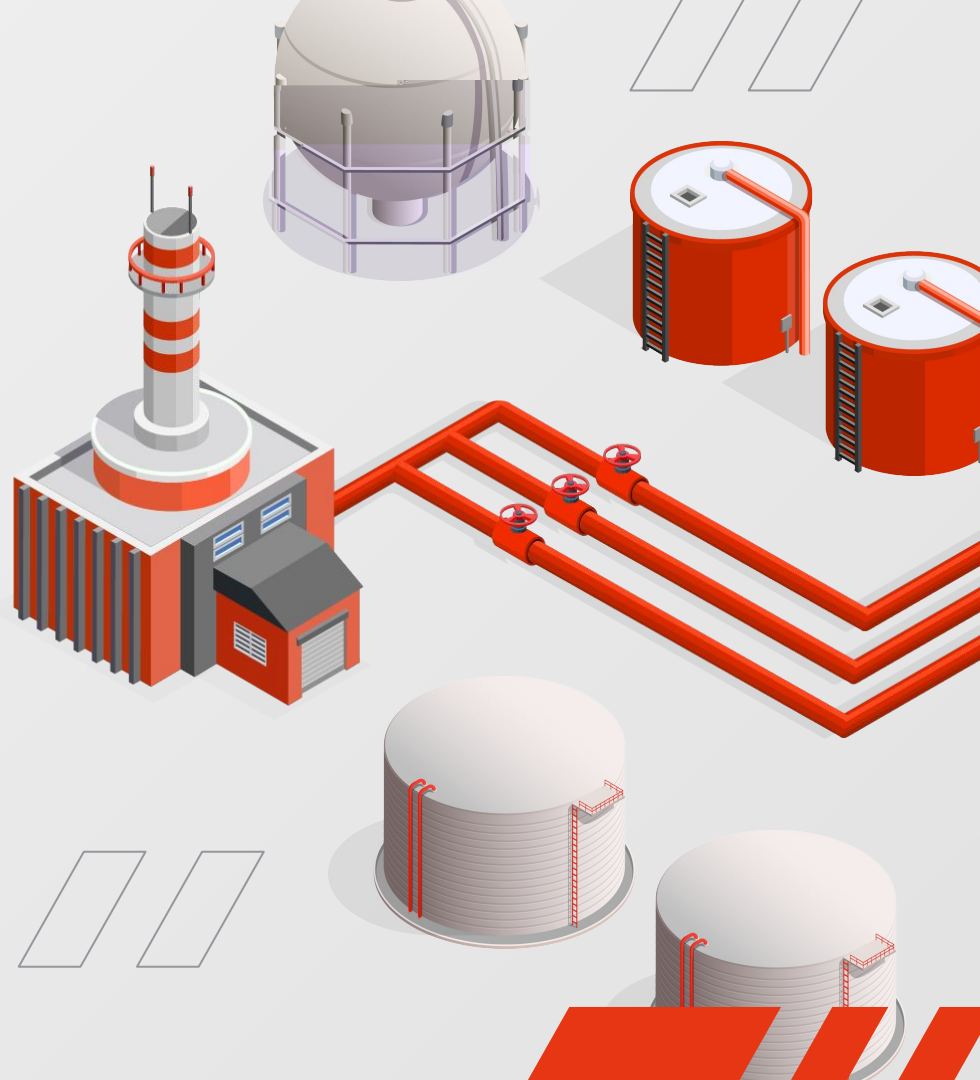


Análise de preços de combustíveis

1º Semestre - Brasil

P



Sumário

01

Dataset

Sobre o que se trata o dataset e seus dados

03

Resultados

Alguns dos exemplos obtidos através do projeto

02

Desenvolvimento

Metodologia utilizada no projeto

04

Referências

Agradecimentos e referências utilizadas



01

Dataset

Sobre o que se trata o dataset e
seus dados



Dataset

Contém preços de compra e venda diversos combustíveis automotivos no período do primeiro semestre de 2022.

"regiao", "estado", "municipio", "revenda", "cnpj_da_revenda",
"nome_da_ rua", "numero_ rua", "complemento", "bairro",
"cep", "produto", "data_da_coleta", "valor_de_venda",
"valor_de_compra", "unidade_de_medida", "bandeira"



02

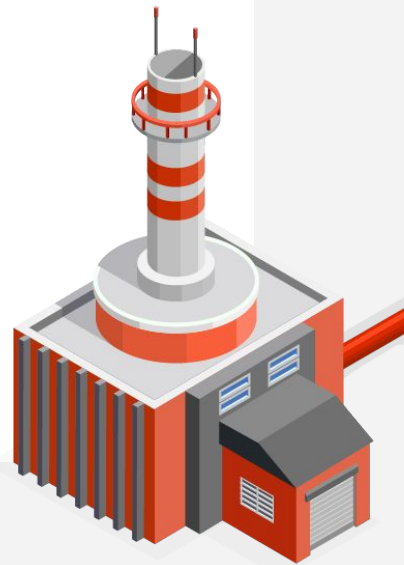
Desenvolvimento

Metodologia utilizada no projeto

MongoDB

```
...  
  
services:  
  mongodb:  
    container_name: mongodb  
    image: mongo  
    restart: always  
    environment:  
      - MONGO_INITDB_DATABASE=br_fuel_prices  
      - MONGO_INITDB_ROOT_USERNAME=br_fuel_prices  
      - MONGO_INITDB_ROOT_PASSWORD=br_fuel_prices  
    networks:  
      hadoop_network:  
        ipv4_address: 172.18.0.10  
  
...
```

docker-compose.yml



MongoDB

```
import csv
from pymongo import MongoClient

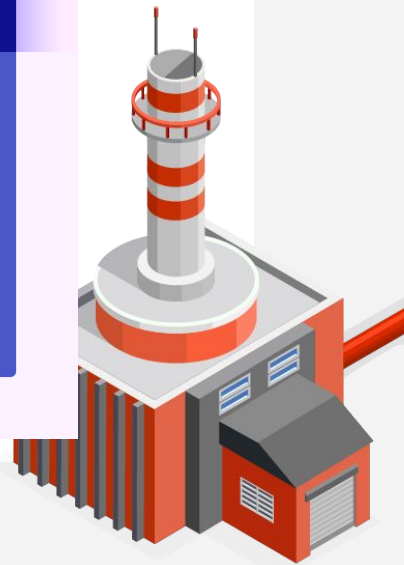
client = MongoClient('mongodb://br_fuel_prices:br_fuel_prices@172.18.0.10:27017')
db = client.br_fuel_prices
fuel_prices = db.fuel_prices

# Insercao dos dados no banco
with open('/root/precos-semestrais-ca-2022-01.csv', 'r') as csvfile:
    header =
    ["regiao", "estado", "municipio", "revenda", "cnpj_da_revenda", "nome_da_rua", "numero_rua", "complemento", "bairro", "cep", "produto", "data_da_coleta", "valor_de_venda", "valor_de_compra", "unidade_de_medida", "bandeira"]
    reader = csv.reader(csvfile, delimiter=',')
    next(reader)

    for row in reader:
        doc = {}
        for n in range(0, len(header)):
            doc[header[n]] = row[n]

        fuel_prices.insert_one(doc)
```

apps/seed_db.py



Spark

```
from pyspark.conf import SparkConf
from pyspark.context import SparkContext
from pyspark.sql import SparkSession

def init_spark():
    password = 'br_fuel_prices'
    user = 'br_fuel_prices'
    host = '172.18.0.10'
    db_auth = ''
    database = 'br_fuel_prices'
    collection = 'fuel_prices'
    mongo_conn = f"mongodb://{user}:{password}@{host}:{port}/{database}"

    conf = SparkConf()

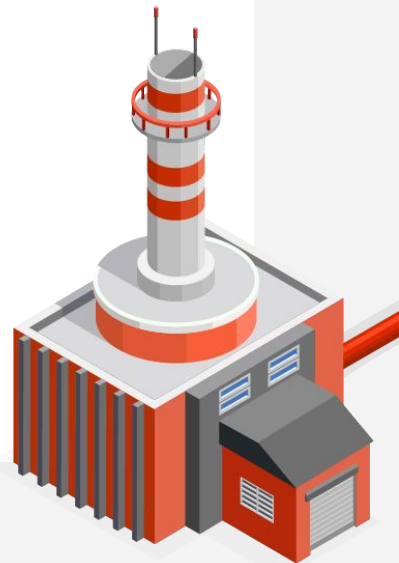
    conf.set("spark.mongodb.read.connection.uri", mongo_conn)
    conf.set("spark.mongodb.read.database", database)
    conf.set("spark.mongodb.read.collection", collection)

    conf.set("spark.mongodb.write.connection.uri", mongo_conn)
    conf.set("spark.mongodb.write.database", database)
    conf.set("spark.mongodb.write.collection", collection)
    conf.set("spark.mongodb.write.operationType", "update")

    SparkContext(conf=conf)

    return SparkSession \
        .builder \
        .appName('br_fuel_prices') \
        .getOrCreate()
```

apps/init_spark.py



Spark

```
from init_spark import init_spark

spark = init_spark()

df = spark.read.format("mongodb").load()

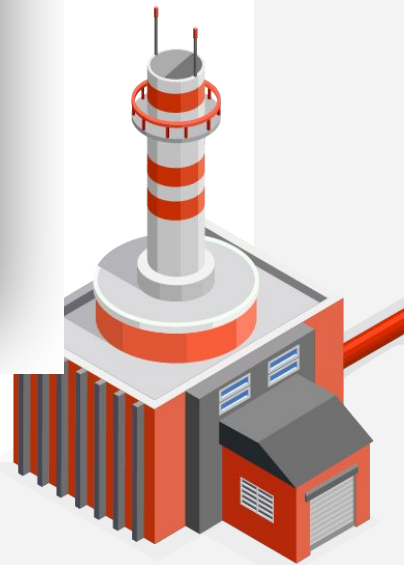
# Filtra por municipio e produto
df = df.filter((df['municipio'] == 'NATAL') & (df['produto'] == 'GASOLINA'))

# Mapeia e converte valores do produto de string para float
df = df.rdd.map(lambda x: (x.municipio, float(x.valor_de_venda.replace(",", ".")))).toDF(["municipio",
"valor"])

# Obtem a media dos valores para o municipio em questao
df = df.groupBy("municipio").avg("valor")

df.show()
```

apps/average_price_by_city.py





03

Resultados

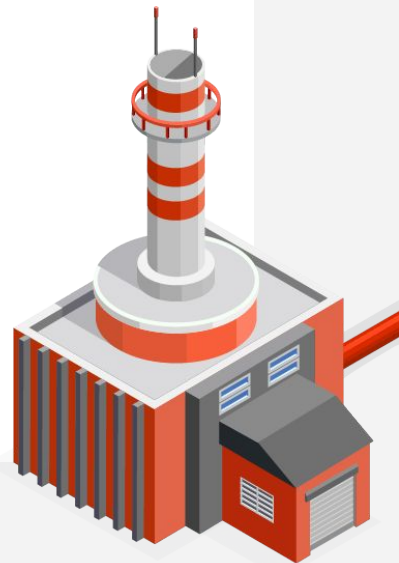
Alguns dos exemplos obtidos
através do projeto

Preço médio de gasolina por município

Comando

```
spark-submit --packages org.mongodb.spark:mongo-spark-connector:10.0.5 apps/average_price_by_city.py
```

```
+-----+-----+
|municipio|    avg(valor)|
+-----+-----+
|    NATAL|7.4131060606032|
+-----+-----+
```

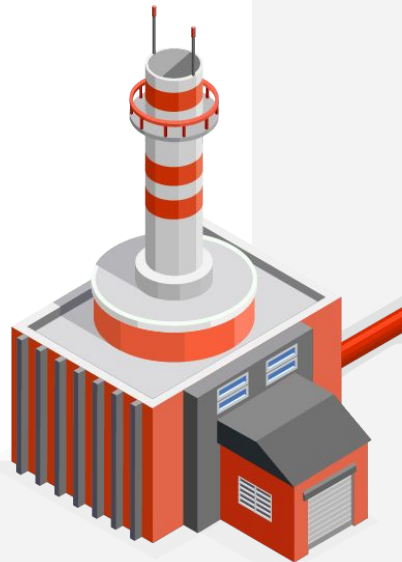


Maior preço de gasolina no semestre

Comando

```
spark-submit --packages org.mongodb.spark:mongo-spark-connector:10.0.5 apps/highest_price.py
```

```
+-----+-----+  
| produto|max(valor)|  
+-----+-----+  
| GASOLINA|      8.99|  
+-----+-----+
```

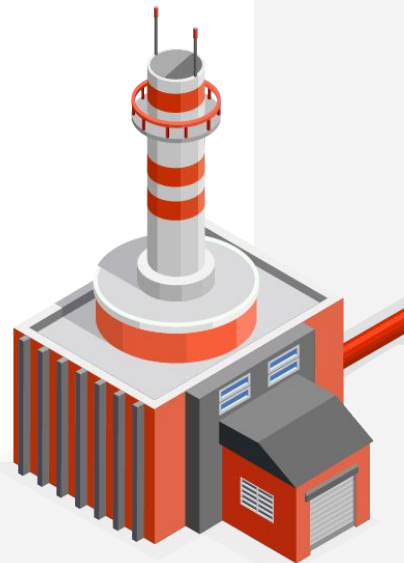


Menor preço de gasolina por região

Comando

```
spark-submit --packages org.mongodb.spark:mongo-spark-connector:10.0.5 apps/lowest_price_by_region.py
```

```
+-----+-----+  
|regiao|min(valor)|  
+-----+-----+  
|    NE|    5.59|  
|    N|    5.19|  
|    S|    5.56|  
|   SE|    5.48|  
|    CO|    5.79|  
+-----+-----+
```





Obrigado!

Referências

- Docker Hadoop Cluster por cmdviegas
- 1o Sem 2022 - Combustíveis Automotivos por dados.gov.br

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon** and infographics & images by **Freepik**

