Desafio Técnico - Engenheiro de Dados

Objetivo: O desafio consiste em desenvolver um processo de **extração, transformação e carregamento (ETL)** dos dados de desmatamento do último ano PRODES para o **bioma Cerrado**, obtidos a partir do GeoServer do TerraBrasilis (https://terrabrasilis.dpi.inpe.br/download-de-dados/). O objetivo final é armazenar os dados processados em um banco de dados **PostgreSQL com suporte a dados espaciais**, garantindo a integridade e a eficiência das consultas geoespaciais.

Requisitos do Desafio

O candidato deverá desenvolver uma solução que:

1. **Implementar uma classe para download dos dados do GeoServer do TerraBrasilis**
   - Criar uma classe modular para interagir com o serviço WFS (Web Feature Service).
   - A classe deve permitir parametrização de **bioma, tipo de dado (desmatamento) e período de interesse**.
   - A URL de acesso ao GeoServer deve ser formada corretamente, considerando os seguintes elementos:
     - `base_url`: "https://terrabrasilis.dpi.inpe.br/geoserver"
     - `workspace`: corresponde ao bioma desejado (exemplo: "prodes-cerrado-nb").
     - `layer`: representa a camada de informações do GeoServer (exemplo: "yearly_deforestation").
     - `full_url`: deve seguir o padrão **`{base_url}/{workspace}/{layer}/wfs`** para que a consulta ocorra corretamente.
   - Implementar um mecanismo que trate possíveis falhas na conexão com o servidor, garantindo tentativas adicionais antes de interromper a execução do script.
2. **Criar um script principal que utilize essa classe** para:
   - Obter apenas os dados de **desmatamento do PRODES** para o **bioma Cerrado**, considerando o período desejado.
   - Definir filtros de data e paginamento para otimizar a recuperação de informações.
   - Garantir que o formato de download permita futura integração com um banco de dados geoespacial.
   - Criar uma estrutura para **armazenamento dos dados em um banco de dados PostgreSQL**, assegurando que a tabela seja criada caso ainda não exista.
3. **Processar os dados baixados** e garantir que:
   - As geometrias estejam corretas e sejam manipuladas adequadamente.
   - As coordenadas sigam um padrão consistente e adequado para análise geoespacial.
   - O esquema de dados seja estruturado de forma a facilitar consultas e análises futuras.
   - Garantir que todas as geometrias da tabela final sejam **válidas**, implementando um método para corrigir geometrias inválidas, caso existam.
   - Criar índices na tabela para otimização das consultas SQL.
4. **Armazenar os dados no banco de dados PostgreSQL**:
   - Criar a tabela em um esquema específico `OUTPUT_SCHEMA = "raw_data"` para organização dos dados.
   - Implementar boas práticas de integração de dados geoespaciais.
   - Garantir a integridade das informações e evitar redundâncias.
   - Criar mecanismos para otimizar buscas e consultas no banco de dados.
5. **Gerar um Relatório** que explique:
   - O fluxo de trabalho e a abordagem utilizada na solução.
   - Quais foram as dificuldades encontradas e como foram resolvidas.
   - Quais foram as otimizações aplicadas para melhorar a performance.
   - Sugestões de melhorias para futuras versões do processo.

- Indicar quais soluções poderiam ser utilizadas para **corrigir geometrias inválidas**.
- Informar o **número total de áreas encontradas de desmatamento com geometrias válidas para todo o bioma Cerrado**.

Dicas Importantes:

- O GeoServer pode limitar a quantidade de registros retornados por requisição. A solução deve prever um **mecanismo de paginação** para garantir que todos os dados sejam recuperados corretamente.
- Para evitar problemas de conexão, é recomendado implementar um **sistema de tentativas automáticas**, garantindo que requisições falhas sejam repetidas antes de interromper a execução do processo.

Critérios de Avaliação

O desafio será avaliado com base nos seguintes aspectos:

1. **Exatidão e Funcionalidade:** A solução atende a todos os requisitos do desafio e funciona corretamente?
2. **Eficiência:** O código é otimizado para lidar com grandes volumes de dados?
3. **Boas Eficiência Práticas:** O código segue padrões profissionais de desenvolvimento, incluindo modularização, boas práticas de banco de dados e tratamento adequado de erros?
4. **Clareza e Documentação:** O relatório explica claramente as decisões tomadas e as soluções implementadas?
5. **Capacidade de Resolução de Problemas:** O candidato demonstrou habilidade para lidar com desafios e encontrar soluções eficazes?

Prazo de Entrega

- **Início do teste:** A partir do envio do teste
- **Prazo final:** 23/02/2024 até 23:59 (horário de Brasília)

Forma de Entrega

O candidato deve enviar um **repositório no GitHub** e um **arquivo compactado (.zip)** contendo:

1. O código-fonte da solução.
2. Um README com instruções de execução.
3. O relatório explicando o desenvolvimento.

Estamos ansiosos para avaliar sua solução e conhecer sua abordagem para este desafio! Boa sorte!