enem"
jdbc_properties = {
    "user": "root",
    "password": "root",
    "driver": "com.mysql.cj.jdbc.Driver"
}
```

Define a URL JDBC e as propriedades de conexão para o Spark se comunicar com o MySQL.

```
%%time
# Grava dataframe spark diretamente no MySQL usando JDBC
df_itens_prova.write \
    .option("batchsize", 5000) \
    .option("truncate", "true") \
    .jdbc(url=jdbc_url, table="itens_prova_2020", mode="overwrite", properties=jdbc_properties)
```

Escreve o DataFrame **df_itens_prova** na tabela **itens_prova_2020** do MySQL.

- **batchsize=5000**: envia os dados em lotes de 5000 linhas.
- **truncate=true**: limpa a tabela antes de inserir os dados (válido apenas com overwrite).
- **mode="overwrite"**: substitui os dados existentes na tabela.

```
%%time
# Grava dataframe spark diretamente no MySQL usando JDBC
df_microdados.write \
    .option("batchsize", 5000) \
    .option("truncate", "true") \
    .jdbc(url=jdbc_url, table="microdados_enem_2020", mode="overwrite", properties=jdbc_properties)
```

Mesma lógica, mas agora para o DataFrame **df_microdados**, gravando na tabela **microdados_enem_2020**.

4.4. Java JDK 17

O Apache Spark é desenvolvido em Scala, uma linguagem que é executada sobre a Java Virtual Machine (JVM). Mesmo ao utilizar o PySpark, que oferece uma interface em Python, o código Python atua apenas como uma camada de controle, pois é o núcleo do Spark que realiza o processamento, e ele depende diretamente da JVM para funcionar.

Por isso, a instalação do Java JDK é indispensável. Neste projeto, foi utilizado o JDK 17, uma versão estável com suporte de longo prazo, garantindo compatibilidade e desempenho adequado com o Spark 3.5.x.

4.4.1. Onde baixar

O Java pode ser baixado no site oficial da Oracle:
https://www.oracle.com/java/technologies/javase-downloads.html

4.5. Apache Spark 3.5.6 com Hadoop 3

O Spark precisa de um runtime engine para processar dados distribuídos. Ele pode usar o Hadoop para isso. Mesmo que o sistema de arquivos do Hadoop (HDFS) não esteja sendo utilizado, o Spark usa bibliotecas do Hadoop para leitura de arquivos, compressão, etc.

Para este projeto, optou-se pela versão **Spark 3.5.6**, uma versão estável, compatível com **Hadoop 3**. O Hadoop 3 traz melhorias de desempenho, suporte a novos formatos e melhor integração com sistemas modernos.

Durante o desenvolvimento do script em PySpark, o Spark foi executado em **modo local** no ambiente Windows, sem utilização de cluster. Essa abordagem simplifica testes e validações iniciais, permitindo rodar o código diretamente na máquina do desenvolvedor.

4.5.1. Onde baixar

https://spark.apache.org/releases/spark-release-3-5-0.html

4.6. MySQL

Pode-se utilizar diferentes ferramentas para modelar, administrar e criar os objetos de um banco de dados MySQL, dependendo das preferências e necessidades do desenvolvedor. Uma opção bastante popular é o **VS Code**, que, com a instalação da extensão apropriada

para MySQL, permite executar consultas SQL, visualizar estruturas de tabelas e interagir com o banco diretamente do editor.

Outra alternativa robusta é o **MySQL Community**, que inclui o **MySQL Workbench**, uma ferramenta gráfica voltada especificamente para o MySQL. O Workbench oferece recursos avançados como modelagem de dados (diagrama Entidade-Relacionamento — ER), administração de usuários, monitoramento de desempenho, backup/restauração e execução de scripts SQL. O Workbench é especialmente útil em cenários que exigem maior controle e visualização detalhada da estrutura do banco.

Assim, tanto o VS Code com extensões quanto o MySQL Workbench são ferramentas eficazes e complementares para o desenvolvimento e administração de bancos de dados MySQL.

4.6.1. Onde baixar

Para baixar o VS Code:
https://code.visualstudio.com/download
Para baixar o MySQL Community:
https://dev.mysql.com/downloads/installer/

4.6.2. Conexão com o Docker

Utilize a configuração:

- **Host:** 127.0.0.1 (localhost). Conecta via TCP.
- **Porta:** 3306. Porta mapeada do Docker.
- **Usuário**: root. Usuário definido no container.
- **Senha**: root. Senha definida no container.
- **Banco de Dados**: enem. Nome do banco, a ser conectado, no Docker.

4.7. Tableau Public

Para a presente solução, foi adotado o Tableau Public como ferramenta de visualização de dados. A escolha se deu por sua acessibilidade, recursos robustos de criação de dashboards interativos e a possibilidade de publicação online gratuita, o que facilita o compartilhamento dos resultados.

Como a versão utilizada não permite conexão direta com o banco de dados enem (MySQL), optou-se por um fluxo de trabalho baseado na exportação dos dados em formato CSV. Esses arquivos foram preparados previamente, contendo os dados estruturados e limpos, e então importados para o Tableau Public para construção das visualizações.

Essa abordagem, embora mais manual, garantiu flexibilidade e controle sobre os dados utilizados, permitindo:

- Curadoria dos dados antes da visualização, assegurando qualidade e relevância.
- Criação de dashboards dinâmicos, para facilitar a interpretação dos dados por usuários não técnicos.
- Publicação online, tornando os insights acessíveis por meio de links públicos.

Apesar das limitações da versão gratuita, como a impossibilidade de manter os dados privados ou conectar-se a fontes em tempo real, o Tableau Public se mostrou eficaz para os objetivos do projeto, oferecendo uma solução visual poderosa.

4.7.1. Painéis

Foram desenvolvidos doze painéis:

- Painel 1: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel1?publish=yes
- Painel 2: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel2?publish=yes
- Painel 3: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel3?publish=yes
- Painel 4: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel4?publish=yes
- Painel 5: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel5?publish=yes
- Painel 6: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel6?publish=yes
- Painel 7: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel7?publish=yes
- Painel 8: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel8?publish=yes
- Painel 9: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel9?publish=yes
- Painel 10: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel10?publish=yes
- Painel 11: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel11?publish=yes
- Painel 12: https://public.tableau.com/app/profile/alexandre.ruas/viz/enem_2020/Painel12?publish=yes

5. Base de Dados

No projeto, o banco de dados **enem** (MySQL) é executado a partir de um container Docker.

5.1.1. Esquema

Foi adotado o esquema estrela para modelagem dos dados. O diagrama ER correspondente está disponível no arquivo **esquema.pdf**, localizado em **\Teste-Analista-de-Dados-SX\sql\ o**u arquivo **esquema.mwb**, modelo para o MySQL Workbench.

5.1.2. Tabelas fato

No contexto da modelagem dimensional em esquema estrela, as tabelas fato foram estruturadas para armazenar os dados quantitativos relacionados aos participantes do ENEM 2020, incluindo informações sobre as escolas declaradas, locais de prova, presença nas aplicações, itens das provas, notas obtidas e respostas ao questionário socioeconômico.

Os scripts utilizados para criação e relacionamento das tabelas fato podem ser encontrados em **\ Teste-Analista-de-Dados-SX\sql\tabelas fato**.

5.1.3. Tabelas dimensionais

As tabelas dimensionais foram estruturadas para armazenar informações descritivas que contextualizam os dados presentes nas tabelas fato. Essas dimensões incluem atributos como faixa etária, nacionalidade, gênero, cor, raça e outros dados relacionados aos participantes, além de informações sobre os municípios e estados onde as escolas estão localizadas, os locais de aplicação das provas e demais elementos que permitem a segmentação e análise detalhada dos dados do ENEM 2020.

Os scripts utilizados para criação e relacionamento das tabelas dimensionais podem ser encontrados em **\Teste-Analista-de-Dados-SX\sql\tabelas dimensionais**.

5.1.4. Tabelas viz

As tabelas com o prefixo **viz_** foram criadas a partir de dados previamente ajustados e filtrados, com o objetivo de facilitar sua exportação para arquivos **.csv**. Esses arquivos foram utilizados na construção das visualizações no **Tableau Public**, uma vez que essa versão da ferramenta não oferece suporte à conexão direta com bancos de dados.

5.1.5. Outras Tabelas

As tabelas **microdados_enem_2020** e **itens_prova_2020** contém os dados RAW dos arquivos **MICRODADOS_ENEM_2020.csv** e **ITENS_PROVA_2020.csv**, respectivamente.

5.1.6. Views

As views foram desenvolvidas com o objetivo de facilitar o acesso, a consulta e a análise dos dados consolidados a partir das tabelas fato e dimensionais. Elas representam estruturas lógicas que agregam, filtram e organizam informações relevantes para diferentes perspectivas analíticas, como desempenho dos participantes, distribuição geográfica, perfil socioeconômico e estatísticas por escola ou local de prova.
Essas visões permitem abstrair a complexidade dos relacionamentos entre tabelas e oferecem uma camada de consulta otimizada para geração de relatórios e dashboards.

Os scripts utilizados para criação das views podem ser encontrados em **\Teste-Analista-de-Dados-SX\sql\views**.

ANEXO I

Responder às seguintes perguntas:

Observação:

Para responder às perguntas 1, 2, 3, 6,7 e 8 foram consideradas as notas de Ciências da Natureza, Ciências Humanas, Linguagens e Códigos, Matemática e Redação.

1. Qual a escola com a maior média de notas?

Observação:
Infelizmente, os microdados do ENEM 2020 não incluem o nome ou o código das escolas. Por isso, em vez de calcular a maior média por escola, optei por calcular a maior média das escolas localizadas em um
determinado município, utilizando o código de município como referência. Isso representa um total de 5.534 escolas,
considerando todas as redes (municipais, estaduais, federais e privadas), independentemente de estarem situadas em
áreas urbanas ou rurais, e se estão em funcionamento ou não.

Resposta:
Entre todos os municípios analisados, Santana do Manhuaçu, em Minas Gerais (MG), obteve a maior média de notas: 706,56.

1. Qual o aluno com a maior média de notas e o valor dessa média?

Resposta:
O participante com a maior média de notas é o inscrito de número 200005996961, que obteve uma média final de 858,59 pontos.

1. Qual a média geral?

Resposta:
A média geral de notas é 526,58.

1. Qual o % de Ausentes?

Resposta:
O percentual de inscritos ausentes é: 52,15%. Considerando os inscritos que faltaram as provas de
Ciências da Natureza (TP_PRESENCA_CN), Ciências Humanas (TP_PRESENCA_CH), Linguagens e Códigos (TP_PRESENCA_LC) e Matemática (TP_PRESENCA_MT).

1. Qual o número total de Inscritos?

Resposta:
O número total de inscritos é: 5.783.109.

1. Qual a média por disciplina?

Resposta:
Média de Ciências da Natureza (CN): 490,41
Média de Ciências Humanas (CH): 511,15
Média de Linguagens e Códigos (LC): 523,80
Média de Matemática (MT): 520,58
Média de Redação: 573,41

1. Qual a média por Sexo?

Resposta:
Média das notas dos participantes que se identificaram com o gênero feminino: 521,24.
Média das notas dos participantes que se identificaram com o gênero masculino: 534,73.

1. Qual a média por Etnia?

2. **Resposta:**
   Média das notas dos participantes que se declararam como brancos: 556,53.
   Média das notas dos participantes que se declararam como pretos: 500,87.
   Média das notas dos participantes que se declararam como pardos: 508,66
   Média das notas dos participantes que se declararam como amarelos: 524,94.
   Média das notas dos participantes que se declararam como indígenas: 472,84.