**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS  
NÚCLEO DE EDUCAÇÃO A DISTÂNCIA**

**Pós-graduação *Lato Sensu* em Engenharia de Dados**

**Paulo Henrique Santos de Alcantara**

**PROJEÇÃO DA POPULAÇÃO BRASILEIRA**

Belo Horizonte

2023

**Paulo Henrique Santos de Alcantara**

**PROJEÇÃO DA POPULAÇÃO BRASILEIRA**

Trabalho de Conclusão de Curso apresentado ao Curso de Especialização em Engenharia de Dados como requisito parcial à obtenção do título de especialista.

Belo Horizonte

2023**SUMÁRIO**

[**1. Introdução 4**](#_heading=h.30j0zll)

[**1.1. Contextualização 4**](#_heading=h.26in1rg)

[**1.2. O problema proposto 4**](#_heading=h.3znysh7)

[**1.3. Objetivos**](#_heading=h.2et92p0) **4**

[**3. Processamento/Tratamento de Dados 6**](#_heading=h.tyjcwt)

[**4. Análise e Exploração dos Dados**](#_heading=h.3dy6vkm) **8**

**5**[**. Interpretação dos Resultados 9**](#_heading=h.lnxbz9)

**6**[**. Apresentação dos Resultados 10**](#_heading=h.35nkun2)

**7**[**. Links 11**](#_heading=h.4d34og8)

[**REFERÊNCIAS 12**](#_heading=h.17dp8vu)

[**APÊNDICE 13**](#_heading=h.3rdcrjn)

**LISTA DE FIGURAS**

**Figura 1.** Exemplo de montagem de um Query Builder para gerar uma URL. **Fonte.** IBGE

https://servicodados.ibge.gov.br/api/docs/agregados?versao=3#api-bq **5**

**Figura 2.** Fluxograma base para ingestão dos dados. **Fonte.** Lucidchart - Autoria própria **8**

**Figura 3.** Código em python responsável pela extração dos dados da API. **Fonte.** VS Code - Autoria própria **10**

**Figura 4.** Código em python responsável pelo carregamento dos dados da API no Postgres

**Fonte.** VS Code - Autoria própria 20

**LISTA DE TABELAS**

**Tabela 1.** Evasão referente a curso e modalidade. **Fonte:** Autoria Própria **26**

**LISTA DE ABREVIATURAS E SIGLAS**

1. IBGE - Instituto Brasileiro de Geografia e Estatística
2. API - Application Programming Interface / Interface de programação de aplicações
3. URL - Uniform Resource Locator / Localizador Uniforme de Recursos
4. JSON - JavaScript Object Notation
5. SQL - Structured Query Language / Linguagem de Consulta Estrutura
6. ETL - Extract Transform Load

**1. Introdução**

**1.1. Contextualização**

A projeção da população brasileira é uma estimativa do tamanho e da composição da população do Brasil em determinado período futuro, com base em dados demográficos, como taxas de natalidade, mortalidade e outros fatores relevantes. Essas projeções são importantes para o planejamento governamental, a formulação de políticas públicas, o desenvolvimento econômico e a alocação de recursos em diversas áreas, como saúde, educação e previdência social.

Os dados para análises do tema geralmente são realizados pelo IBGE, e são frequentemente revisadas e atualizadas para refletir mudanças nas tendências demográficas e nas políticas governamentais. Essas projeções são fundamentais para o planejamento de políticas públicas em áreas como saúde, educação, habitação, previdência social e infraestrutura, garantindo que o país esteja preparado para enfrentar os desafios decorrentes das mudanças na composição e no tamanho de sua população.

**1.2. O problema proposto**

Atualmente os dados relacionados a projeção populacional brasileira são realizados oficialmente pelo IBGE, dessa forma os dados estão armazenados no site do instituto, a maneira padrão para visualizá-los é visitar o site e baixar o arquivo do tema desejado; porém existe a possibilidade de fazer isso de uma forma mais ágil e essa possibilidade será explorada neste trabalho.

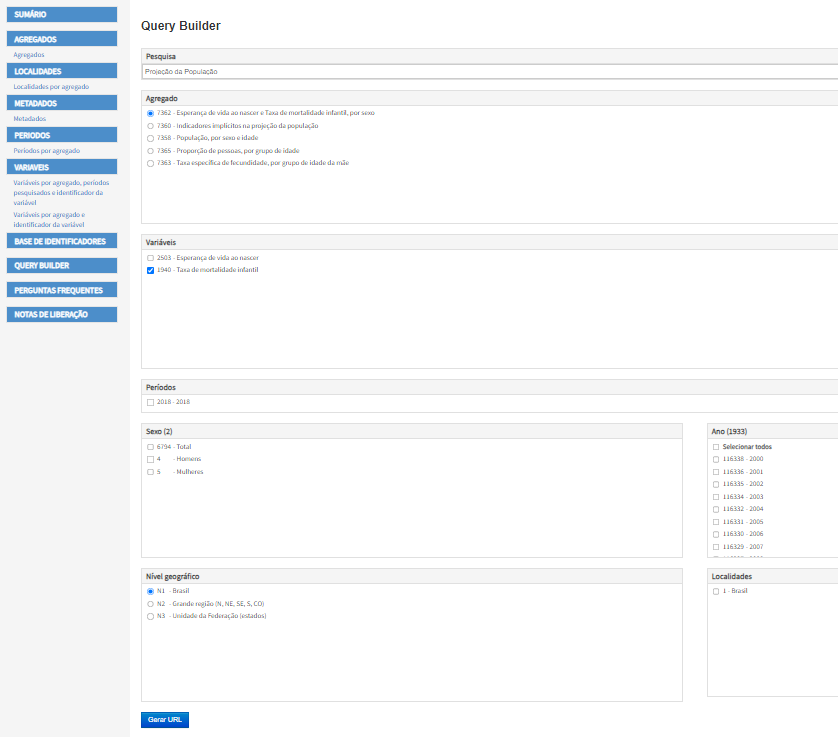
As perguntas que este tópico tem como objetivo responder é: O que é e onde eu consigo capturar esses dados de forma mais ágil? Por que utilizar este método? Quando e como usá-lo? As ferramentas e lógicas implementadas aqui ajudaram a responder a todos os questionamentos citados acima.

**1.3. Objetivos**

Através da engenharia de dados, o objetivo é facilitar a maneira de como melhorar o fluxo de leitura e a compreensão desses dados de projeção populacional, como por exemplo separar assuntos como a taxas de natalidade e mortalidade, óbitos, envelhecimentos da população entre outros temas relacionados em tabelas geradas por meio de APIs e após isso gerar orquestrações e visualizações sistemáticas desses dados, para colaborar na tomada de decisões e ajudar com fatores socioeconômicos que estão ligados ao tópico.

**2. Modelagem conceitual**

Os dados utilizados são disponibilizados pelo IBGE, dentro de um conjunto de dataset com diversos temas e tópicos importantes relacionados a pesquisas e demarcação demográficas. As APIs para levantamento dos dados foram gerados através de *“Query Builders”* - são ferramentas ou bibliotecas que permitem construir consultas de maneira programática para interagir com uma API de maneira mais flexível e eficiente. Desta forma as URLs são geradas automaticamente ao escolher os checkboxes com os dados desejados, conforme o exemplo da imagem abaixo.

**Figura 1.** Exemplo de montagem de um Query Builder para gerar uma URL.

**Fonte.** IBGE - https://servicodados.ibge.gov.br/api/docs/agregados?versao=3#api-bq

Originalmente depois de montado o script para fazer a conexão com a API esses dados são no formato de arquivo JSON, o que é na qualidade de dados não estruturados por conta de ter um formato de lista dentro deste padrão e não é qualquer banco de dados que aceita tal esquema.

Critérios de aceite para os dados relacionados à pesquisas sobre “Projeção da População” :

Agregado. 7362 - Esperança de vida ao nascer e Taxa de mortalidade infantil, por sexo

Variáveis. 2503 - Esperança de vida ao nascer

1940 - Taxa de mortalidade infantil

Agregado. 7360 - Indicadores implícitos na projeção da população

Variáveis. 10600 - Nascimentos

10601 - Óbitos

10602 - Saldo migratório total

10603 - Saldo migratório interno

10604 - Saldo migratório internacional

10605 - Taxa de crescimento geométrico

10606 - Taxa bruta de natalidade

10607 - Taxa bruta de mortalidade

10608 - Taxa líquida de migração

2493 - Taxa de fecundidade total

10609 - Razão de dependência total

10610 - Razão de dependência de jovens

10611 - Razão de dependência de idosos

10612 - Índice de envelhecimento

10613 - Idade mediana

Agregado. 7358 - População, por sexo e idade

Variáveis. 606 - População

Agregado. 7365 - Proporção de pessoas, por grupo de idade

Variáveis. 10615 - Proporção de pessoas

Agregado. 7363 - Taxa específica de fecundidade, por grupo de idade da mãe

Variável. 10614 - Taxa específica de fecundidade

Período. 2018

Nível geográfico. N1 - Brasil

Localidades. 1 - Brasil

Sexo. Homens e Mulheres

Demais opções. Todas

**2.1 Ferramentas**

Nessa seção, serão abordadas as ferramentas utilizadas para a obtenção e tratamentos dos dados da pesquisa. Para códigos são o Visual Studio Code e GitHub, o banco de dados é Postgresql, visualização dos dados o PowerBI e fluxograma Lucidchart.

**2.1.1 Visual Studio Code**

Ambiente de desenvolvimento integrado leve e altamente popular, muito conhecido também como VS Code, foi criado pela Microsoft. Ele é usado principalmente para escrever códigos de programação, depurar e gerenciar projetos de software.

Todo o fluxo de scripts será realizado no VS Code, a implementação e execução ocorrerá neste software.

**2.1.2 GitHub**

Uma plataforma de desenvolvimento de software baseada na web que oferece uma série de serviços relacionados ao controle de versão, colaboração e hospedagem de projetos de código-fonte. Fundado em 2008, o GitHub se tornou uma das ferramentas mais utilizadas pelos desenvolvedores de todo o mundo.

Será usado como repositório dos códigos desenvolvidos no trabalho.

**2.1.3 Postgresql**

Normalmente é falado mais “Postgres” é um sistema de gerenciamento de banco de dados relacional de código-fonte aberto. Muito conhecido por ser altamente confiável, poderoso e altamente extensível.

O Postgresql será o banco de dados responsável por armazenar as tabelas e consequentemente os seus dados, todo tratamento entre as camadas no mesmo database.

**2.1.4 Power BI**

Power BI é um software criado pela Microsoft para soluções em análise de negócios que permite visualizações de dados e compartilhamentos em toda organização e permite inseri-los em um app ou site, conecta centenas de fontes de dados com dashboard e relatórios.

Neste trabalho o PowerBI será responsável por apresentar as visualizações dos dados obtidos através da API do IBGE, alimentado pelas tabelas do Postgres.

**2.1.5 Lucidchart**

É uma plataforma de criação de diagramas e fluxogramas online que permite aos usuários criar, editar uma variedade grande de diagramas, fluxos, organogramas, mapas mentais entre outros.

O fluxo inicial para toda estruturação posterior ao trabalho deu-se início pelo Lucidchart. Ponto de partida para progredir com os conceitos e ideias implementadas.

**2.2 Tecnologias**

Neste tópico serão abordadas as tecnologias que são as principais linguagens utilizadas para a obtenção e tratamentos dos dados da pesquisa. As linguagens utilizadas foram Python e SQL.

**2.2.1 Python**

É uma linguagem de programação de alto nível, de código aberto, interpretada e multiparadigma. Foi criada por Guido van Rossum e lançada pela primeira vez em 1991. Possui uma sintaxe simples e legível, o que a torna muito popular.

Python funcionará como a base de todos os códigos que conectam com API do IBGE, por fazer a integração direta com o banco de dados, tratar os dados obtidos e orquestrar todo o fluxo da ingestão desses dados.

**2.2.2 SQL**

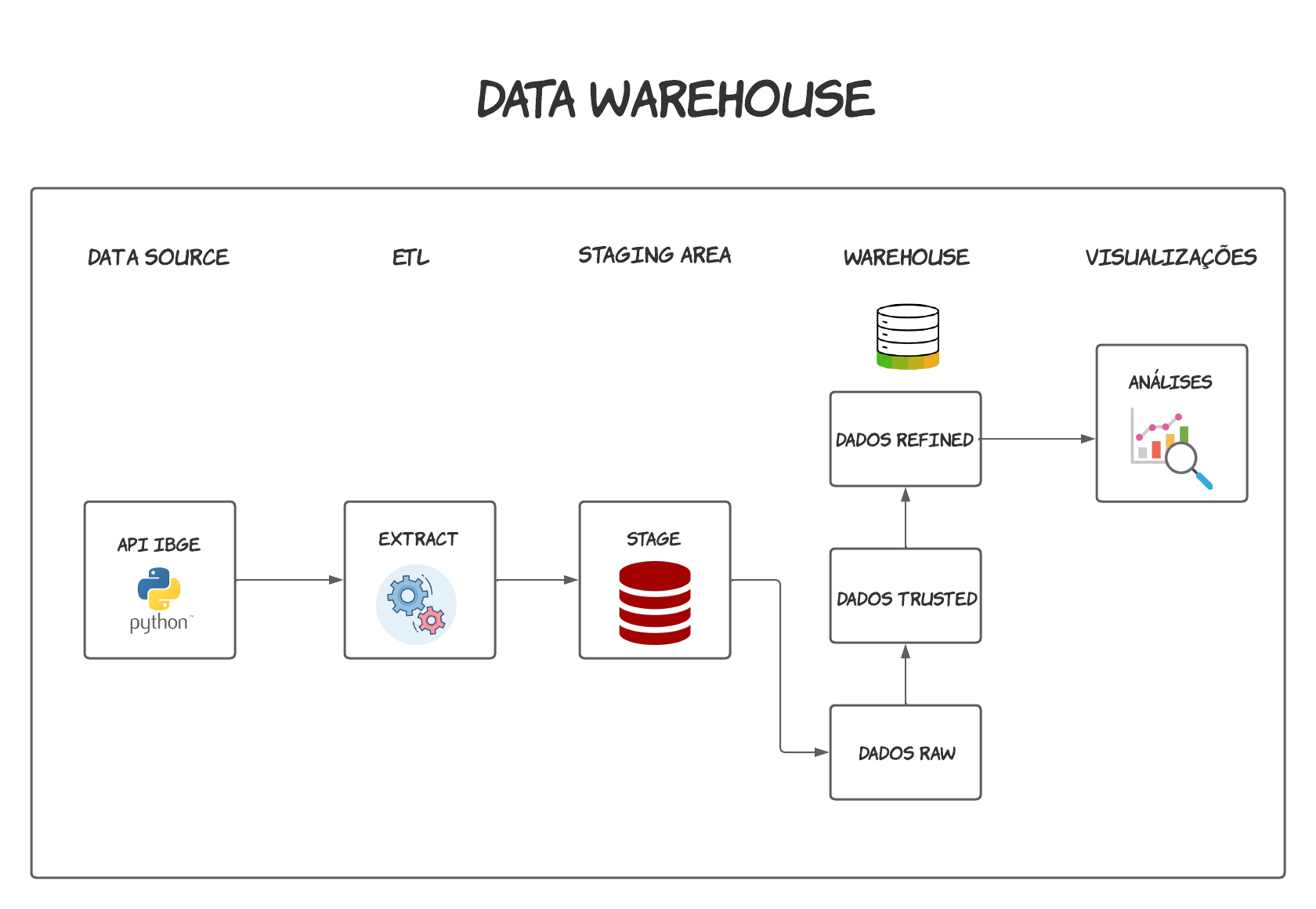
Linguagem de programação muito utilizada para gerenciar, consultar e manipular banco de dados relacionais. O SQL é a linguagem padrão para interação com os bancos de dados e é amplamente usada em sistemas de gerenciamento de banco de dados como MySQL, Postgres Oracle entre outros.

SQL será essencial para a criação e refinamento das tabelas, responsável também por integrar as camadas raw, trusted e refined, também dar a liberdade para fazer qualquer tipo de manipulação com os dados dentro do banco de dados.

**3. Aplicando conceitos e técnicas de engenharia de dados**

Nesta seção, serão abordados os passos seguidos para resolução do problema aplicando conceitos e técnicas de engenharia de dados.

Dado o início da abordagem, foi desenvolvido um fluxograma onde será a base para as etapas seguintes do processo, , funcionará com ponto de partida nas tomadas de decisões e estrutura do projeto, conforme figura abaixo:

**Figura 2.** Fluxograma base para ingestão dos dados

**Fonte.** Lucidchart - Autoria própria

**3.1 Ingestão de dados**

O processo de coleta, transferência e armazenamento de informações da fonte de dados do IBGE é o principal local onde contém os dados deste trabalho. Para definição da ingestão conforme a *Figura 2,* tem como base **Data Source,** **ETL e Staging Area.**

* **Data Source:** refere-se a origem de onde os dados são coletados ou extraídos para serem usados nas análises, processamentos e armazenamento. Neste caso a origem é a API do IBGE, a tecnologia utilizada para coletar esses dados é a linguagem python.

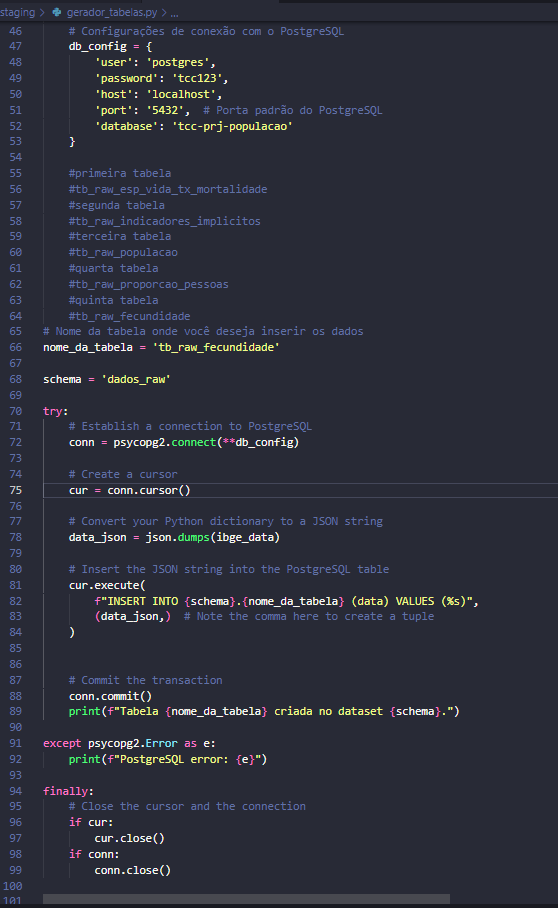
Os detalhes sobre esta primeira etapa do fluxo são referenciados pela documentação do site do IBGE (<https://servicodados.ibge.gov.br/api/docs/>), um ponto importante sobre, é que esta é uma API aberta, sendo assim não é necessário usar qualquer tipo de autenticação, basicamente é fazer uma requisição para URL base + endpoint e receber a resposta, ou seja os dados da origem.

* **ETL:** Nesta etapa é seguida alguns passos essenciais para coletar, transformar e carregar os dados. Utilizando bastante o conceito de ELT aqui, pois o processo de transformação é bem mais eficaz quando trabalho nas camadas do processo, então a distribuição fica dessa forma:
  + **Extração** - faz a chamada na API através URL base + endpoint, exemplo URL base é igual a https://servicodados.ibge.gov.br/api/v3/agregados/ e endpoint 7363/periodos/2018/variaveis/10614?localidades=N1[all]&classificacao=950[all]|1933[all].

**Figura 3.** Código em python responsável pela extração dos dados da API

**Fonte.** VS Code - Autoria própria

* + **Carregamento -** etapa responsável por integrar o banco de dados para carregar os dados diretamente no database do Postgres, para esta conexão é necessário estabelecer os parâmetros corretos, como user, password, host, port e database; é preciso ainda apontar a tabela desejada e camada.



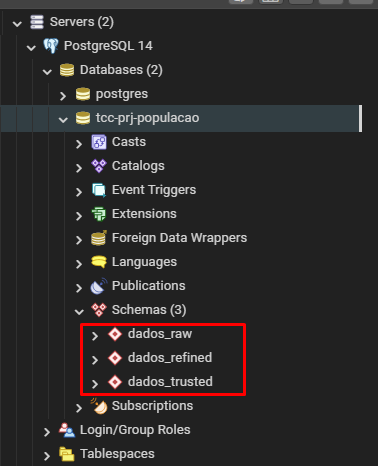
**Figura 4.** Código em python responsável pelo carregamento dos dados da API no Postgres

**Fonte.** VS Code - Autoria própria

* + **Transformação** - a etapa de transformação dos dados foi aplicada entre as camadas raw, trusted e refined que será abordada posteriormente na seção de tratamento dos dados.
* **Staging Area:** por meio do código em python acontece o esquema de staging area onde ao executar, os dados são temporariamente armazenados e movidos posteriormente para o data warehouse do fluxo apresentado na *Figura 2.* A função principal da área de staging é facilitar a transformação e a integração dos dados antes que sejam carregados para a próxima fase do processo, garantindo os dados que serão movidos.

**3.2 Tratamento dos dados**

Os dados já foram extraídos e preparados para ser tratados, existe algumas formas de se trabalhar a partir disso, pode-se coletar e inserir esses dados sem um tratamento prévio direto no banco de dados ou ele pode ser tratado e operado em camadas para uma disposição e maior interpretação de qual fase o dado se encontra. Neste trabalho foram implementadas três camadas raw, trusted e refined no database chamado “tcc-prj-população” .



**Figura 4.** Schema de camadas no Postgres

**Fonte.** Postgres - Autoria própria

**falar sobre o data warehouse principalmente**

**3.2.1 Camada Raw**

**tabelas**

**3.2.2 Camada Trusted**

**tabelas**

**3.2.3 Camada Refined**

**tabelas**

**3.3 Orquestração dos dados**

**relatar o fluxo com um todo, exemplificar uma rotina no airflow**

**3.3 Visualização de dados**

**powerbi**

**4. Orquestração de dados**

Nessa seção você deve deixar registrado todo o seu processo de orquestração de dados.

**5. Visualização de dados**

Nessa seção você deve deixar registrado todo o seu processo de visualização de dados.

**8. Links**

Aqui você deve disponibilizar os links para o vídeo com sua apresentação de 5 minutos e para o repositório contendo os dados utilizados no projeto, scripts criados, etc.

Link para o vídeo: youtube.com/...

Link para o repositório: github.com/...

**REFERÊNCIAS**

Um projeto/relatório técnico de Engenharia de Dados não requer revisão bibliográfica. Portanto, a inclusão das referências não é obrigatória. No entanto, caso você deseje incluir referências relacionadas às tecnologias ou às metodologias usadas em seu trabalho, relacione-as de acordo com o modelo a seguir.

SOBRENOME DO AUTOR, Nome do autor. **Título do livro ou artigo.** Cidade: Editora, ano.

SOBRENOME DO AUTOR, Nome do autor. **Título do livro ou artigo.** Cidade: Editora, ano.

SOBRENOME DO AUTOR, Nome do autor. **Título do livro ou artigo.** Cidade: Editora, ano.

SOBRENOME DO AUTOR, Nome do autor. **Título do livro ou artigo.** Cidade: Editora, ano.

SOBRENOME DO AUTOR, Nome do autor. **Título do livro ou artigo.** Cidade: Editora, ano.

SOBRENOME DO AUTOR, Nome do autor. **Título do livro ou artigo.** Cidade: Editora, ano.

**APÊNDICE**