

# AVALIAÇÃO DATAOPS

## Sumário

Avaliação DataOps .....	3
1. Desenvolvimento .....	4
1.1. Criação MongoDB Local .....	4
1.2. Criação do Pandas Dataframe .....	4
1.3. Salvar Pandas Dataframe no MongoDB .....	5
1.4. Criando Agregação no MongoDB .....	5
1.5. Agrupando informações .....	5
2. Salvando os scripts .....	6
3. Finalização .....	6
4. Desempenho .....	6

## Autoavaliação DataOps

O intuito deste teste é avaliar o desempenho em relação as ferramentas que serão utilizadas dentro do time de DataOps. Para este teste, iremos utilizar as seguintes ferramentas, MongoDB e Python.

Apenas para critério de conhecimento, responda as questões abaixo de acordo com sua percepção:

Como você considera seu nível de conhecimento em Python?

- ☐ NENHUM
- ☐ BÁSICO
- ☒ INTERMEDIÁRIO
- ☐ AVANÇADO

Como você considera seu nível de conhecimento em MongoDB?

- ☐ NENHUM
- ☒ BÁSICO
- ☐ INTERMEDIÁRIO
- ☐ AVANÇADO

Como você considera seu nível de conhecimento em ETL?

- ☐ NENHUM
- ☐ BÁSICO
- ☒ INTERMEDIÁRIO
- ☐ AVANÇADO

Como você considera seu nível de conhecimento em Pentaho?

- ☒ NENHUM
- ☐ BÁSICO
- ☐ INTERMEDIÁRIO
- ☐ AVANÇADO

## 1. O Desenvolvimento

Para o teste prático, será necessário criar um script em python, onde irá conter dois pandas dataframe populados com os dados pré-definidos, esses dataframes deverão ser enviados e salvos em um banco de dados MongoDB, na sequência, deverá ser desenvolvido uma agregação onde unira essas duas collection de acordo com o campo informado.

Para realizar estes testes, deverá seguir os seguintes passos descritos abaixo:

### 1.1. Criação MongoDB Local

Instalar o MongoDB local em sua máquina, e criar duas collection “Carros” e “Montadoras”, por enquanto estas collection ficaram vazias, e será populada nos próximos passos.

Obs. para facilitar a utilização do MongoDB, poderá utilizar o MongoDB Compass (Ou outro de sua preferência) para interface dos dados.

### 1.2. Criação do Pandas Dataframe

O início de teste se baseia em criar um arquivo em python contendo dois pandas dataframe, o primeiro onde será armazenado as informações do carro, e o segundo onde será armazenado as informações da montadora.

Para o primeiro dataframe, o resultado esperado é:

Carro	Cor	Montadora
Onix	Prata	Chevrolet
Polo	Branco	Volkswagen
Sandero	Prata	Renault
Fiesta	Vermelho	Ford
City	Preto	Honda

Para o segundo dataframe, o resultado esperado é:

Montadora	País
Chevrolet	EUA
Volkswagen	Alemanhã
Renault	França
Ford	EUA
Honda	Japão

## 1.3. Salvar Pandas Dataframe no MongoDB

Após criar os pandas dataframes, definidos no passo anterior, deves salvar-los nas collection correspondentes no MongoDB.

Para isso, será necessário criar uma conexão entre o Python e o MongoDB para persistir estas informações no Banco de dados.

Sendo o primeiro dataframe salvo na collection “Carros” e o segundo dataframe salvo na collection “Montadoras”.

## 1.4. Criando Agregação no MongoDB

Com os dados salvos no MongoDB, o objetivo atual, é fazer o relacionamento entre as duas collection “Carros” e “Montadoras”, para esta relação vamos utilizar a coluna “Montadora” disponibilizadas nas duas collection.

Com o resultado deste relacionamento, precisamos extrair o campo País, de modo a obter o seguinte resultado:

_id	Carro	Cor	Montadora	Montadoras	País
[id] 63dc1256c4ae345e2d161a0d	[img] Onix	[img] Prata	[img] Chevrolet	[img] [ 1 elements ]	[img] EUA
[id] 63dc126dc4ae345e2d161a0e	[img] Polo	[img] Branco	[img] Volkswagen	[img] [ 1 elements ]	[img] Alemanha
[id] 63dc128cc4ae345e2d161a0f	[img] Sandero	[img] Prata	[img] Renault	[img] [ 1 elements ]	[img] França
[id] 63dc129cc4ae345e2d161a10	[img] Fiesta	[img] Vermelho	[img] Ford	[img] [ 1 elements ]	[img] EUA
[id] 63dc12b3c4ae345e2d161a11	[img] City	[img] Preto	[img] Honda	[img] [ 1 elements ]	[img] Japão

## 1.5. Agrupando informações

Ainda dentro da mesma agregação criada no passo anterior, será necessário, agrupar os dados pelo campo de “País”, de modo que as informações sobre o carro, fique inserida dentro de um elemento chamado “Carros”, como representado na imagem abaixo:

_id	Carros
[img] Alemanha	[img] [ 1 elements ]
[img] Japão	[img] [ 1 elements ]
[img] França	[img] [ 1 elements ]
[img] EUA	[img] [ 2 elements ]

Após finalizar a agregação, salve-a em um arquivo .js

## 2. Salvando os scripts

Ao finalizar os passos anteriores, salve todos os scripts desenvolvidos, exporte as duas collection criadas no MongoDB, e faça o upload em um repositório GIT com o nome de sua preferência, certifique-se que o repositório esteja visível para o público. Finalize também o questionário inserido no início e fim deste documento, e salve-o no GIT.

Lembre-se, o repositório deverá conter os seguintes arquivos:

- Arquivo Python com os pandas dataframes salvando as informações no banco de dados.
- Agregação criada no MongoDB com os passos descritos neste documento.
- Collections do MongoDB exportadas.
- Arquivo Word contendo os questionários respondidos.

## 3. Finalização

Assim que finalizar, informe a pessoa responsável pelo processo seletivo, que as atividades desenvolvidas foram concluídas, e envie também, o link do repositório GIT onde as informações foram salvas.

## 4. Desempenho

Descreva abaixo quais foram os pontos mais fáceis, e mais difíceis encontrados nesta atividade, caso não tenha conseguido executar algum passo nesta atividade, especifique quais as dificuldades enfrentadas que o impediu de realizá-las:

*O ponto mais fácil foi o desenvolvimento do script em Python devido a ter utilizado mais vezes tanto em projetos pessoais/acadêmicos quanto em meus estágios.*

*Encontrei um pouco mais de dificuldade na utilização do MongoDB, pois utilizei poucas vezes a há um certo tempo, porém não foi impeditivo para a realização da atividade, precisei pesquisar, revisar pontos antigos e estudar alguns pontos novos para realizar a avaliação, o que também foi uma oportunidade de praticar e aprender um pouco mais sobre o Mongo em um contexto de engenharia de dados.*