Desafio para área de Data Engineering

Parabéns pela aprovação na primeira etapa do processo seletivo para a área de Data Engineering da DHAUZ!

Seguindo com o nosso processo, nesta etapa você deverá realizar um desafio técnico, detalhado abaixo. O objetivo é avaliar sua capacidade de tratamento e manipulação dos dados, análise das informações e metodologia na resolução de problemas.

Um objetivo específico do teste é avaliar a sua capacidade de **manipular dados utilizando SQL**. Você pode utilizar a linguagem, ferramentas e frameworks que se sentir mais confortável para elaborar a solução. Recomendamos o desenvolvimento do racional e exploração em uma ferramenta, como por exemplo o Jupyter Notebook, que facilita a inclusão de comentários e gráficos explicativos.

Sinta-se livre para utilizar qualquer outra funcionalidade não listada acima que demonstre suas habilidades!

O projeto deve ser feito em um repositório no Github e o seu link enviado no final do desafio. Caso queira, fique à vontade para manter o repositório fechado e compartilhe o acesso com o nosso time.

Ingestão de novas informações

Você foi contratado pela DHAUZ como engenheiro de dados para trabalhar na operação de um ambiente analítico de dados e recebe como tarefa realizar a análise e manipulação de informações objetivando responder a algumas perguntas de negócio.

O seu desafio é realizar a leitura dessas informações de uma fonte externa, aplicar os processamentos necessários e então responder as questões levantadas.

O banco de dados que você irá trabalhar é um RDS da AWS (MySQL Community 8.0.28).
As credenciais de acesso são:

- Host name (endpoint): dhauz-instance.cutloqirhpd7.us-east-1.rds.amazonaws.com
- Port: 3306
- Username: candidate_user
- Password: D3@bGh664%$1VHv*

Você terá acesso (de leitura e criação de tabelas temporárias) ao banco “db_hiring_test” e às seguintes tabelas:

- db_hiring_test.raw_transactions_table
    - Essa tabela representa transações de criptomoedas. Então você possui disponível informações como carteira origem, carteira destino, total enviado, status da transação e data da transação.
    - Também está disponível uma coluna chamada ImportDate. Essa coluna representa o dia que um determinado lote de linhas foi capturado e carregado nessa tabela.
    - Exemplo da tabela:

IdTransaction	AddressOrigin	AddressDestination	TotalSent	Status	SentDate	ImportDate
ID1002	A-77	A-49	293,659.00	Confirmed	2021-01-08 13:34:04	2021-01-31 23:59:59
ID2014	A-24	A-58	542,285.00	Confirmed	2021-01-17 13:34:04	2021-01-31 23:59:59
ID1092	A-15	A-20	57,493.00	Confirmed	2021-01-03 03:07:57	2021-01-31 23:59:59
ID1603	A-84	A-59	883,745.00	Confirmed	2021-01-02 06:36:39	2021-01-31 23:59:59
ID253	A-86	A-44	194,591.00	Confirmed	2021-01-14 20:22:08	2021-01-31 23:59:59

- db_hiring_test.raw_transactions_fee
    - Essa tabela informa a lógica de cobrança de taxas sobre as transações.
    - Exemplo: Se uma transação de 2000 (totalSent) for realizada, uma taxa de 10% (valor da primeira faixa) seria contabilizada em cima do montante (2000) enviado.
    - Esse montante não é debitado e nem altera o valor da transação. Também não altera o saldo da carteira.
    - Exemplo da tabela:

range-start	range-end	fee-percentage
0.00	160000.00	10.00
160000.01	340000.00	8.00
340000.01	500000.00	6.00

Informações sobre o dataset:

- Os valores são aleatórios e não relacionadas com nenhuma informação real.

Abordagem recomendada:

- Recomendamos utilizar uma IDE em python para se comunicar com o banco de dados e enviar instruções SQL a fim de responder as perguntas abaixo.
- Por mais que seja possível manipular as informações utilizando python na sua própria máquina, é desejado que as manipulações com os dados sejam realizadas o máximo possível em SQL (isto é, executadas no servidor/banco). O objetivo disso é simular um ambiente de trabalho similar ao da oportunidade em questão.

Fase 1 – Análise sobre as transações:

Utilizando a tabela de transações, você deve implementar trechos de código que respondam as seguintes perguntas:

1. Qual é o endereço (carteira) com maior volume de transações enviadas?
2. Qual é o dia do mês com maior volume de transações realizadas?
3. Em qual dia da semana geralmente mais transações são realizadas?
4. Quais transações possuem condições atípicas e precisam ser validadas com o time responsável pela disponibilização dos dados?
5. Qual a carteira com o maior saldo final? (considere que todas as carteiras estejam zeradas no início das análises e que seja possível existir saldo negativo).

Importante:

- Considere sempre que as análises devem ser realizadas utilizando as informações mais atualizadas possíveis.
- Nessa questão não é necessário considerar a existência das taxas.

Fase 2 – Análise sobre as taxas:

Utilizando a tabela de transações e a de taxas, você deve implementar trechos de código que respondam as seguintes perguntas:

1. Considerando que a carteira origem é responsável por pagar as taxas de envio, qual carteira seria responsável pelo maior pagamento de taxas em janeiro de 2021?
2. E em fevereiro de 2021?
3. Qual é o id da transação com a maior taxa paga?
4. Qual é a média de taxa paga considerando todas as transações realizadas?

Importante:

- Considere sempre que as análises devem ser realizadas utilizando as informações mais atualizadas possíveis.

Fase 3 – Arquitetura (pergunta bônus):

Considere agora que você tenha que montar uma arquitetura em cloud, idealmente na Google Cloud Platform, para realizar de forma recorrente as tarefas acima citadas.

Descreve a arquitetura proposta. Quais soluções/serviços/tecnologias você utilizaria?

Essa questão é opcional. Fique à vontade para respondê-la caso você se sinta confortável.