

Desafio – Analytics Engineer

Sobre a SmarttBot

A missão da SmarttBot é **democratizar o investimento realmente inteligente no Brasil, empoderando o investidor com tecnologia de ponta.**

Fornecemos uma solução **amigável, segura, 100% online e precisa** para que os traders façam operações com robôs traders na B3. Por meio de nossa plataforma, o investidor tem acesso a estratégias que acompanham o mercado em tempo real e realizam **operações automatizadas** de acordo com a parametrização escolhida pelo usuário.

Dentro do universo de mais de 3 milhões de investidores pessoa física na B3, aproximadamente 300.000 são os chamados **day traders**, ou seja, investidores que realizam operações de compra e venda de um mesmo ativo em um mesmo dia, ganhando ou perdendo de acordo com a variação do preço deste ativo entre o momento em que realizou a compra e o momento em que realizou a venda. É este perfil de investidor (trader) que hoje a SmarttBot foca.

Este público em geral aceita um **alto risco** em seus investimentos em busca de um **alto retorno** em um **curto prazo**, muitas vezes de poucos dias, mas boa parte deles **não possui conhecimento** para criar as suas próprias estratégias de investimento (robôs).

Sobre o teste

O teste possui **2 fases** em que vamos avaliar você e sua forma de resolver o problema, pensar de forma holística, apresentar seu raciocínio de forma concisa e lógica e propor soluções factíveis para os mesmos, com foco em resultados.

As fases são: **Análise Exploratória e Implementação**. Vamos começar?

O problema:

O presente problema se refere aos dados de uma empresa que produz bicicletas.

O objetivo deste desafio é compreender os seus conhecimentos e experiência analisando os seguintes aspectos:

1. Fazer a modelagem conceitual dos dados;
2. Realizar uma análise exploratória dos dados;
3. Utilizar SQL para responder os questionamentos abaixo (pode usar alguma biblioteca que facilite o uso do SQL em um Jupyter Notebook);
4. Criar um relatório em qualquer ferramenta de visualização de dados.

Os seguintes arquivos devem ser importados (ETL) para o Jupyter Notebook, onde deverá ser feita toda a resolução do problema.

- Sales.SpecialOfferProduct.csv
- Production.Product.csv
- Sales.SalesOrderHeader.csv
- Sales.Customer.csv
- Person.Person.csv
- Sales.SalesOrderDetail.csv

Resolução:

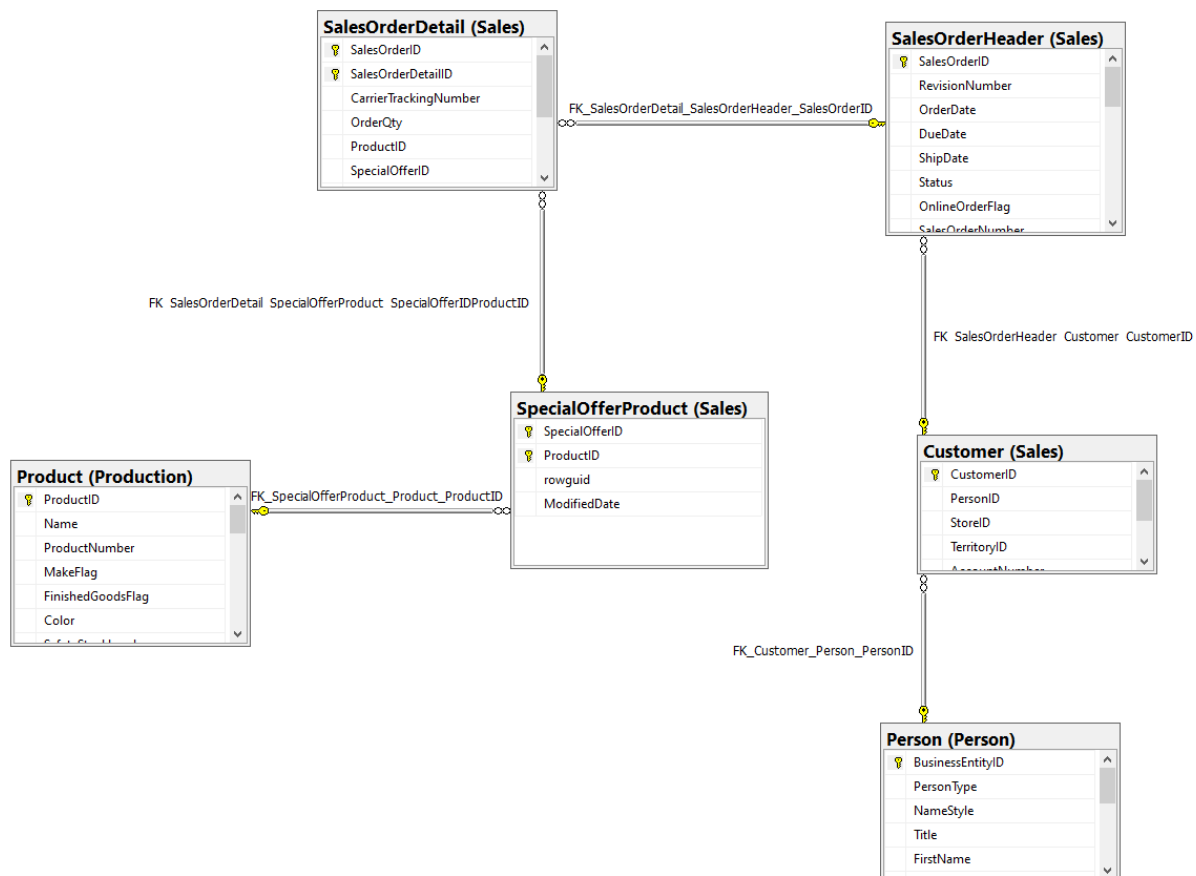
Para a realização deste teste, você deve utilizar alguma ferramenta para criação de notebooks, como por exemplo o Jupyter Notebook. Pedimos também que compartilhe conosco o código fonte (preferencialmente através de um repositório git público).

Além disso, inclua um arquivo README.md onde você deve compartilhar as decisões de análise, implementação e instruções sobre como executar o software. Caso tenha criado um relatório em alguma ferramenta de visualização de dados, inclua um screenshot do mesmo no arquivo.

Análise de Dados:

Com base na solução implantada responda aos seguintes questionamentos:

1. Escreva uma query que retorne a quantidade de linhas na tabela Sales.SalesOrderDetail pelo campo SalesOrderID, desde que tenham pelo menos três linhas de detalhes.
2. Escreva uma query que ligue as tabelas Sales.SalesOrderDetail, Sales.SpecialOfferProduct e Production.Product e retorne os 3 produtos (Name) mais vendidos (pela soma de OrderQty), agrupados pelo número de dias para manufatura (DaysToManufacture).
3. Escreva uma query ligando as tabelas Person.Person, Sales.Customer e Sales.SalesOrderHeader de forma a obter uma lista de nomes de clientes e uma contagem de pedidos efetuados.
4. Escreva uma query usando as tabelas Sales.SalesOrderHeader, Sales.SalesOrderDetail e Production.Product, de forma a obter a soma total de produtos (OrderQty) por ProductID e OrderDate.
5. Escreva uma query mostrando os campos SalesOrderID, OrderDate e TotalDue da tabela Sales.SalesOrderHeader. Obtenha apenas as linhas onde a ordem tenha sido feita durante o mês de setembro/2011 e o total devido esteja acima de 1.000. Ordene pelo total devido decrescente.



Critérios de Avaliação

O nosso teste para **Analytics Engineer** é feito para todos os níveis de senioridade. O que fará você se diferenciar no teste, é a forma que vai apresentar seus raciocínios.

Vamos te analisar nas 2 fases, através destes 5 itens:

- **Entendimento e Análise:** O quanto sua proposta de solução está coerente com o problema descrito
- **Técnica e Desenvolvimento:** O quão escalável, robusto e estável está sua solução, considerando custo
- **Aprendizagem e Melhorias:** Quais seriam os pontos de melhoria em próximas versões da solução.

Orientações

- Recomendamos que a linguagem utilizada seja Python (gostamos muito) mas isto não é impeditivo. Caso deseje realizar em outra linguagem, sinta-se à vontade.
- Utilizar alguma ferramenta de criação de notebooks, como o Jupyter Notebook, para a análise exploratória de dados.
- O uso de ferramentas e componentes open-source também é fortemente recomendado.

Observações

- A nossa avaliação é subjetiva e, por isso, te damos a liberdade de escolher a forma de representar os dados. Avaliamos que o tempo médio para terminar o foco principal desta tarefa é de cerca de 5 dias.
- Pense neste programa como uma biblioteca que será utilizada por outras pessoas. Logo, sua usabilidade e formato de distribuição também serão avaliados.
- Será avaliado o que for entregue. Assim, é importante documentar as dores e



dificuldades.

- Encontrando qualquer dúvida, você poderá entrar em contato conosco via email.

Prazo

Você terá 5 dias corridos para entrega do teste a partir do seu recebimento.

Boa sorte <3