**Documentação Detalhada do Script de Transferência para Zona RawDescrição**

Este script em Python automatiza a transferência de arquivos do bucket de processamento para o bucket raw no AWS S3, validando a integridade e unicidade dos dados através da comparação de hashes.

**Estrutura do Script**

**Classe cp**

Responsável por armazenar as configurações estáticas, como caminhos dos buckets do S3, detalhes de acesso ao banco de dados DynamoDB, configurações de e-mail e definições dos tipos de arquivos que serão processados.

**Classe zf**

Inclui métodos utilitários para a obtenção da data e hora atuais, geração de números aleatórios, cálculo do hash SHA-512 para verificação de dados, operações de consulta e inserção no banco de dados DynamoDB, além de funções para manipulação e validação de arquivos ZIP.

**Função copiar\_pasta**

Varre o bucket de processamento do S3 em busca de arquivos prontos para serem movidos para o bucket raw, executando a função copiar\_arquivo ou copiar\_arquivo\_csv para cada arquivo encontrado.

**Função copiar\_arquivo**

Transfere um arquivo do bucket de processamento para o bucket raw após a validação da unicidade do hash no DynamoDB. Se o arquivo já existir, o processo é ignorado para evitar duplicidade.

**Função copiar\_arquivo\_csv**

Similar à função copiar\_arquivo, mas utiliza um arquivo CSV para o controle de hashes em vez do DynamoDB.

**Função Principal main**

Orquestra o processo de transferência, iniciando pelo estabelecimento da conexão com a AWS e percorrendo os caminhos definidos para a execução da função copiar\_pasta.

**Pré-requisitos**

* Python 3.x
* Biblioteca boto3 instalada e configurada.
* Acesso à AWS com permissões apropriadas para os serviços S3 e DynamoDB.

**Configuração**

As variáveis na classe cp devem ser configuradas para refletir os buckets do S3 e as tabelas do DynamoDB utilizados na infraestrutura do usuário. As informações de e-mail devem ser fornecidas para permitir a notificação em caso de falhas.

**Execução**

Para executar o script, utilize um terminal ou ambiente de desenvolvimento Python, iniciando o script através do comando:

bash

Copy code

python nome\_do\_script.py

**Processo de Execução Detalhado**

1. A execução do script começa com a chamada à função main.
2. A função main imprime a mensagem de início, estabelece a conexão com os serviços da AWS e inicia a iteração pelos caminhos de arquivos definidos.
3. Para cada caminho, a função copiar\_pasta é chamada, que por sua vez verifica cada arquivo no bucket de processamento e decide se deve copiá-lo para o bucket raw.
4. A decisão de copiar é feita com base na validação do hash do arquivo, que é comparado com os registros existentes no DynamoDB ou no arquivo CSV.
5. Os arquivos validados são renomeados com um padrão que inclui informações da companhia aérea, tipo de arquivo e data de processamento, e então são transferidos para o bucket raw.
6. Os registros de controle de hash são atualizados no DynamoDB ou no arquivo CSV para refletir a transferência do arquivo.

**Manutenção e Monitoramento**

Os registros de operações e falhas são mantidos para permitir um monitoramento eficaz e a rápida resolução de problemas. Estes registros são essenciais para a manutenção contínua e devem ser regularmente revisados.

**Personalização**

O script pode ser adaptado para diferentes infraestruturas e necessidades de negócios, ajustando as variáveis e funções conforme necessário.

**Notas Finais**

É recomendável testar o script em um ambiente controlado antes de movê-lo para produção. Mantenha a documentação atualizada e acessível à equipe de operações para referência e suporte.

Lembre-se de que esta documentação é um guia e deve ser adaptada às especificações e necessidades do ambiente onde o script será executado.