Trabalho da matéria de Big Data do modulo de pós graduação Ciência de Dados, Técnicas e Aplicações do Instituto Mauá de Tecnologia.

Igor Amaral Correa

20.83992-8

Professor:

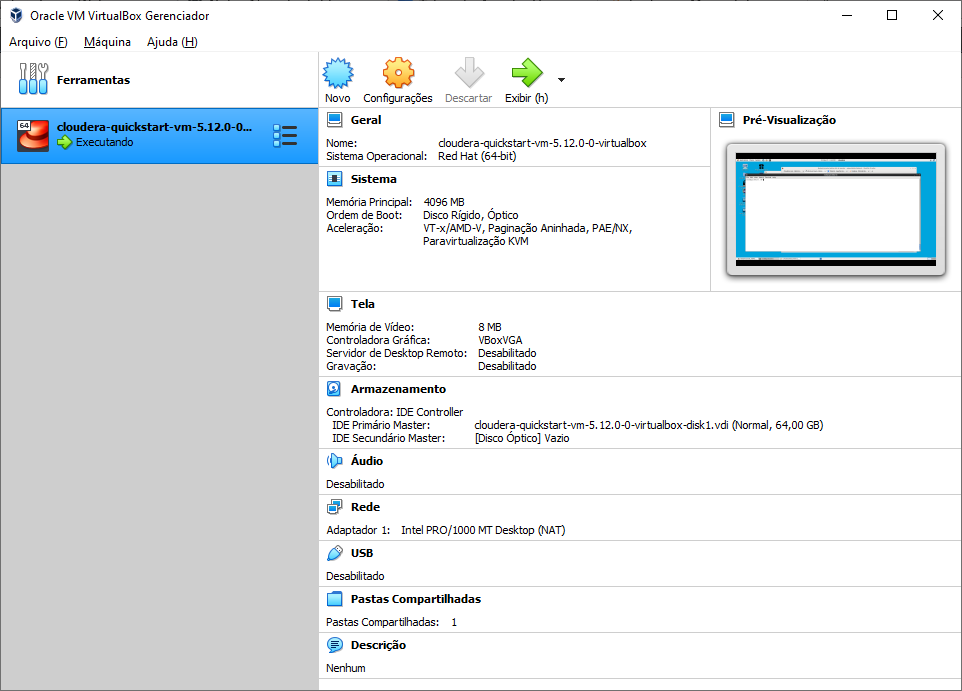
Murilo Zanini de carvalho

1. **Introdução**

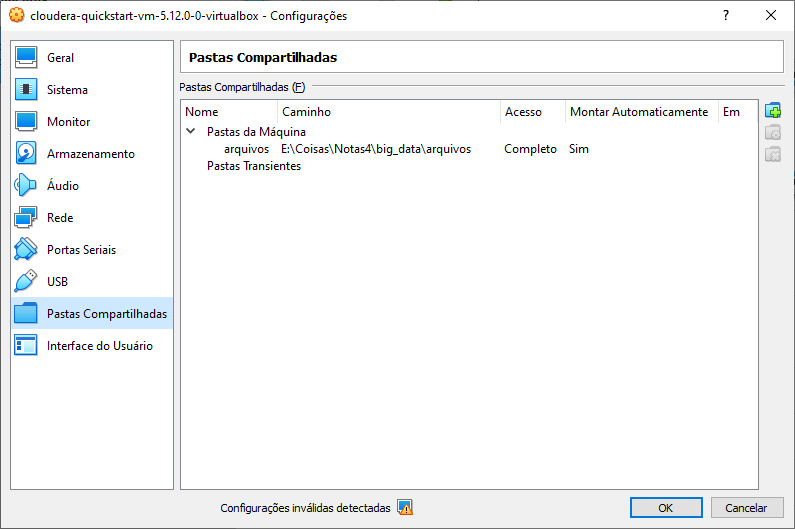
Esse trabalho tem como intuito aplicar os conceitos aprendidos em aula sobre Big Data, Hadoop e todo seu ecosistema, utilizando dados de contratos futuros disponibilizados pela B3, e será usada uma máquina virtual da Cloudera para todo o processo.

1. **Máquina**

A máquina utilizada foi a da Cloudera, com todas as ferramentas já baixadas (Hadoop, Spark, Hive, Hbase, etc...).

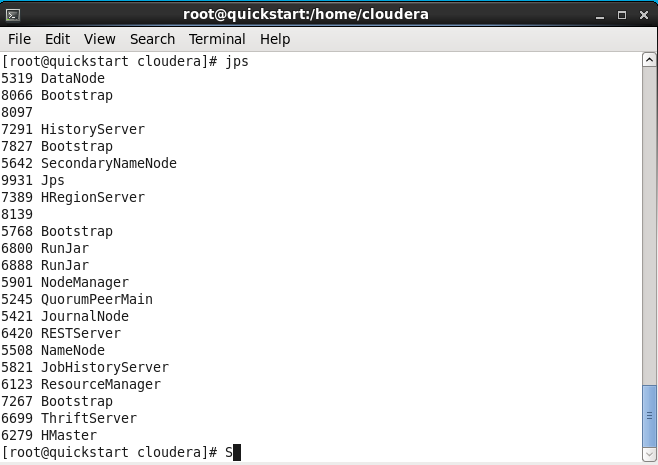


Foi compartilhada uma pasta com a máquina virtual para poder subir os arquivos.

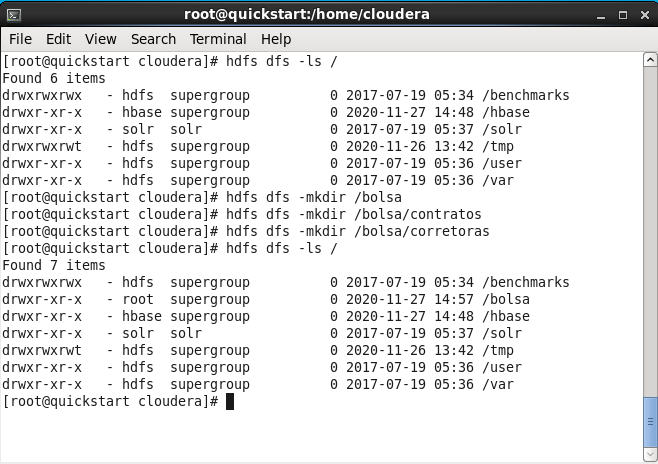


1. **HDFS**

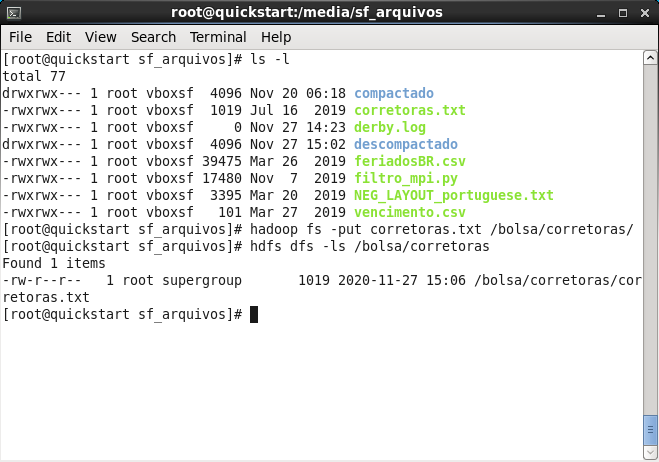
Logo ao ligar a máquina, conferimos se os recursos estão rodando com o comando ‘jps’.



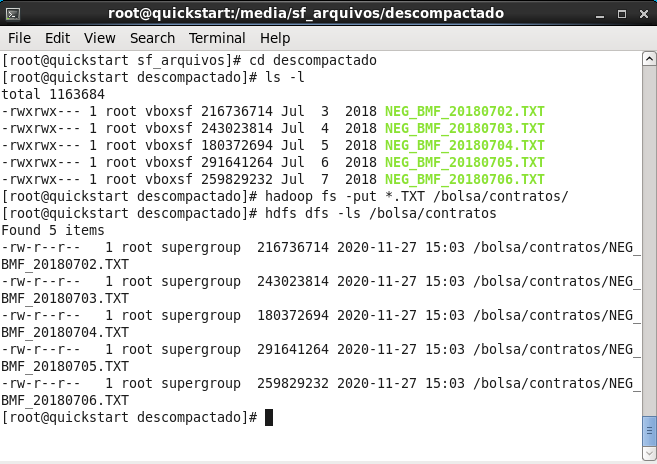
Os recursos que iremos utilizar já estão rodando, do HDFS (DataNode, SecondaryNameNode e NameNode) e do Yarn (NodeManager e ResourceManager), logo, vamos colocar nossos arquivos dentro do HDFS.



Passaremos primeiro os dados das corretoras.

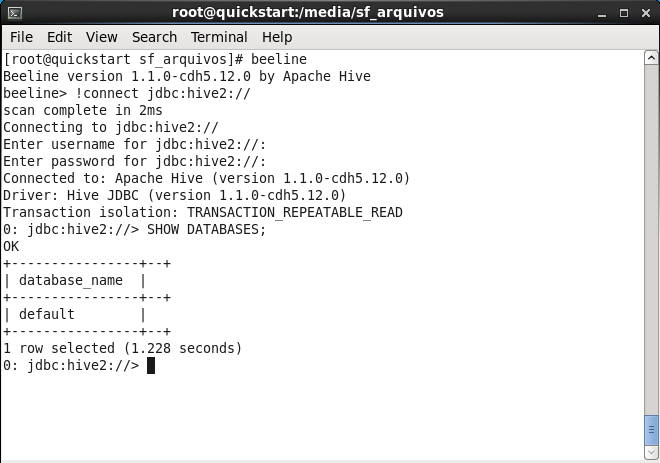


E agora os dados de contratos.

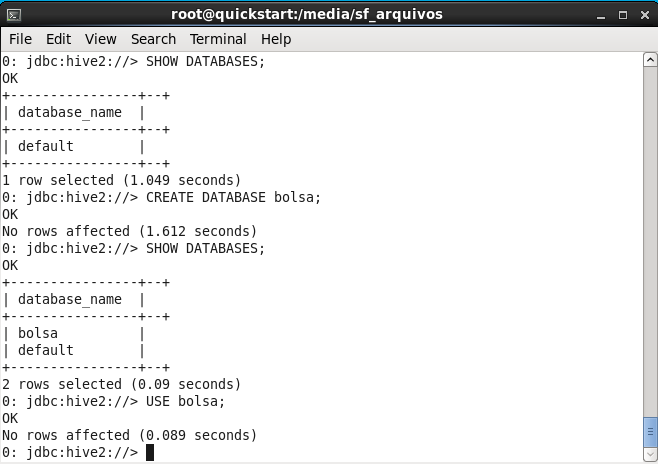


1. **Hive**

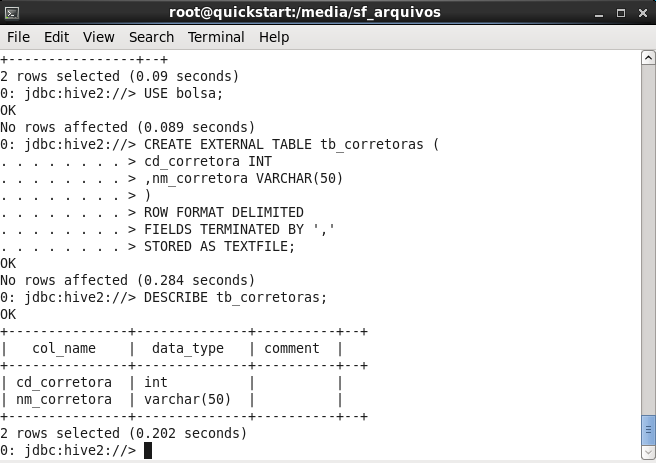
Com todos os arquivos no HDFS vamos entrar no Hive.

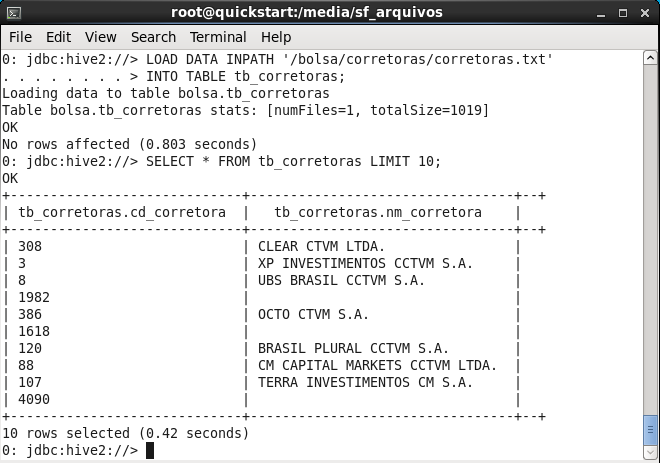


Criaremos nossa Database ‘bolsa’.

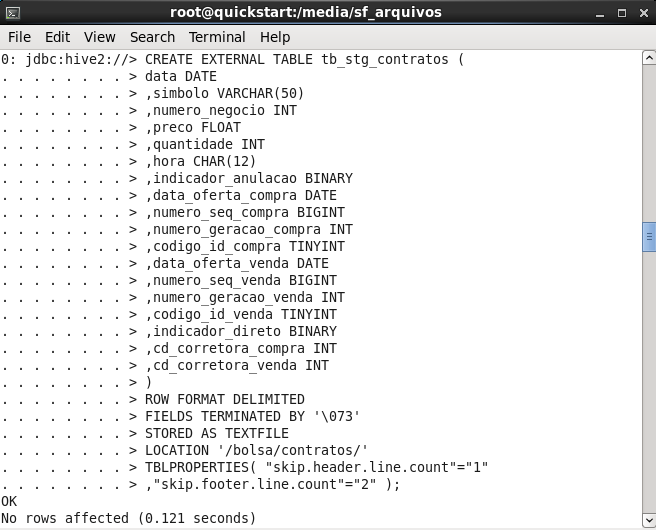


Criaremos primeiro a tabela de corretoras, onde temos o codigo e o nome delas.

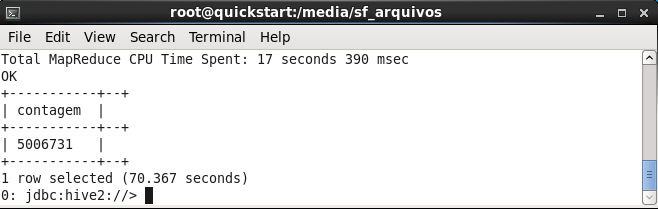




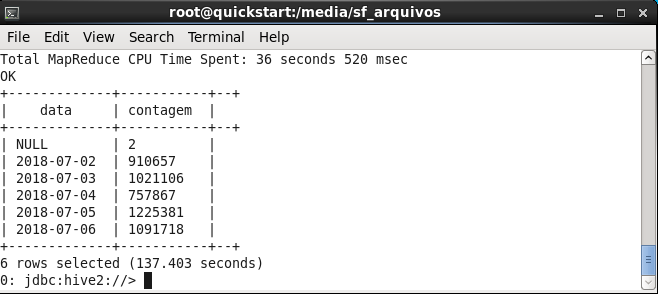
Agora criaremos uma tabela stage, onde serão carregados os dados, e logo após uma tabela final particionada pela data, que iremos carregar atravez da tabela stage.



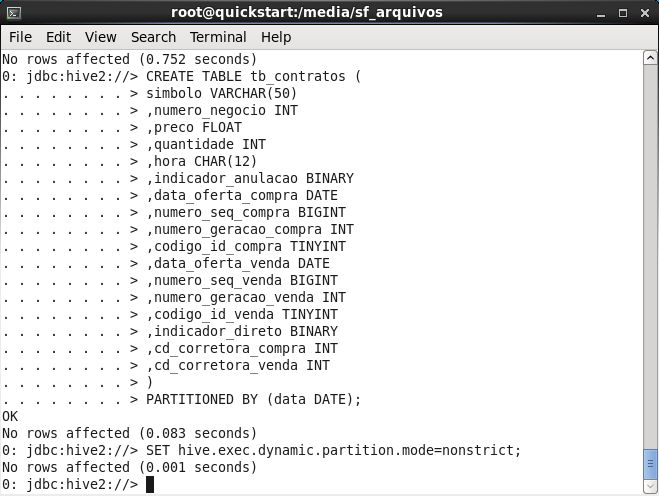
Vendo quantas linhas tem a tabela ( SELECT COUNT(\*) contagem FROM tb\_stg\_contratos; )



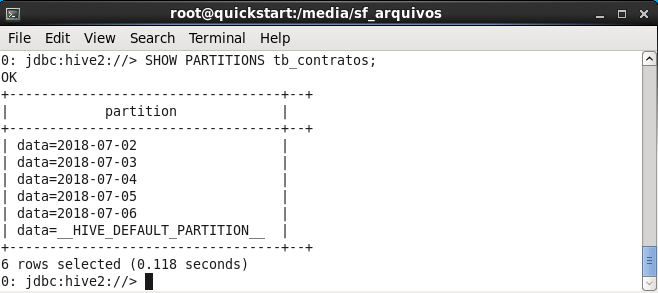
Contando quantas linhas temos por dia ( SELECT data, COUNT(\*) contagem FROM tb\_stg\_contratos GROUP BY data ORDER BY data; )



Agora criaremos a tabela particionada e vamos carregar os dados da tabela stage.







Contando quantas linhas temos por dia ( SELECT data, COUNT(\*) contagem FROM tb\_contratos GROUP BY data ORDER BY data; )



Vamos agora fazer uma consulta utilizando as duas tabelas, afim de encontrar as 3 corretoras que compraram mais contratos para cada um dos dias, utilizaremos o código SQL abaixo.

WITH cte AS (

SELECT con.data ,SUM(con.quantidade) sum\_qnt ,cor.nm\_corretora

FROM tb\_contratos con

INNER JOIN tb\_corretoras cor ON con.cd\_corretora\_compra = cor.cd\_corretora

WHERE con.data = '2018-07-02'

GROUP BY con.data, cor.nm\_corretora

ORDER BY sum\_qnt DESC

LIMIT 3

UNION ALL

SELECT con.data ,SUM(con.quantidade) sum\_qnt ,cor.nm\_corretora

FROM tb\_contratos con

INNER JOIN tb\_corretoras cor ON con.cd\_corretora\_compra = cor.cd\_corretora

WHERE con.data = '2018-07-02'

GROUP BY con.data, cor.nm\_corretora

ORDER BY sum\_qnt DESC

LIMIT 3

UNION ALL

SELECT con.data ,SUM(con.quantidade) sum\_qnt ,cor.nm\_corretora

FROM tb\_contratos con

INNER JOIN tb\_corretoras cor ON con.cd\_corretora\_compra = cor.cd\_corretora

WHERE con.data = '2018-07-02'

GROUP BY con.data, cor.nm\_corretora

ORDER BY sum\_qnt DESC

LIMIT 3

UNION ALL

SELECT con.data ,SUM(con.quantidade) sum\_qnt ,cor.nm\_corretora

FROM tb\_contratos con

INNER JOIN tb\_corretoras cor ON con.cd\_corretora\_compra = cor.cd\_corretora

WHERE con.data = '2018-07-02'

GROUP BY con.data, cor.nm\_corretora

ORDER BY sum\_qnt DESC

LIMIT 3

UNION ALL

SELECT con.data ,SUM(con.quantidade) sum\_qnt ,cor.nm\_corretora

FROM tb\_contratos con

INNER JOIN tb\_corretoras cor ON con.cd\_corretora\_compra = cor.cd\_corretora

WHERE con.data = '2018-07-02'

GROUP BY con.data, cor.nm\_corretora

ORDER BY sum\_qnt DESC

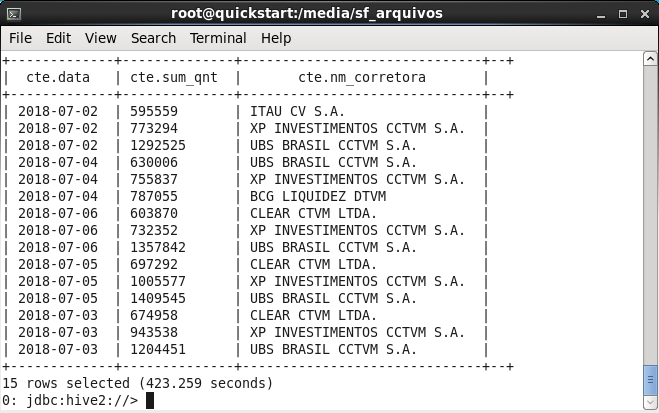
LIMIT 3

)

SELECT \*

FROM cte

ORDER BY 1, 2 DESC;



1. **Conclusão**

Hadoop é uma ferramenta que está para ficar, principalmente com o aumento da utilização e do armazenamento de dados subindo exponencialmente, e Hive é uma ferramenta do seu ecosistema muito útil e fácil, que permite rodar jobs de MapReduce utilizando linguagem SQL, como pudemos ver.