**Relatório de Automação RPA**

Este relatório apresenta uma análise detalhada sobre o script de RPA desenvolvido em Python, responsável por automatizar a sincronização de dados entre dois bancos de dados PostgreSQL. O objetivo é explicar o funcionamento, etapas e benefícios dessa automação.

**Introdução**

RPA (Robotic Process Automation) é uma tecnologia que permite automatizar tarefas repetitivas e baseadas em regras, executadas normalmente por humanos em sistemas digitais. O script analisado realiza uma automação completa de integração entre dois bancos de dados, garantindo que ambos permaneçam atualizados sem necessidade de intervenção manual.

**Objetivo do Script**

O código desenvolvido tem como propósito sincronizar informações entre duas bases de dados PostgreSQL, chamadas no código de origem e destino. O processo identifica novos registros, atualizações e exclusões para manter as tabelas idênticas em ambas as bases.

Em resumo, a automação substitui o processo manual de verificação e atualização de tabelas, reduzindo erros e economizando tempo.

**Estrutura e Funcionamento**

O script é dividido em etapas lógicas e bem definidas que seguem o modelo ETL (Extract, Transform, Load):

**• Extração (Extract):** os dados são obtidos da base de origem utilizando consultas SQL.

• **Transformação (Transform):** os dados são convertidos e tratados por funções auxiliares, como remoção de espaços, padronização de colunas e validação de tipos.

**• Carga (Load):** os registros são inseridos, atualizados ou removidos na base de destino, conforme as diferenças detectadas entre as tabelas.

**Principais Funções do Código**

**transform\_data(row):** realiza limpeza e padronização dos registros extraídos.

**filter\_columns\_exist(dest\_cursor, table, values):** garante que apenas colunas existentes na tabela de destino sejam utilizadas durante as inserções ou atualizações.

**sync\_tables(origem\_url, destino\_url, tabelas):** é a função principal responsável por todo o processo de sincronização, incluindo operações de INSERT, UPDATE e DELETE.

Essas funções trabalham de forma integrada, criando um fluxo automatizado e confiável de replicação de dados entre os ambientes.

**Considerações Técnicas**

O script utiliza bibliotecas como psycopg2 para conexão ao PostgreSQL e dotenv para leitura de variáveis de ambiente (.env). Ele foi projetado para rodar de forma segura e modular, permitindo ajustes simples conforme a necessidade de novas tabelas ou alterações estruturais.

A lógica implementada garante que o sistema mantenha a integridade dos dados, realizando commits e rollbacks automáticos em caso de falhas nas operações.

**Conclusão**

O projeto de automação RPA desenvolvido por Vitor Maciel representa uma solução prática e eficiente para sincronização de dados entre bancos PostgreSQL. A automação reduz erros, aumenta a produtividade e traz agilidade às operações de integração de sistemas.

O código segue boas práticas de desenvolvimento e está estruturado de forma clara e sustentável, podendo ser facilmente expandido para novas aplicações empresariais