1. **Descritivo**

O LuizaLabs identificou a possiblidade de agregar valor durante o processo de venda presencial ou online através de logs de compra de usuários. Essas informações serão utilizadas tanto nas lojas na venda direta (durante um atendimento presencial) assim como pela equipe de marketing para ofertar produtos em função das últimas compras e tendência dos últimos meses de cada usuário. A proposta inicial é ser uma ferramenta que durante a venda pode trazer rápida informação para que sejam feitas boas sugestões de outros produtos a serem ofertados na mesma compra.

Será possível também verificar as compras por cidade para focar as campanhas de forma regional uma vez que a informação já encontra-se disponível.

1. **Arquitetura**

A informação de log fica armazenado em um MongoDB e faz parte da estrutura assíncrona do fluxo de venda (online ou presencial).

A arquitetura de destino é um Cassandra que apresenta um modelo de dados com foco nas perguntas que queremos responder.

O pipeline foi desenvolvido em Python e faz o input dos dados pertinentes a equipe de vendas e marketing através de arquivos.

No laboratório criado para executar a POC, usamos o .csv mas poderíamos fazer em outro formato com alterações simples.

O cassandra é uma solução pensada principalmente para otimização de escrita e torna-se uma boa candidata para muitos insert. Fora essa questão como o intuito é utilizar essa informação como uma importante ferramenta para aproximar o vendedor e o consumidor é importante ter essa solução com grande disponibilidade e facil crescimento.

1. **Projeto**

O projeto é composto pelo arquivos:

1. SetupCassandra.py

Processo criado para limpeza e recrição do modelo de dados do cassandra

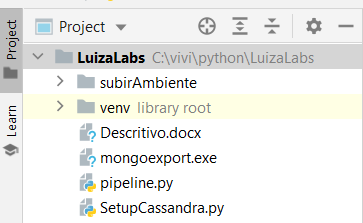
1. Pipeline.py

Executa a transferencia de dados da estrutura do mongo para o cassandra respeitado as necessidades do modelo do destino.

Ferramentas incorpordas ao processo:

1. Mongoexport.exe

Ferramenta utilizada para faciltar a geração da massa de dados para transferência



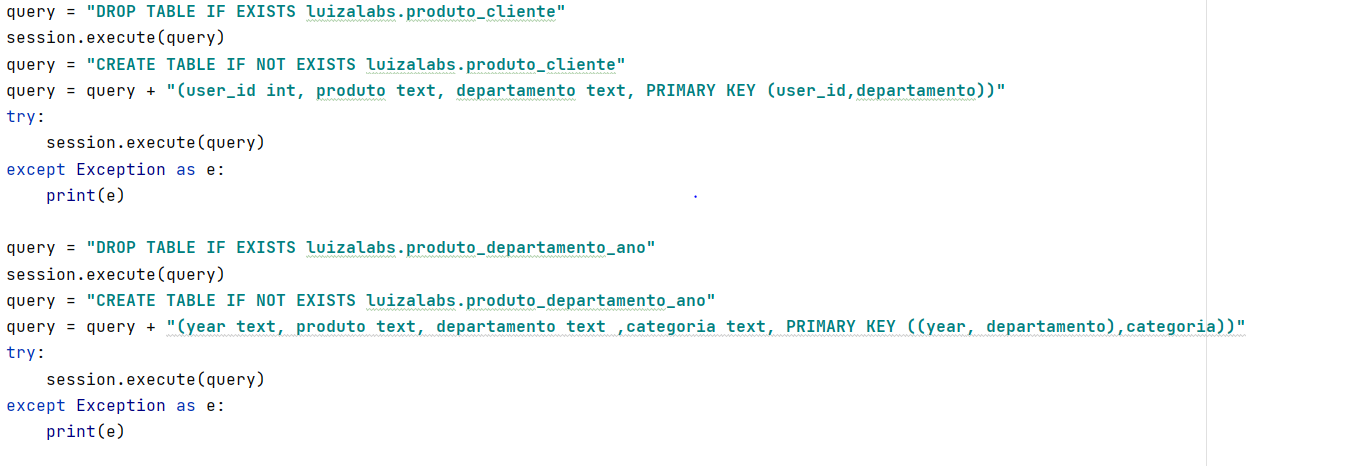
1. **Setup Cassandra.py detalhes técnicos**



Para a nossa POC criamos localmente o Cassandra sendo assim usamos “SimpleStrategy” e “1” para class e replicator\_factor respectivamente.

Como é um ambiente de DEV e não temos mais nodes o SimplesStrategy. Outra opção para esse paramtero é utilizarmos NetworkTopologyStrategy e nessa opção é possível dizer que as replicas estão em locais separados para evitar indisponibilidade.

Para o nosso fator de replicação é 1 mas o case de mercado para alta disponibilidade são 3 nodes. Com o fator de replicação igual a 1 significa que temos apenas q copia de cada linha no cluster. No caso de 2 ou mais estamos considerando um cenário em que cada linha tem copia em nodes diferentes.



Um conceito fundamental quando desejamos utilizar o Cassandra é entendermos o que queremos responder porque o modelo de dados já precisa ser criado em função de quais questões precisamos resolver.

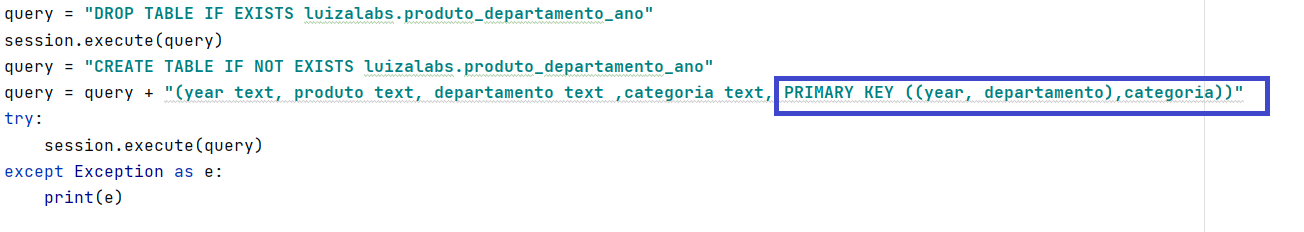
E com isso podemos ter dois ou mais objetos com os mesmos dados. Não tem foco em normalização.

Para a POC tínhamos duas questões que viraram dois objetos:

1. Quais foram os produtos comprados por um determinado cliente ordenado por departamento

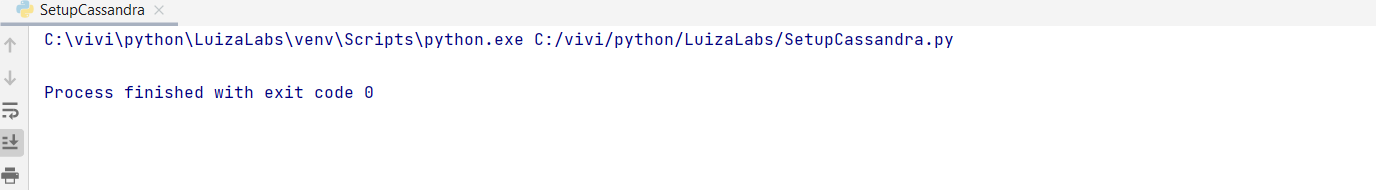


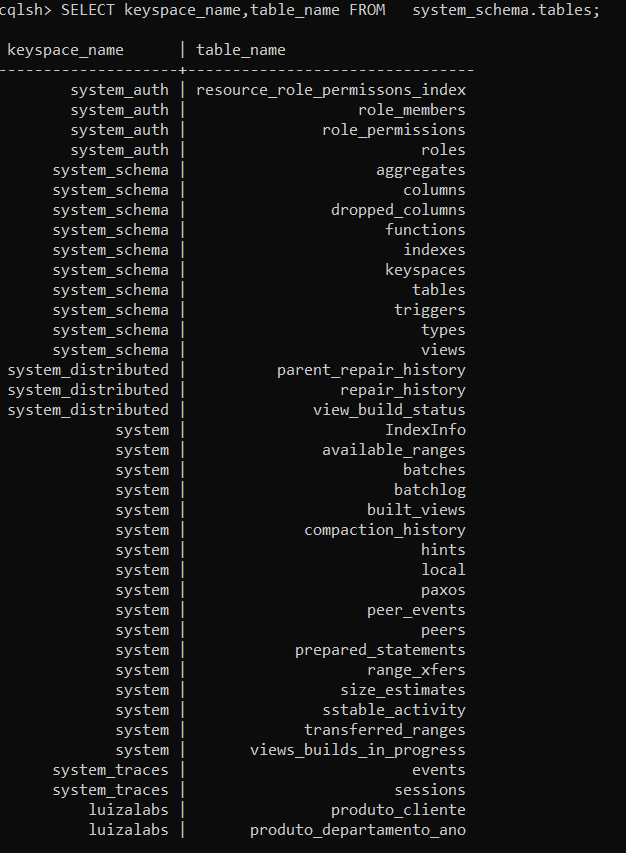
1. Quais foram os produtos comprados em um determinado ano (2021 por exemplo) em um departamento (de livros por exemplo) ordenado por categoria



Nessa etapa tem uma abordagem conceitual importante para ocnseguirmos atender o solicitado que tem relação com o conceito de primary key, partition key e clustering columns. A primary key é composta por partition key e clustering key sendo o primeiro elemento da primary key a partition e o segundo o clustering (sort).

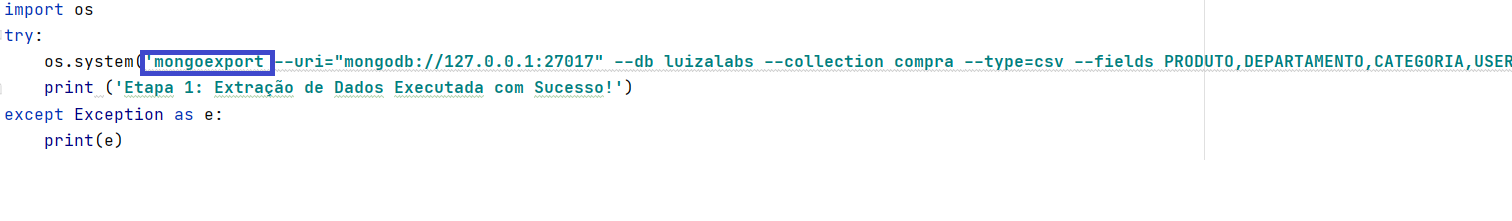
Resultado da execução:





1. **Execução do pipeline.py**

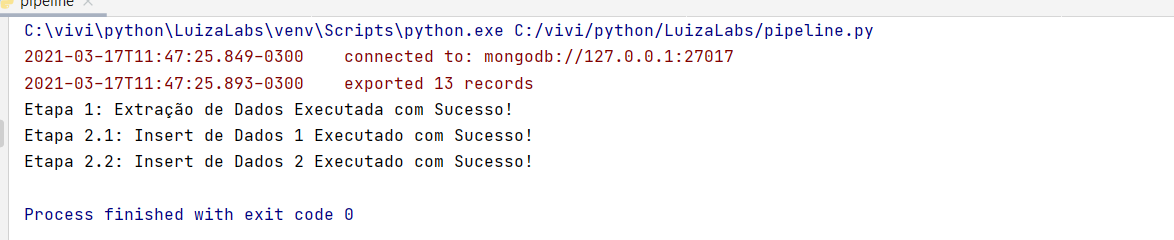
Uma forma eficiente de exportar os dados do mongo é utilizar as própria tools que o mongo oferece e por isso para a POC a abordagem foi seguir na solução oferecida pelo produto

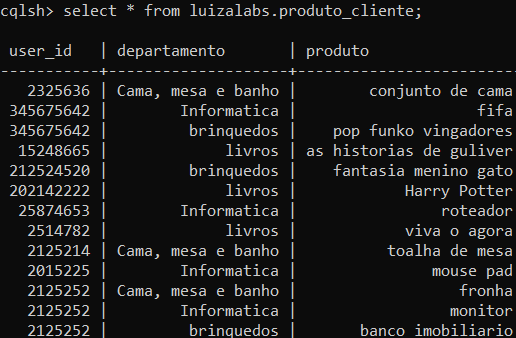
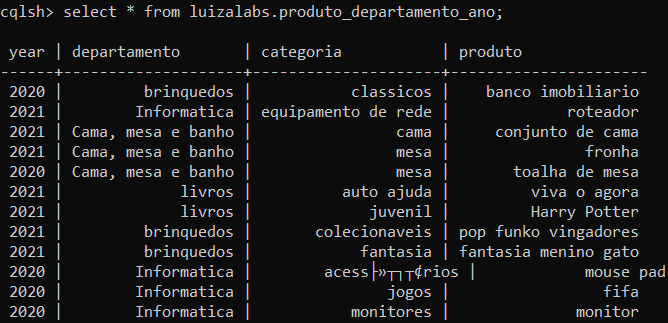


Na sequência já temos o import da informação para o Cassandra seguindo o modelo de dados criado.



Resultado da execução:



1. **Conclusão do Laboratório**

Esse processo completo pode ser executado quantas vezes forem necessárias

1. **Alteração para ambientes com infra mais robusta**

Criar um cluster cassandra permitindo a leitura e escrita em múltiplos nodes.

No caso do mongo poderíamos ter um cluster mongo para garantir HA.