

Brito, Ricardo.

Administrador de Banco de Dados

<https://www.linkedin.com/in/ricardorbrito/>

<https://github.com/ricardorbrito/ArquiteturaLambda>

Fonte Original: Data Lakes: Big Data com Arquitetura Lambda – professor Fernando Amaral

# Tutorial

Este é um passo a passo de como implementar na PRÁTICA um Data Lake e um Data Warehouse com ARQUITETURA LAMBDA, utilizando técnicas de ingestão de dados oriundos de uma base transacional para um sistema de arquivos distribuídos HDFS HADOOP, com SPARK, SQOOP e HIVE.

Supomos que uma empresa tem um banco de dados no MYSQL onde há uma tabela de clientes gigantesca e que a todo momento sofre alterações e inserções de novos registros pelos vários sistemas que acessam essa base de dados, além disso, o time de BI necessita de informações em tempo real pelo menos 5 (cinco) vezes ao dia, uma delas é o total de clientes por cidade e por estado e que estas informações não podem ser obtidas diretamente na fonte de dados TRANSACIONAL (MySql).

Diante desta demanda, vamos criar um processo de ingestão de dados brutos não minerados em um Data Lake, em seguida criaremos um Data Warehouse com um bando de dados MASTER, onde os dados serão tratados e disponibilizados para consultas. Estes procedimentos serão atualizados de formas automáticas com dois tipos de processamentos, batch e stream, evitando assim a consulta direto no banco de transação.

## **Esse tutorial está dividido em 05 (cinco) etapas.**

1. Na primeira etapa vamos criar um JOB para realizar a ingestão de dados do banco original (MYSQL) para um Data Lake em um sistema de arquivos distribuídos HADDOP utilizando a ferramenta SQOOP.
2. Já na segunda etapa, iremos criar um Data WareHouse, onde serão importados os dados do Data Lake para um banco de dados MASTER, para isso, utilizaremos o HIVE.
3. Na terceira etapa, iremos construir as duas camadas de processamentos. BatchLayer e SpeedLayer.
4. Nesta etapa, construiremos duas VIEWS, a primeira com total de clientes por cidade e a segunda com total de clientes por estado.
5. E por fim, faremos a automatização dos scripts para rodarem de forma automática.

**É importante seguir os passos corretamente na sequência.**

**Então, vamos para a prática  
#pracima.**

## 1. A primeira etapa:

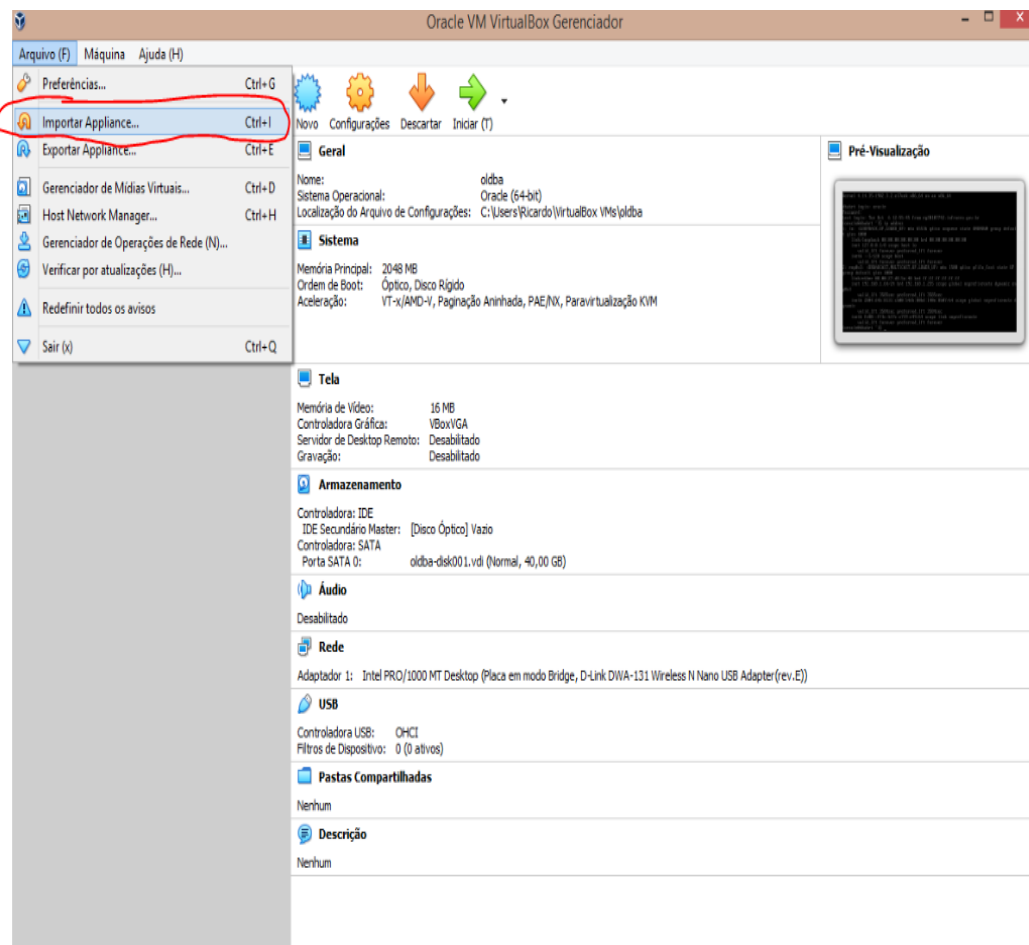
- ❖ Criação de um DATA LAKE em um sistema de arquivos distribuídos utilizando as ferramentas MYSQL, HADOOP E SQOOP.

**OBSERVAÇÃO: Não vamos abordar aqui a instalação e configuração das ferramentas porque não é esse o foco.**

Faça o download da VM com todas os requisitos e ferramentas já instaladas no link abaixo:

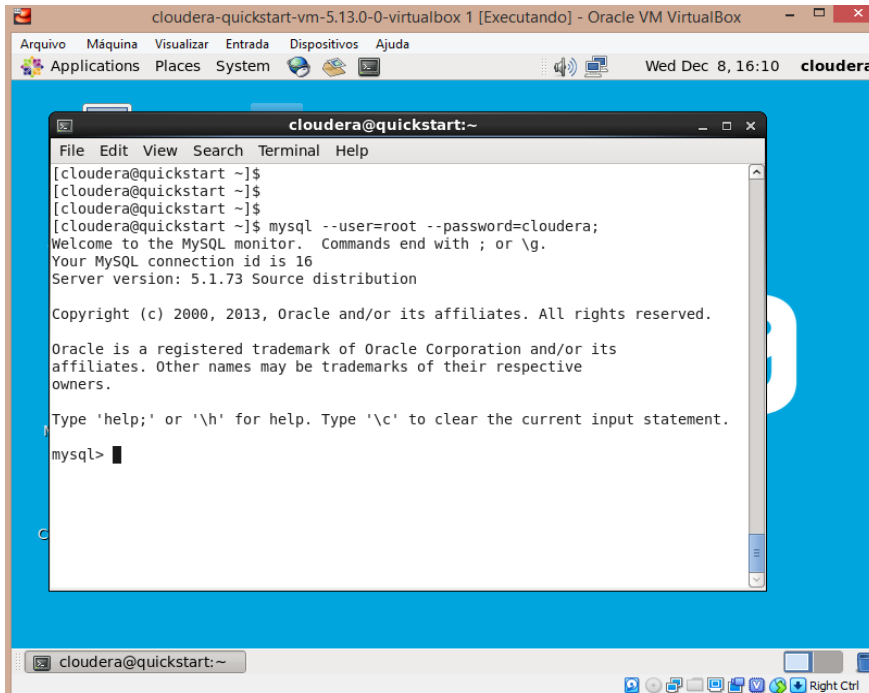
<https://drive.google.com/file/d/1mNOjzicy31axLKe5yHDK7Viv-FiOoDGN/view?usp=sharing>

Após O DOWNLOAD, faça o importe do appliance para dentro do seu virtual box e vamos a prática.

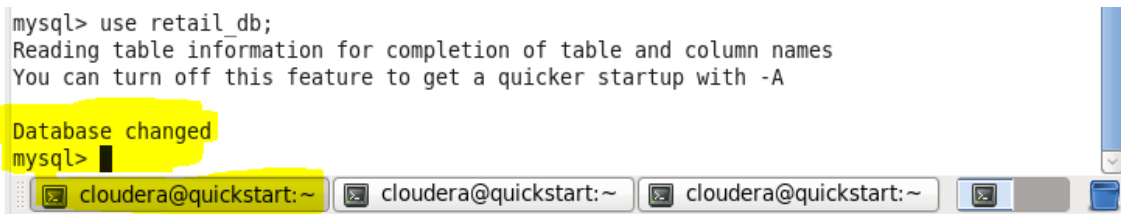


## PASSO 01 – ACESSE O MYSQL E ALTERE A BASE DE DADOS:

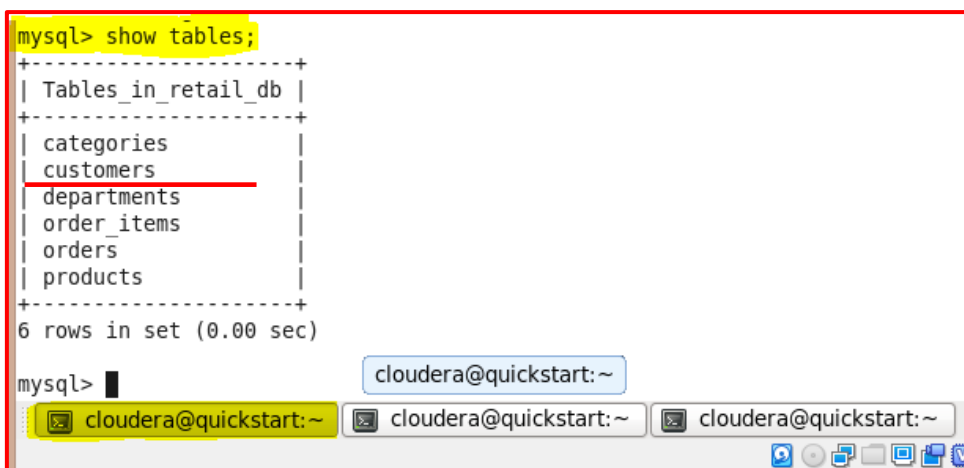
- Abra o terminal e digite: `mysql --user=root --password=cloudera;`



- Altere o banco de dados para `retail_db`:  
`use retail_db;`



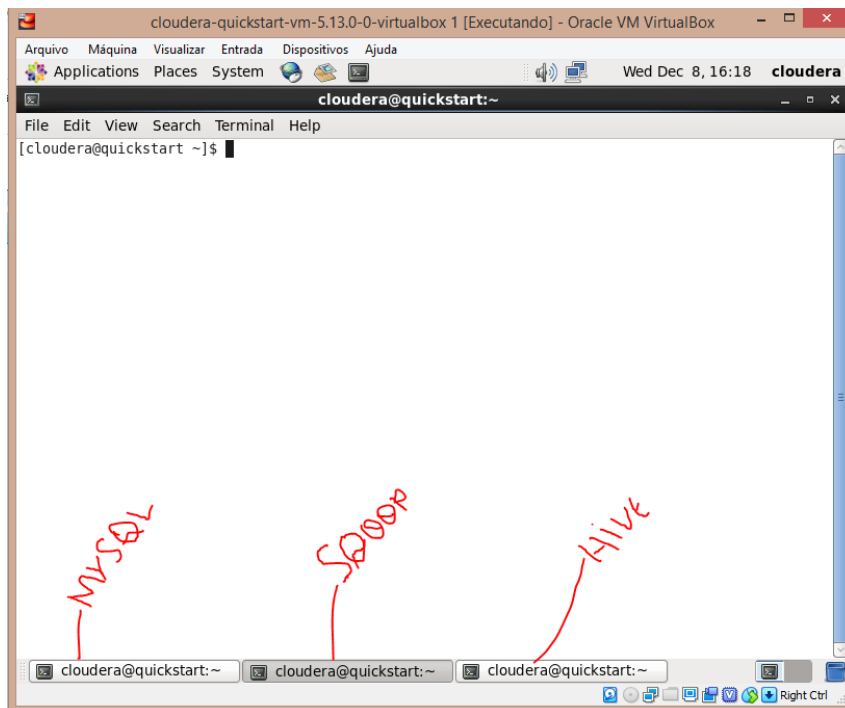
- Visualize as tabelas com o comando: `show tables;`



Note que existe a tabela **CUSTOMERS**, ela será nossa tabela de exemplo.

**PASSO 02 -** Agora vamos fazer a ingestão de dados no HDFS com o Sqoop. Iremos trazer a tabela de clientes (**CUSTOMERS**), para dentro do sistema de arquivos distribuídos HADOOP.

**Importante:** Abra mais dois terminais na sua VM para você interagir com as telas e assim poder ver as atualizações dos dados, de acordo com a ilustração abaixo:



- No segundo terminal (SQOOP) digite o seguinte comando:  
`sqoop import --connect jdbc:mysql://localhost/retail_db --table customers --username root --password cloudera --check-column customer_id --incremental append --last-value 0 -m 1`

**PASSO 03 -** vamos verificar se os dados foram importados para o HDFS, digite o comando a seguir:

```
sudo hdfs dfs -ls /user/cloudera;
```

```
[cloudera@quickstart ~]$ sudo hdfs dfs -ls /user/cloudera;  
Found 2 items  
drwxr-xr-x - cloudera cloudera      0 2021-12-08 16:22 /user/cloudera/ sqoop  
drwxr-xr-x - cloudera cloudera      0 2021-12-08 16:22 /user/cloudera/customers  
[cloudera@quickstart ~]$
```

Observe que foi criado um diretório com o nome da tabela **CUSTOMERS**.

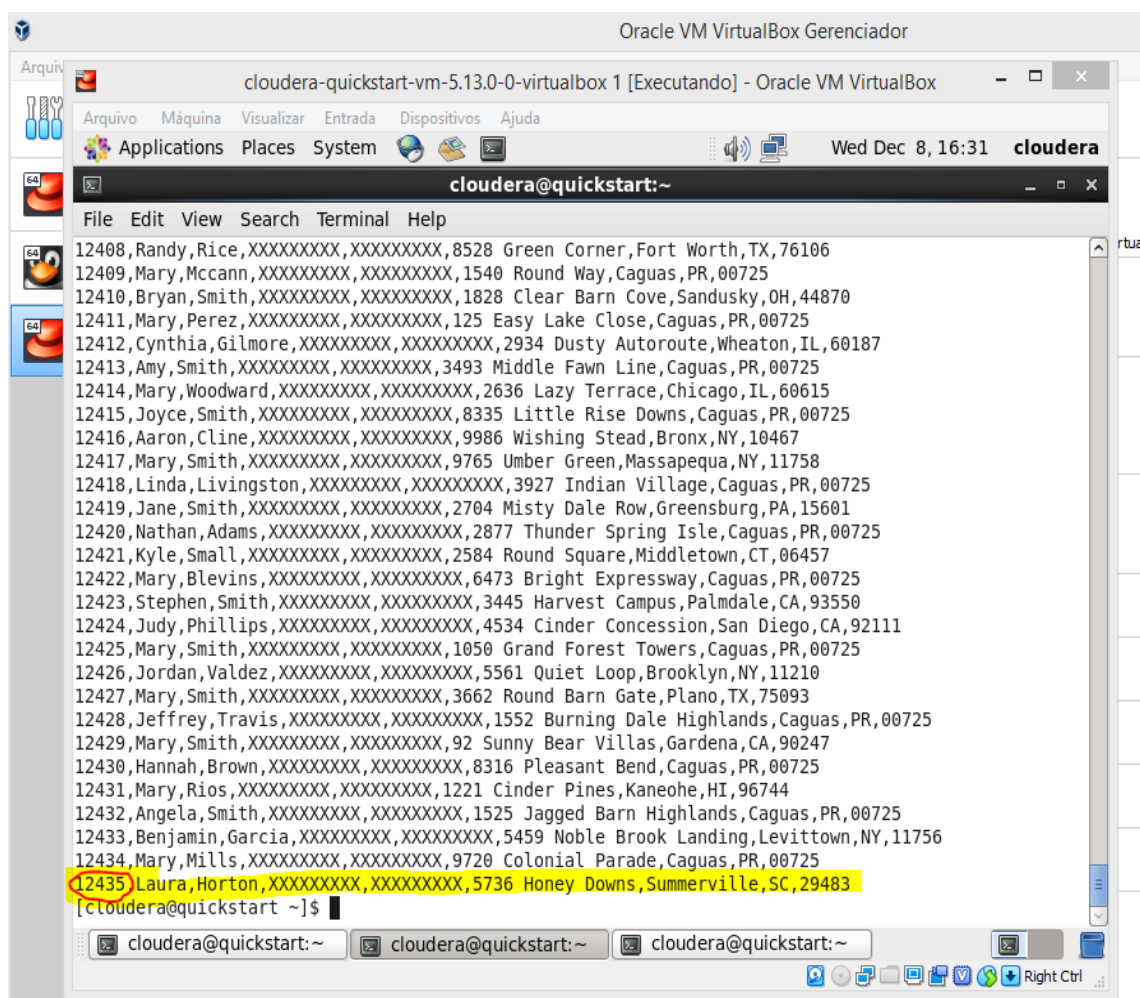
## PASSO 04 - agora vamos acessar esse diretório e listar os arquivos:

```
sudo hdfs dfs -ls /user/cloudera/customers;
```

```
[cloudera@quickstart ~]$ sudo hdfs dfs -ls /user/cloudera/customers;
Found 1 items
-rw-r--r-- 1 cloudera cloudera 953525 2021-12-08 16:22 /user/cloudera/customers/part-m-0000
0
[cloudera@quickstart ~]$
```

## PASSO- 05 vamos visualizar o conteúdo do arquivo “part-m-00000”

```
sudo hdfs dfs -cat /user/cloudera/customers/part-m-00000;
```



Podemos observar que a tabela **CUSTOMERS** foi importada com sucesso e que o SQOOP está funcionando perfeitamente.

## PASSO-06 vamos apagar esses dados com o seguinte comando:

```
sudo hdfs dfs -rm -r /user/cloudera;
```

```
[cloudera@quickstart ~]$ sudo hdfs dfs -rm -r /user/cloudera;  
Deleted /user/cloudera  
[cloudera@quickstart ~]$
```

Pronto, nosso HDFS está novamente limpo.

**Agora é hora de encapsularmos nossa ingestão de dados com a criação de JOB.**

**Passo - 07 crie um JOB com o seguinte comando:**

```
sqoop job -D sqoop.metastore.client.record.password=true --create  
batchlayer -- import --connect jdbc:mysql://localhost/retail_db --table  
customers --username root --password cloudera --check-column  
customer_id --incremental append --last-value 0 -m 1
```

**PASSO – 08 Verifique os JOBS existentes:**

*Digite o seguinte: `sqoop job --list;`*

```
[cloudera@quickstart ~]$ sqoop job --list;  
Warning: /usr/lib/sqoop/./accumulo does not exist! Accumulo imports will fail.  
Please set $ACCUMULO_HOME to the root of your Accumulo installation.  
21/12/08 16:38:35 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.13.0  
Available jobs:  
batchlayer  
[cloudera@quickstart ~]$
```

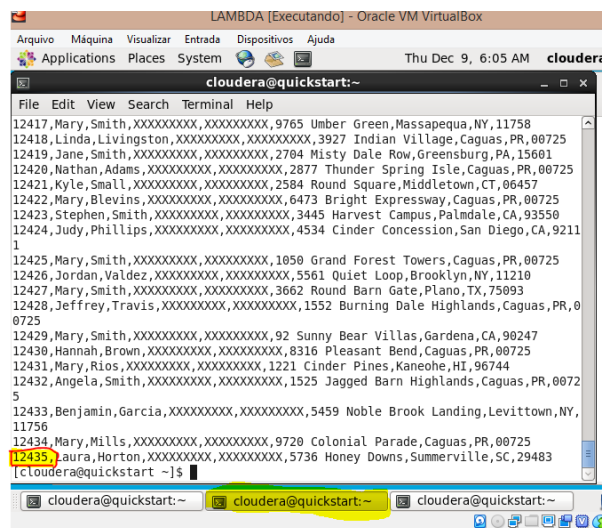
Pronto nosso JOB foi criado com o nome de batchlayer;

**PASSO – 09 Agora vamos executar o JOB:**

```
sqoop job --exec batchlayer;
```

**PASSO - 10 Verifique novamente os dados:**

```
sudo hdfs dfs -cat /user/cloudera/customers/part-m-00000;
```



```
LAMBDA [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
Applications  Places  System
Thu Dec 9, 6:05 AM  cloudera

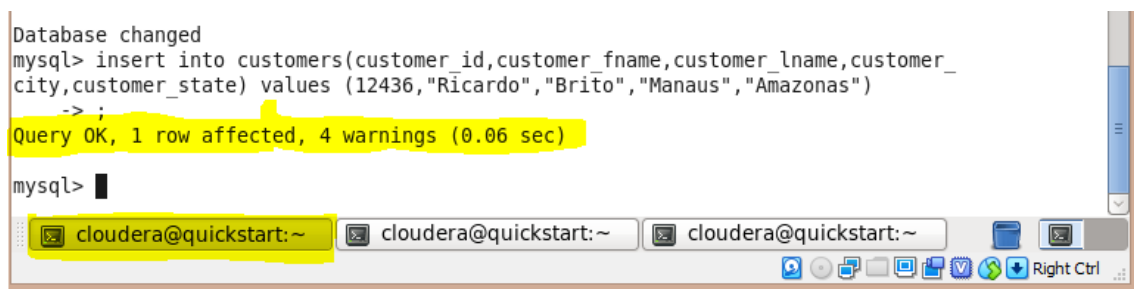
cloudera@quickstart:~
File Edit View Search Terminal Help
12417, Mary, Smith, XXXXXXXX, XXXXXXXX, 9765 Umber Green, Massapequa, NY, 11758
12418, Linda, Livingston, XXXXXXXX, XXXXXXXX, 3927 Indian Village, Caguas, PR, 00725
12419, Jane, Smith, XXXXXXXX, XXXXXXXX, 2794 Misty Dale Row, Greensburg, PA, 15601
12420, Nathan, Adams, XXXXXXXX, XXXXXXXX, 2877 Thunder Spring Isle, Caguas, PR, 00725
12421, Kyle, Small, XXXXXXXX, XXXXXXXX, 2584 Round Square, Middletown, CT, 06457
12422, Mary, Blevins, XXXXXXXX, XXXXXXXX, 6473 Bright Expressway, Caguas, PR, 00725
12423, Stephen, Smith, XXXXXXXX, XXXXXXXX, 3445 Harvest Campus, Palmdale, CA, 93550
12424, Judy, Phillips, XXXXXXXX, XXXXXXXX, 4534 Cinder Concession, San Diego, CA, 92111
1
12425, Mary, Smith, XXXXXXXX, XXXXXXXX, 1050 Grand Forest Towers, Caguas, PR, 00725
12426, Jordan, Valdez, XXXXXXXX, XXXXXXXX, 5561 Quiet Loop, Brooklyn, NY, 11210
12427, Mary, Smith, XXXXXXXX, XXXXXXXX, 3662 Round Barn Gate, Plano, TX, 75093
12428, Jeffrey, Travis, XXXXXXXX, XXXXXXXX, 1552 Burning Dale Highlands, Caguas, PR, 00725
12429, Mary, Smith, XXXXXXXX, XXXXXXXX, 92 Sunny Bear Villas, Gardena, CA, 90247
12430, Hannah, Brown, XXXXXXXX, XXXXXXXX, 8316 Pleasant Bend, Caguas, PR, 00725
12431, Mary, Rios, XXXXXXXX, XXXXXXXX, 1221 Cinder Pines, Kaneohe, HI, 96744
12432, Angela, Smith, XXXXXXXX, XXXXXXXX, 1525 Jagged Barn Highlands, Caguas, PR, 00725
5
12433, Benjamin, Garcia, XXXXXXXX, XXXXXXXX, 5459 Noble Brook Landing, Levittown, NY, 11756
12434, Mary, Mills, XXXXXXXX, XXXXXXXX, 9720 Colonial Parade, Caguas, PR, 00725
12435, Laura, Horton, XXXXXXXX, XXXXXXXX, 5736 Honey Downs, Summerville, SC, 29483
cloudera@quickstart ~$
```

Note que a última chave inserida é a **12435**, se executarmos outra vez o JOB não acontecerá nada, mas, podemos ir ao MYSQL e inserimos um registro para verificarmos se de fato o nosso JOB está funcionando. **Vamos fazer esse teste? Só se for agora!**

**PASSO - 11** Inserindo um registro na tabela de CUSTOMERS.

Vá até ao terminal do MYSQL e digite:

```
insert into
customers(customer_id,customer_fname,customer_lname,customer_city,customer_
state) values (12436,"Ricardo","Brito","Manaus","Amazonas");
```



```
Database changed
mysql> insert into customers(customer_id,customer_fname,customer_lname,customer_
city,customer_state) values (12436,"Ricardo","Brito","Manaus","Amazonas")
-> ;
Query OK, 1 row affected, 4 warnings (0.06 sec)

mysql>
```

Um Registro foi inserido;

**PASSO - 12** Vamos executar novamente o nosso JOB batchlayer;

```
sqoop job --exec batchlayer;
```

**PASSO - 13** Vamos agora listar os arquivos:

```
sudo hdfs dfs -ls /user/cloudera/customers;
```



The screenshot shows a terminal window titled "LAMBDA [Executando] - Oracle VM VirtualBox". The terminal is running the command `sudo hdfs dfs -ls /user/cloudera/customers;`. The output shows two files: `part-m-00000` and `part-m-00001`. The file `part-m-00001` is highlighted with a red circle. The terminal also shows the command `clear` being executed.

Veja que agora tem dois arquivos ("`part-m-00000`" e "`part-m-00001`").

**PASSO – 14** Vamos visualizar o segundo arquivo: "`part-m-00001`" digite o comando a seguir:

```
sudo hdfs dfs -cat /user/cloudera/customers/part-m-00001;
```

The screenshot shows a terminal window titled "LAMBDA [Executando] - Oracle VM VirtualBox". The terminal is running the command `sudo hdfs dfs -cat /user/cloudera/customers/part-m-00001;`. The output shows the text: `12436,Ricardo,Brito,Manaus,Amazonas,`. The terminal also shows the command `clear` being executed.

Olha que fantástico, Veja que foi criado outro arquivo com apenas um registro que fora inserido no MYSQL: ("`12436,Ricardo, Brito.....`").

**Perfeito**, a primeira etapa do nosso projeto está pronta, fizemos um JOB no SQOOP que atualiza os dados no HADOOP HDFS.

Nos próximos dias estarei atualizando este  
arquivo com a segunda etapa

**Forte abraço e até lá!**