

## Teste - Desenvolvedor Python

### Instruções Iniciais:

- Nossa intenção é avaliar tanto se seu código funciona quanto a forma que você programa. Então se empacar em alguma coisa, não desanime! Capriche no restante, deixando o código legível e autoexplicativo, fazendo uma boa documentação e não economizando nas boas práticas de programação.
- Se tiver qualquer dúvida, pode enviar um email para [dataengineering@hitechnologies.com.br](mailto:dataengineering@hitechnologies.com.br) com o título “Desafio - Engenharia de Dados” e vamos te responder com o maior prazer!
- Você tem até 7 dias corridos para concluir as atividades aqui solicitadas e enviar o projeto para a pessoa do RH responsável que te encaminhou o desafio.
- **Fique à vontade para utilizar tecnologias, frameworks e técnicas não citadas nas atividades, porém a linguagem base deverá ser Python.**

### Instruções do teste:

Crie uma aplicação para coletar postagens do Twitter em tempo real utilizando o “filtered stream” da api do Twitter com as seguintes regras:

- `{"value": "Futebol lang:pt", "tag": "Soccer rule"}`,
  - `{"value": "Saúde lang:pt", "tag": "Health rule"}`,
  - `{"value": "Comida lang:pt", "tag": "Food rule"}`
1. Quanto mais tweets melhor, mas fica a seu critério quando parar de coletar os dados. Para as análises que pedimos no final seria legal você coletar os dados ao longo do dia
  2. Modele e implemente um banco de dados para armazenar as informações retiradas pela API. Você tem liberdade para escolher o banco que quiser, seja ele local ou na nuvem, relacional ou não relacional. Lembre-se de pensar na forma de salvar os dados, pois isso vai influenciar na forma que eles serão usados depois.
  3. Faça um script separado que consulta a sua base de dados e responda às seguintes perguntas:
    - a. Qual o horário do tweet mais antigo e do mais recente para cada regra?
    - b. Qual o período do dia em que cada regra se torna mais frequente?
    - c. Qual o tweet mais longo em número de caracteres para cada regra? E o mais curto?

**Importante:** realize esses cálculos utilizando SQL/NoSQL, dependendo da base de dados que você implementou. Utilize Python apenas para se conectar ao banco e para apresentar os resultados, seja na forma de gráficos, prints, ou de outra forma que preferir.

4. Utilize o Github para publicar seu projeto. A documentação fica a seu critério.

### **DIFERENCIAIS:**

Se você quiser se destacar, aqui seguem alguns pontos para prestar atenção:

- Utilize o Git não apenas para publicar seu código, mas também para gerenciar as versões conforme você for avançando no projeto
- Inclua no README o que seu projeto precisa para funcionar, como a versão do Python e suas bibliotecas.
- Utilizar orientação a objetos não é obrigatório, mas recomendamos fortemente.
- Tente deixar seu código o mais auto explicativo possível. Mas lembre-se: comentários demais nem sempre contribuem para isso. Tente achar aquele bom e velho meio termo.
- Se tiver alguma dúvida, não hesite em mandar email pra gente. Teremos prazer em te ajudar.

**BOA SORTE!!**