

HEY, PRONTAS PARA O DESAFIO FINAL?

Desenvolver uma aplicação em Python para carga de arquivos em um banco de dados SQL e gerar relatórios estatísticos visando a descoberta de fraudes em cartão de crédito.

A nossa apresentação está disponível em:

https://www.canva.com/design/DAE5RRn7Ewc/GxgrssVmJyw-EnnClCWx7A/view?utm_content=DAE5RRn7Ewc&utm_campaign=designshare&utm_medium=link&utm_source=publishsharelink

DICAS DE HACKER

Pronto! Você recebeu um arquivo zip com vários arquivos mostrando possíveis movimentações bancárias:

- transaction-in-001.csv, transaction-in-002.csv ...
- transaction-out-001.csv, transaction-out-002.csv ...
- clients-001.csv, clients-002.csv ...
- Link Download:
<https://drive.google.com/file/d/1nXHnNt9dj03GB42SErcrNvZOzHwOyAcx/view?usp=sharing>

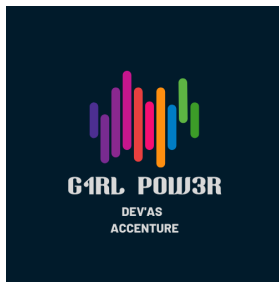
Seu objetivo inicial é analisar estes arquivos criando uma base de dados relacional para fazer a carga e depois analisá-la. O cartão fraudado, será aquele que tiver movimentações abaixo de 2 minutos de espaçamento entre as transações.

ENTREGAS MÍNIMAS

1. Script de migração em Python

O Script de migração foi desenvolvido a partir da tecnologia Pandas - biblioteca do python voltada para DataScience.

Após realizar a leitura dos arquivos .csv, realizamos o tratamento de alguns dados antes de realizarmos a inserção nas tabelas, sendo:



- Separamos o telefone dos clientes através de um método Split(), para criar uma nova tabela, a fim de nos proporcionar futuras análises mais completas. O telefone foi dividido em : código de país, código de área, e numeração.
- Tratamos todas as datas que foram inseridas no banco, para remover o fuso horário do servidor, para podermos garantir um cálculo mais preciso.
- Adaptamos o nosso código, para que ao rodar o nosso script você possa inserir os seus dados diretamente no terminal executado. Desse modo o código fica ajustado a qualquer futuro usuário.

2. Modelo de Entidades e Relacionamentos

Disponível para visualização no link:
<https://github.com/1-Dev-as-Accenture/DesafioDataAzure/blob/main/Documenta%C3%A7%C3%A3o/diagrama%20do%20banco%20de%20dados.pdf>

3. Relatórios de análise em SQL e PowerBI

No PowerBI, desenvolvemos a visualização dos dados por meio de análises das seguintes variáveis:

- Total dos valores Fraudados;
- Quantidade de Clientes Fraudados;
- Número de Transações Fraudadas;
- Valor Fraudado IN;
- Valor Fraudado OUT;
- Total de Clientes;
- Total de Transações;
- Valor Total das Transações;
- Porcentagem de Clientes Fraudados;
- Porcentagem de Transações Fraudadas;
- Mapa com a região das Fraudes;
- Gráfico de Fraudes OUT por período;
- Gráfico de Fraudes IN por período.

No SQL, desenvolvemos relatórios de análises das seguintes informações:



- Nomes dos clientes que cometeram fraudes
- Soma dos valores fraudados por tipo de transação
- Utilização de queries para validar os dados analisados no PowerBi

4. Códigos versionados no github.com

Todo o nosso código está disponível em:

<https://github.com/1-Dev-as-Accenture/DesafioDataAzure>

LÓGICA DESENVOLVIDA

1. Semântica do código

Todo o nosso código foi comentado para facilitar a compreensão da nossa lógica de desenvolvimento. Aproveitem as nossas dicas !

2. Lógica desenvolvida

- Desenvolvimento do script python para migração dos dados do arquivo .csv para o banco de dados
- Separação dos dados antes da realização da inserção no banco
 - Remover o fuso horário das datas_de_cadastro de clientes, transacoes_in e transacoes_out;
 - Particionar a string de telefones, separando País, DDD e Número de Telefone;
- Inserir os dados ajustados no banco;
- Retornar as informações das tabelas de Transacoes_in e Transacoes_out individualmente.
 - Os dados retornados foram convertidos em um array do tipo numpy, para melhor manipulação dos dados;



- ii. Os dados foram agrupados por clientes, e ordenados por data.
- iii. Foi realizado um loop, para comparação do cliente da transação atual com a próxima transação do registro;
- iv. Caso o cliente seja o mesmo, é realizado um cálculo entre a transação atual com a próxima transação, para verificar se a diferença entre elas, é igual ou menor (\leq) que 120 segundos. Se o cliente não for o mesmo, apenas é incrementado o contador, e partimos para a próxima transação.
- v. Se o resultado da comparação for "TRUE", a transação comparada é inserida na tabela FRAUDES, e loop prossegue para a próxima transação.

3. Documentação de utilização

Através do arquivo README.md, disponibilizamos um tutorial passo a passo para a utilização e reprodução dos nossos passos.

4. Hospedagem do sistema (nuvem Azure) para rodar os Scripts

Realizamos o desenvolvimento através de uma VM Azure com o sistema operacional Windows. Como todos os membros de nossa equipe utilizam a plataforma Windows, optamos pelo mesmo para podermos dinamizar o trabalho em equipe.