

# Pós-graduação

## Engenharia de Dados / Big Data

Aluno: Charles de Araujo Melo

Infnet - Trabalho Infraestrutura Cassandra

**1.** Explique, com suas palavras, as diferenças entre bases de dados SQL e NoSQL, apresentando exemplos de uso para cada uma delas.

**R.** O grande diferencial dos bancos de dados NoSQL é que toda a informação é agrupada e guardada no mesmo registro. Já no SQL você precisa ter o relacionamento entre várias tabelas para ter a informação, informação esta disposta no modelo entidade e relacionamento.

As bases de dados SQL utilizam uma abordagem de escala vertical, o que significa que escalam adicionando mais poder ao servidor. As bases de dados NoSQL utilizam uma abordagem de escala horizontal, o que significa que são escaladas adicionando mais servidores.

O NoSQL é mais indicado para aqueles sistemas que tenham necessidades maiores de armazenamento e desempenho com dados não estruturados. Já o SQL é indicado quando necessitamos de maior consistência.

**2.** Escolha uma base de dados pública brasileira, que será o alvo para condução de seu Projeto de Disciplina. Esta base será carregada para uma infraestrutura Cassandra, para dar suporte à geração de um processo de análise de dados.

**R.** Base de dados escolhida : [food-preference](https://www.kaggle.com/naufaal98/food-preference)

<https://www.kaggle.com/naufaal98/food-preference>

**3.** Defina pelo menos uma pergunta de negócio que será o objetivo de seu processo de análise a ser desenvolvido a partir dos dados armazenados na infraestrutura Cassandra.

**R.** Todos Clientes com idade maior que 30 anos e que comeram sobremesa.

**R.** Média da idade de todos os clientes do gênero masculino e que não comeram sobremesa.

**1.1** Apresente um planejamento básico da infraestrutura necessária para comportar os dados que deseja importar.

**1.2** Apresente a descrição das configurações feitas em um ambiente computacional para receber a infraestrutura Cassandra. Ao final deste passo, a máquina que hospedará o ambiente deverá estar pronta para receber o conjunto de dados escolhido para o trabalho.

**1.3** Faça a carga de dados para o ambiente configurado e execute todas as atividades necessárias na base para que a infraestrutura fique pronta para a execução de operações com os dados. Apresente uma comprovação de que a carga de dados tenha sido corretamente executada.

Usarei o Docker para criar um container com a imagem do cassandra.

Instalação e configuração do Cassandra (via container Docker):

<https://www.docker.com/products/docker-desktop/>

Baixa a imagem do Cassandra no Docker Hub para a máquina local

1.1 Baixar a imagem:

`docker pull cassandra`

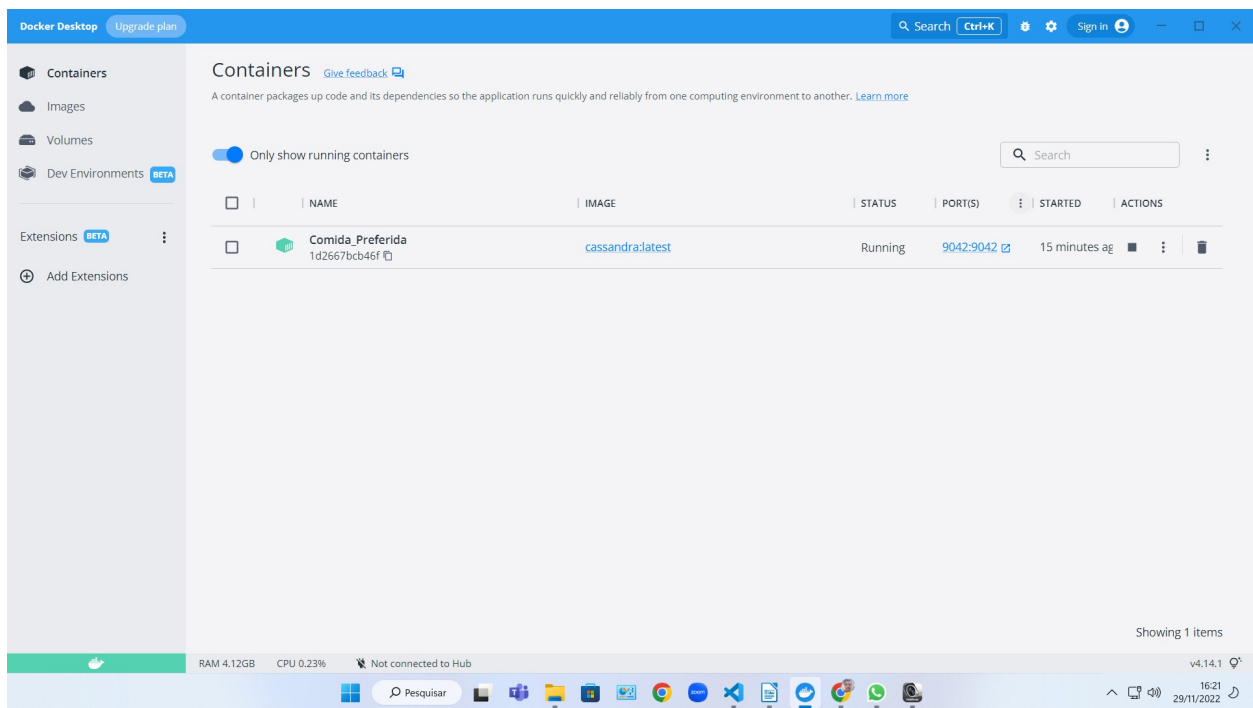
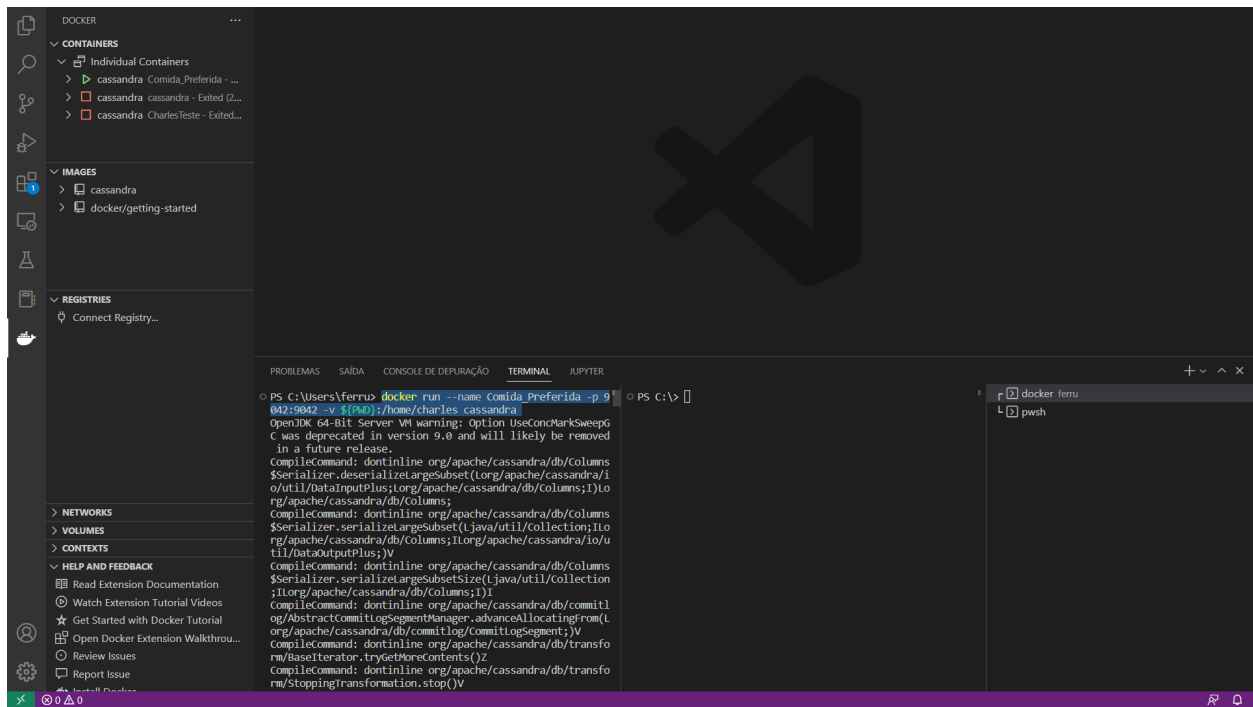
1.2 Visualizar as imagens existentes na máquina

`docker image ls`

Utilizando o terminal do Visual Studio Code para criar meu container com espelhamento em meu diretório local para por meus arquivos para serem enxergados dentro do container.

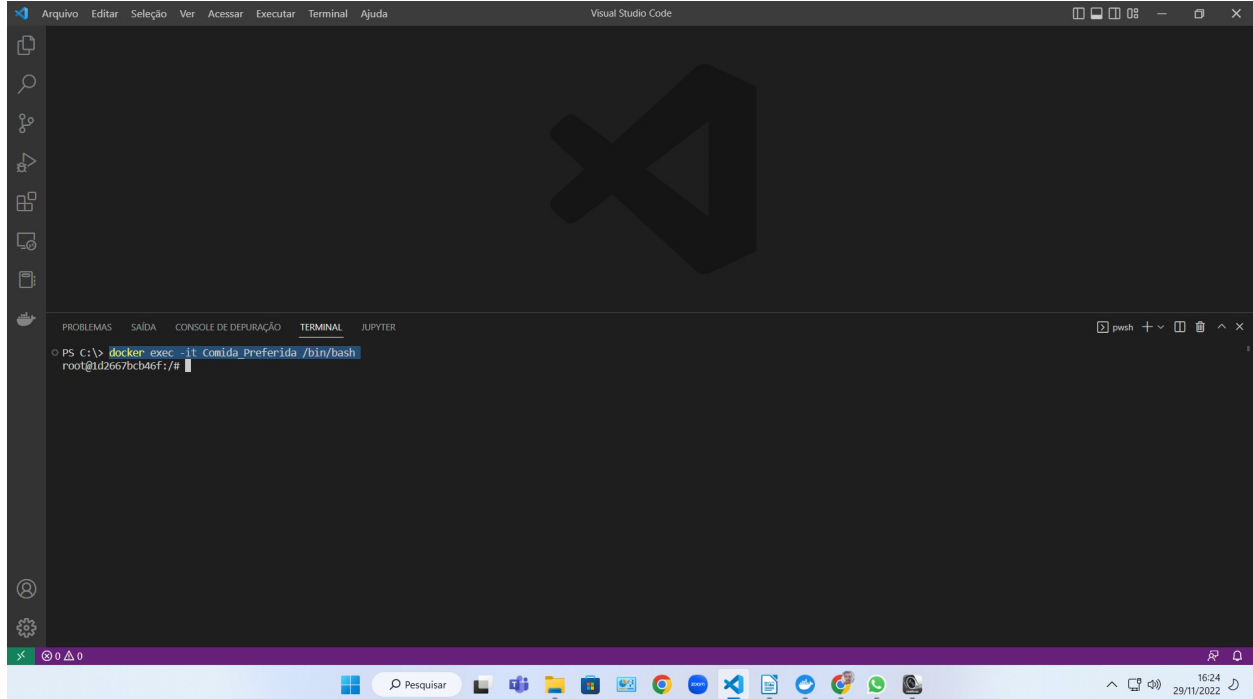
Utilizei o seguinte comando:

`docker run --name Comida_Preferida -p 9042:9042 -v ${PWD}:/home/charles cassandra`



Iniciando meu container através de um terminal do ( Visual Studio Code ).

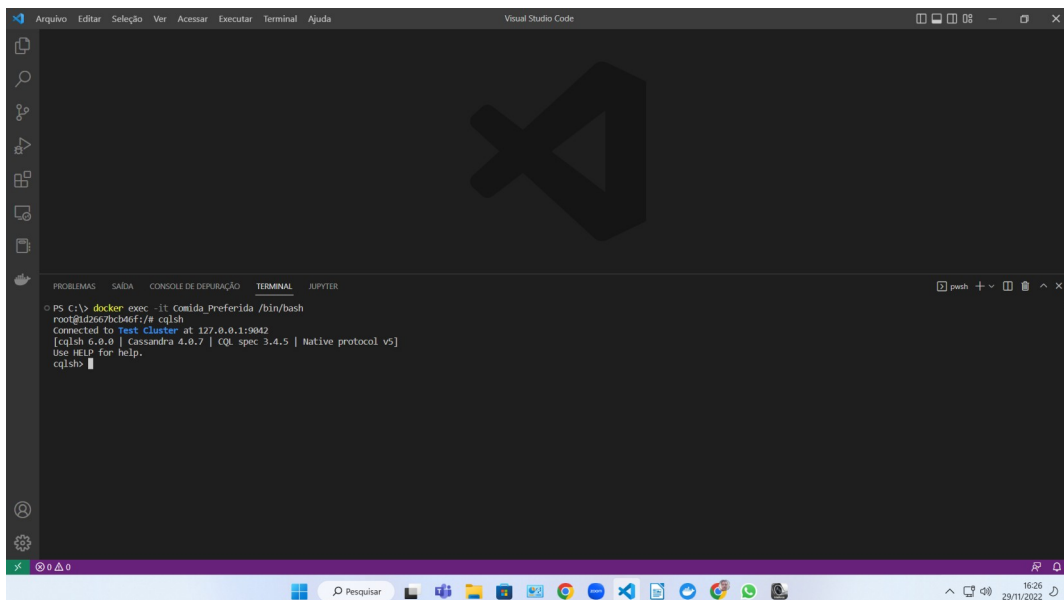
`docker exec -it Comida_Preferida /bin/bash`



```
PS C:\> docker exec -it Comida_Preferida /bin/bash
root@d2667bc46f:/#
```

Linguagem de consulta Cassandra(CQL) é a principal linguagem para se comunicar com o Apache Cassandra.

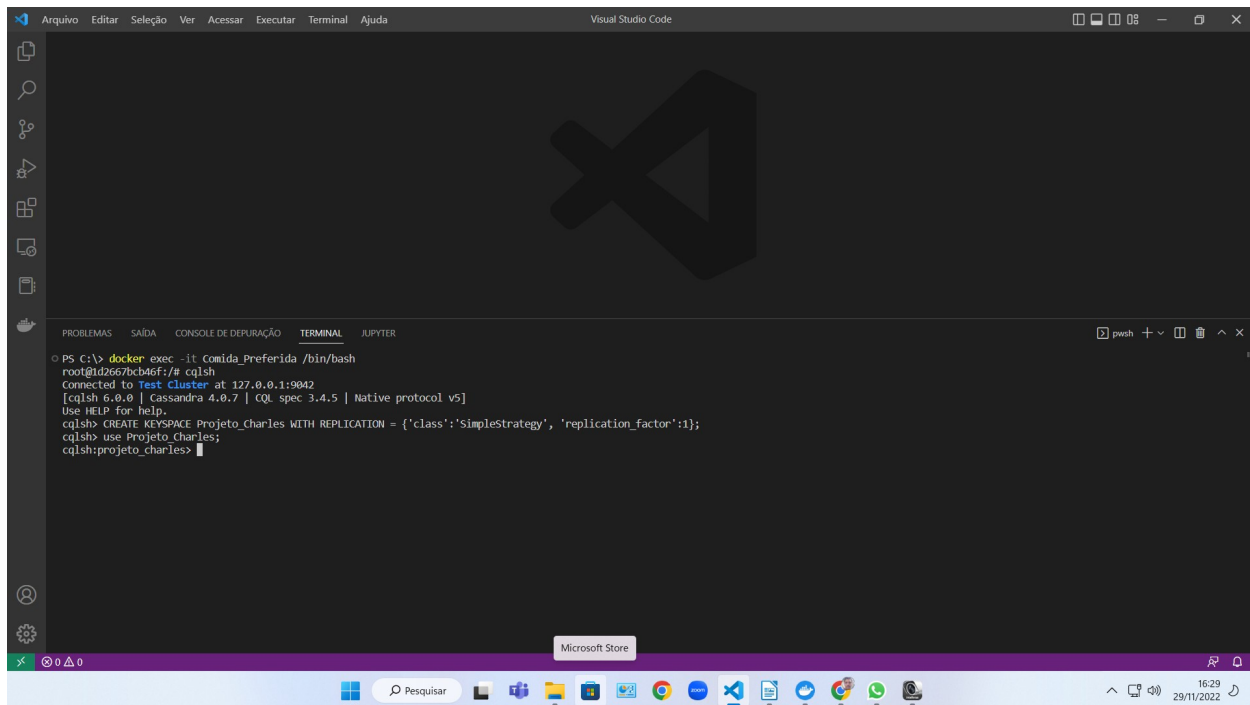
Comando: `CQLSH`



```
PS C:\> docker exec -it Comida_Preferida /bin/bash
root@d2667bc46f:/# cqlsh
Connected to test Cluster at 127.0.0.1:9042
[cqlsh 6.0.0 | Cassandra 4.0.7 | CQL spec 3.4.5 | Native protocol v5]
Use HELP for help.
cqlsh>
```

Criando a Keyspace e acessando:

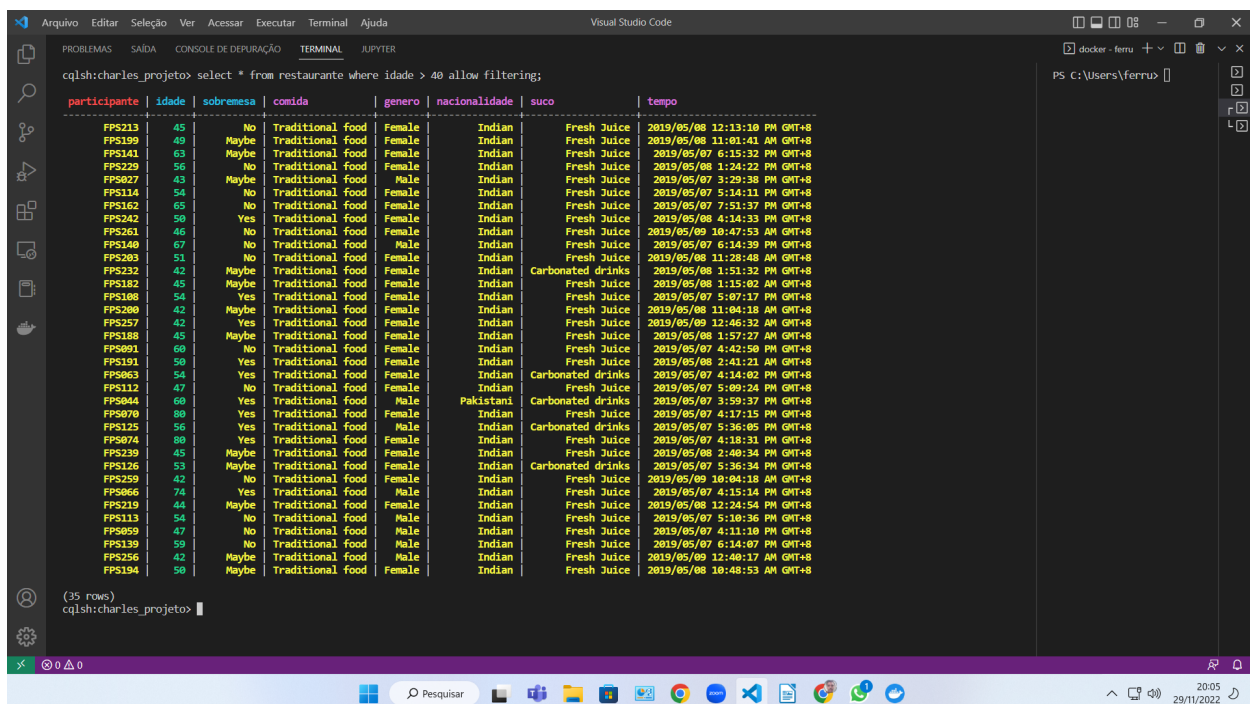
```
CREATE KEYSPACE Projeto_Charles WITH REPLICATION = {'class':'SimpleStrategy', 'replication_factor':1};
```



The screenshot shows a Visual Studio Code window with a terminal. The terminal output is as follows:

```
PS C:\> docker exec -it Comida_Preferida /bin/bash
root@d2667bcb46f:/# cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.0.0 | Cassandra 4.0.7 | CQL spec 3.4.5 | Native protocol v5]
Use HELP for help.
cqlsh> CREATE KEYSPACE Projeto_Charles WITH REPLICATION = {'class':'SimpleStrategy', 'replication_factor':1};
cqlsh> use Projeto_Charles;
cqlsh:projeto_charles>
```

**1.4** Apresente o código e uma comprovação de sucesso de execução de uma consulta de leitura na base de dados carregada para a infraestrutura Cassandra.



The screenshot shows a Visual Studio Code window with a terminal. The terminal output is as follows:

```
cqlsh:charles_projeto> select * from restaurante where idade > 40 allow filtering;
```

participante	idade	sobremesa	comida	genero	nacionalidade	suco	tempo
FPS213	45	No	Traditional food	Female	Indian	Fresh Juice	2019/05/08 12:13:18 PM GMT+8
FPS199	49	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 11:01:41 AM GMT+8
FPS141	63	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/07 6:15:32 PM GMT+8
FPS229	56	No	Traditional food	Female	Indian	Fresh Juice	2019/05/08 1:24:22 PM GMT+8
FPS027	43	Maybe	Traditional food	Male	Indian	Fresh Juice	2019/05/07 3:29:38 PM GMT+8
FPS114	54	No	Traditional food	Female	Indian	Fresh Juice	2019/05/07 5:14:11 PM GMT+8
FPS162	65	No	Traditional food	Female	Indian	Fresh Juice	2019/05/07 7:51:37 PM GMT+8
FPS242	59	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 4:14:33 PM GMT+8
FPS261	46	No	Traditional food	Female	Indian	Fresh Juice	2019/05/09 10:47:53 AM GMT+8
FPS140	67	No	Traditional food	Male	Indian	Fresh Juice	2019/05/07 6:14:39 PM GMT+8
FPS203	51	No	Traditional food	Female	Indian	Fresh Juice	2019/05/08 11:23:48 AM GMT+8
FPS232	42	Maybe	Traditional food	Female	Indian	Carbonated drinks	2019/05/08 1:51:32 PM GMT+8
FPS182	45	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 1:15:02 AM GMT+8
FPS108	54	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/07 5:07:17 PM GMT+8
FPS200	42	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 11:04:18 AM GMT+8
FPS257	42	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/09 12:46:32 AM GMT+8
FPS198	45	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 1:57:27 AM GMT+8
FPS091	60	No	Traditional food	Female	Indian	Fresh Juice	2019/05/07 4:42:58 PM GMT+8
FPS191	50	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 2:41:21 AM GMT+8
FPS063	54	Yes	Traditional food	Female	Indian	Carbonated drinks	2019/05/07 4:14:02 PM GMT+8
FPS112	47	No	Traditional food	Female	Indian	Fresh Juice	2019/05/07 5:09:24 PM GMT+8
FPS044	60	Yes	Traditional food	Male	Pakistani	Carbonated drinks	2019/05/07 3:59:37 PM GMT+8
FPS070	80	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/07 4:17:15 PM GMT+8
FPS125	56	Yes	Traditional food	Male	Indian	Carbonated drinks	2019/05/07 5:36:05 PM GMT+8
FPS074	80	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/07 4:18:31 PM GMT+8
FPS239	45	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 2:40:34 PM GMT+8
FPS126	53	Maybe	Traditional food	Female	Indian	Carbonated drinks	2019/05/07 5:36:34 PM GMT+8
FPS259	42	No	Traditional food	Female	Indian	Fresh Juice	2019/05/09 10:04:18 AM GMT+8
FPS066	74	Yes	Traditional food	Male	Indian	Fresh Juice	2019/05/07 4:15:14 PM GMT+8
FPS219	44	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 12:24:54 PM GMT+8
FPS113	54	No	Traditional food	Male	Indian	Fresh Juice	2019/05/07 5:10:36 PM GMT+8
FPS059	47	No	Traditional food	Male	Indian	Fresh Juice	2019/05/07 4:11:18 PM GMT+8
FPS139	59	No	Traditional food	Male	Indian	Fresh Juice	2019/05/07 6:14:07 PM GMT+8
FPS256	42	Maybe	Traditional food	Male	Indian	Fresh Juice	2019/05/09 12:40:17 AM GMT+8
FPS194	50	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 10:48:53 AM GMT+8

(35 rows)  
cqlsh:charles\_projeto>

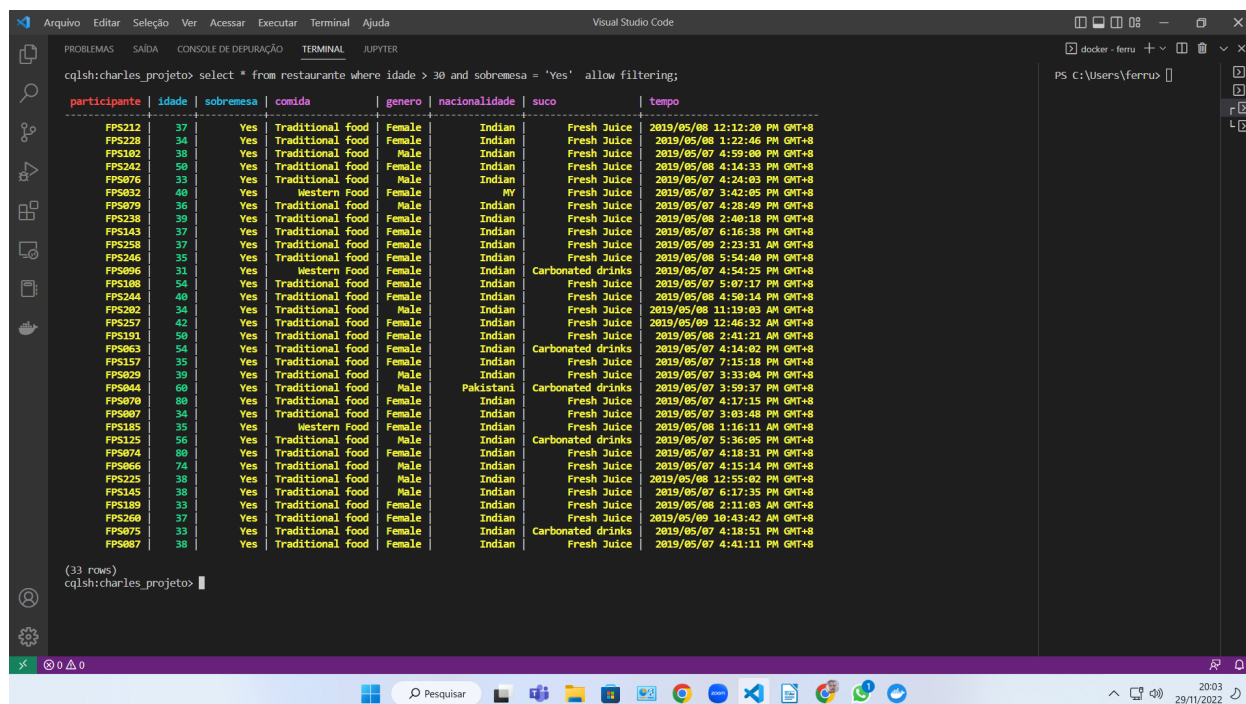
**1.5** Utilizando linguagem Python, extraia o conjunto de informações necessário para dar prosseguimento ao processo de resposta à(s) pergunta(s) de negócio estabelecida(s) na questão 3. Apresente o código criado para a extração e uma comprovação de sucesso da operação.

**1.6** Execute quaisquer operações de manipulação, ajuste e criação de variáveis necessárias para dar apoio à resposta do problema de negócio em estudo. Apresente códigos em Python de pelo menos uma operação de manipulação de dados e uma operação de criação de novas variáveis.

**R. 1.1** Todos clientes com idade maior que 30 anos e que comeram sobremesa.

**R. 1.2** Média da idade de todos os clientes do gênero masculino e que não comeram sobremesa.

1.1

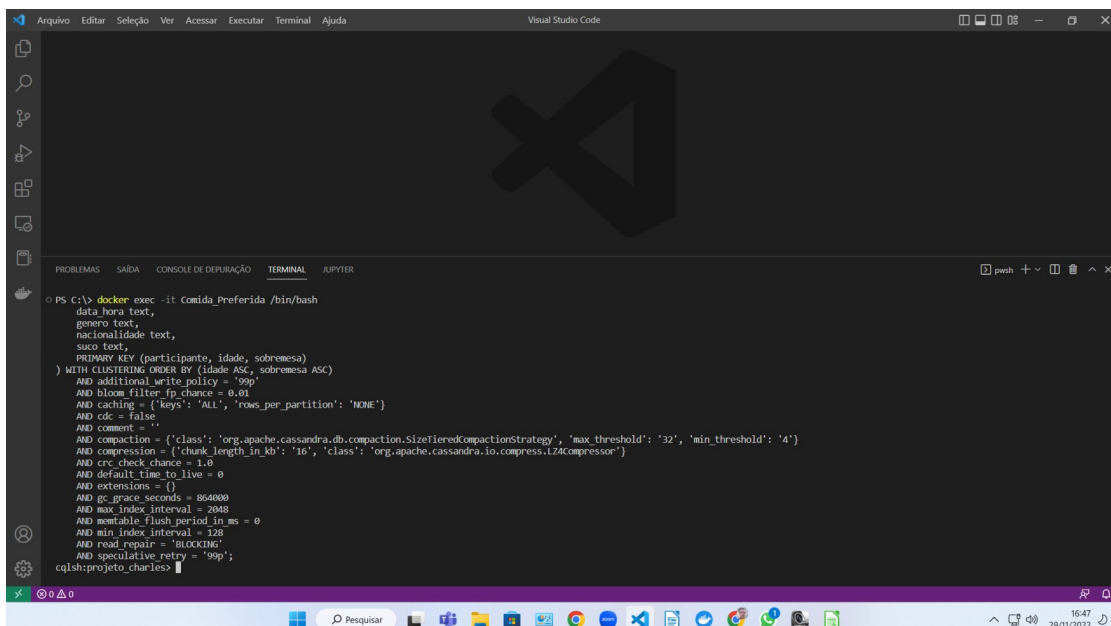


```
cqlsh:charles_projeto> select * from restaurante where idade > 30 and sobremesa = 'Yes' allow filtering;
```

participante	idade	sobremesa	comida	genero	nacionalidade	suco	tempo
FPS212	37	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 12:12:20 PM GMT+8
FPS228	34	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 1:22:46 PM GMT+8
FPS102	38	Yes	Traditional food	Male	Indian	Fresh Juice	2019/05/07 4:59:00 PM GMT+8
FPS242	50	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 4:14:33 PM GMT+8
FPS076	33	Yes	Traditional food	Male	Indian	Fresh Juice	2019/05/07 4:24:03 PM GMT+8
FPS032	40	Yes	Western Food	Female	MY	Fresh Juice	2019/05/07 3:42:05 PM GMT+8
FPS079	36	Yes	Traditional food	Male	Indian	Fresh Juice	2019/05/07 4:28:49 PM GMT+8
FPS238	39	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 2:40:18 PM GMT+8
FPS143	37	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/07 6:16:38 PM GMT+8
FPS238	37	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/09 2:23:31 AM GMT+8
FPS246	35	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 5:54:40 PM GMT+8
FPS096	31	Yes	Western Food	Female	Indian	Carbonated drinks	2019/05/07 4:54:25 PM GMT+8
FPS108	54	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/07 5:07:17 PM GMT+8
FPS244	40	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 4:50:14 PM GMT+8
FPS202	34	Yes	Traditional food	Male	Indian	Fresh Juice	2019/05/08 11:19:03 AM GMT+8
FPS257	42	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/09 12:46:32 AM GMT+8
FPS191	50	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 2:41:21 AM GMT+8
FPS063	54	Yes	Traditional food	Female	Indian	Carbonated drinks	2019/05/07 4:14:02 PM GMT+8
FPS157	35	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/07 7:15:18 PM GMT+8
FPS029	39	Yes	Traditional food	Male	Indian	Fresh Juice	2019/05/07 3:33:04 PM GMT+8
FPS044	60	Yes	Traditional food	Male	Pakistani	Carbonated drinks	2019/05/07 3:59:37 PM GMT+8
FPS070	80	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/07 4:17:15 PM GMT+8
FPS007	34	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/07 3:03:48 PM GMT+8
FPS185	35	Yes	Western Food	Female	Indian	Fresh Juice	2019/05/08 1:16:11 AM GMT+8
FPS125	56	Yes	Traditional food	Male	Indian	Carbonated drinks	2019/05/07 5:36:05 PM GMT+8
FPS074	80	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/07 4:18:31 PM GMT+8
FPS066	74	Yes	Traditional food	Male	Indian	Fresh Juice	2019/05/07 4:15:14 PM GMT+8
FPS225	38	Yes	Traditional food	Male	Indian	Fresh Juice	2019/05/08 12:55:02 PM GMT+8
FPS145	38	Yes	Traditional food	Male	Indian	Fresh Juice	2019/05/08 6:17:35 PM GMT+8
FPS189	33	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 2:11:03 AM GMT+8
FPS200	37	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/09 10:43:42 AM GMT+8
FPS075	33	Yes	Traditional food	Female	Indian	Carbonated drinks	2019/05/07 4:18:51 PM GMT+8
FPS087	38	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/07 4:41:11 PM GMT+8

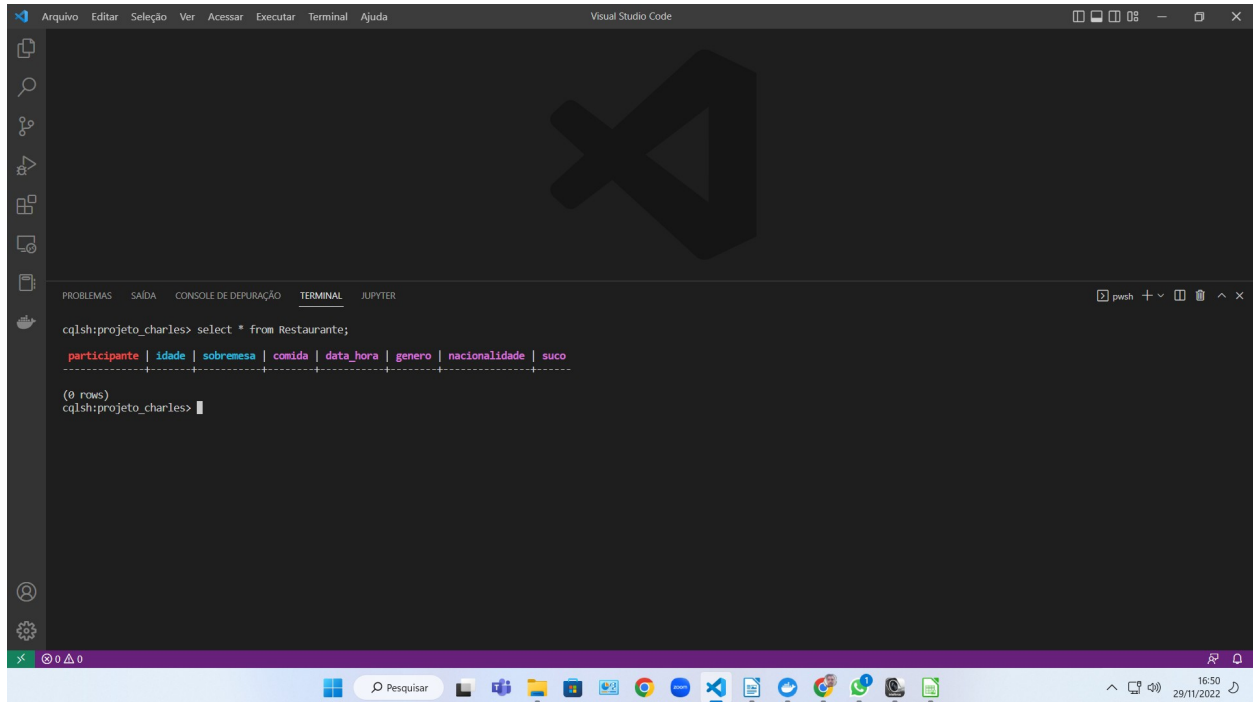
(33 rows)  
cqlsh:charles\_projeto>

1.2



```
PS C:\> docker exec -it Comida-Preferida /bin/bash
data.hora text,
genero text,
nacionalidade text,
suco text,
PRIMARY KEY (participante, idade, sobremesa)
) WITH CLUSTERING ORDER BY (idade ASC, sobremesa ASC)
AND additional_write_policy = '99p'
AND bloom_filter_fp_chance = 0.01
AND caching = {'keys': 'ALL', 'rows_per_partition': 'NONE'}
AND cdc = false
AND comment = ''
AND compaction = {'class': 'org.apache.cassandra.db.compaction.SizeTieredCompactionStrategy', 'max_threshold': '32', 'min_threshold': '4'}
AND compression = {'chunk_length_in_kb': '16', 'class': 'org.apache.cassandra.io.compress.LZ4Compressor'}
AND crc_check_chance = 1.0
AND default_time_to_live = 0
AND extensions = {}
AND gc_grace_seconds = 864000
AND max_index_interval = 2048
AND memtable_flush_period_in_ms = 0
AND min_index_interval = 128
AND read_repair = 'BLOCKING'
AND speculative_retry = '99p';
cqlsh:projeto_charles>
```

Select na tabela criada:



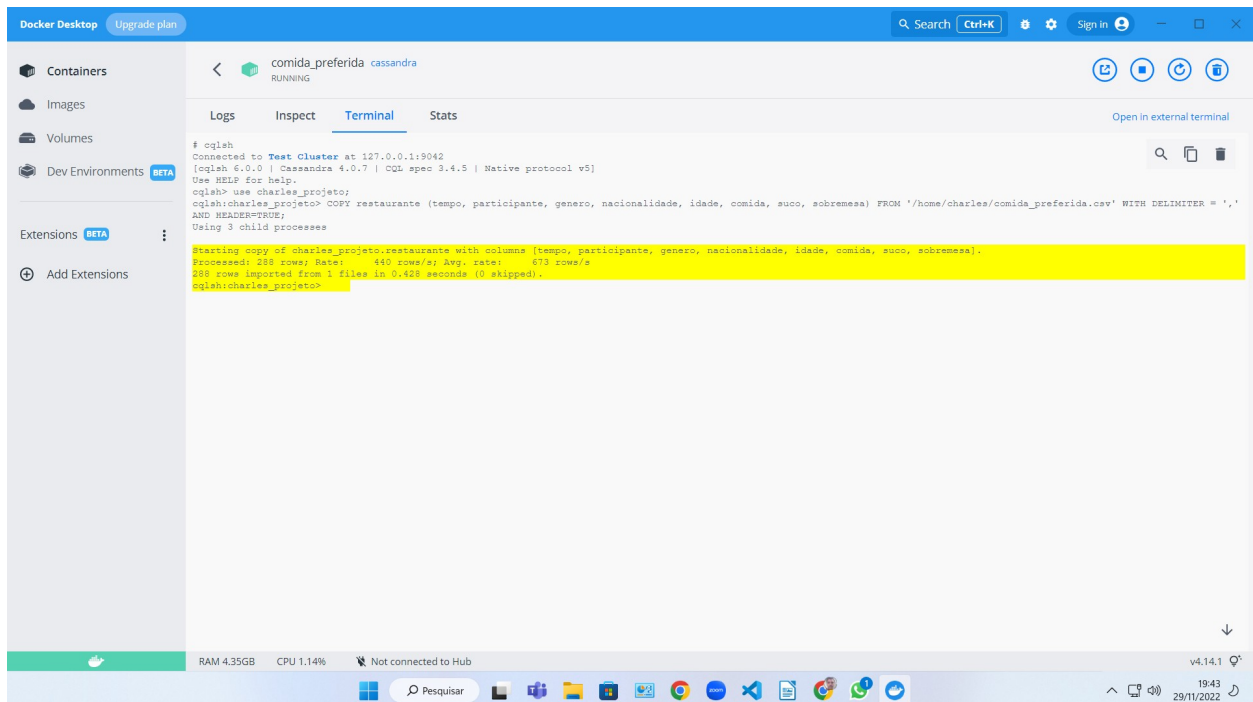
The screenshot shows the Visual Studio Code interface with a terminal window open. The terminal is running a CQL query in a shell named 'cqlsh:projeto\_charles'. The query is 'select \* from Restaurante;'. The output shows the table schema with columns: participante, idade, sobremesa, comida, data\_hora, genero, nacionalidade, and suco. Below the schema, it indicates '(0 rows)'.

```
cqlsh:projeto_charles> select * from Restaurante;

participante | idade | sobremesa | comida | data_hora | genero | nacionalidade | suco
-----
(0 rows)

cqlsh:projeto_charles>
```

Importando os dados do arquivo comida\_preferida.csv para minha tabela restaurante:



The screenshot shows the Docker Desktop interface with a terminal window open for a container named 'comida\_preferida'. The terminal shows the execution of a CQL query to import data from a CSV file into a table named 'restaurante'. The output indicates that 288 rows were imported successfully.

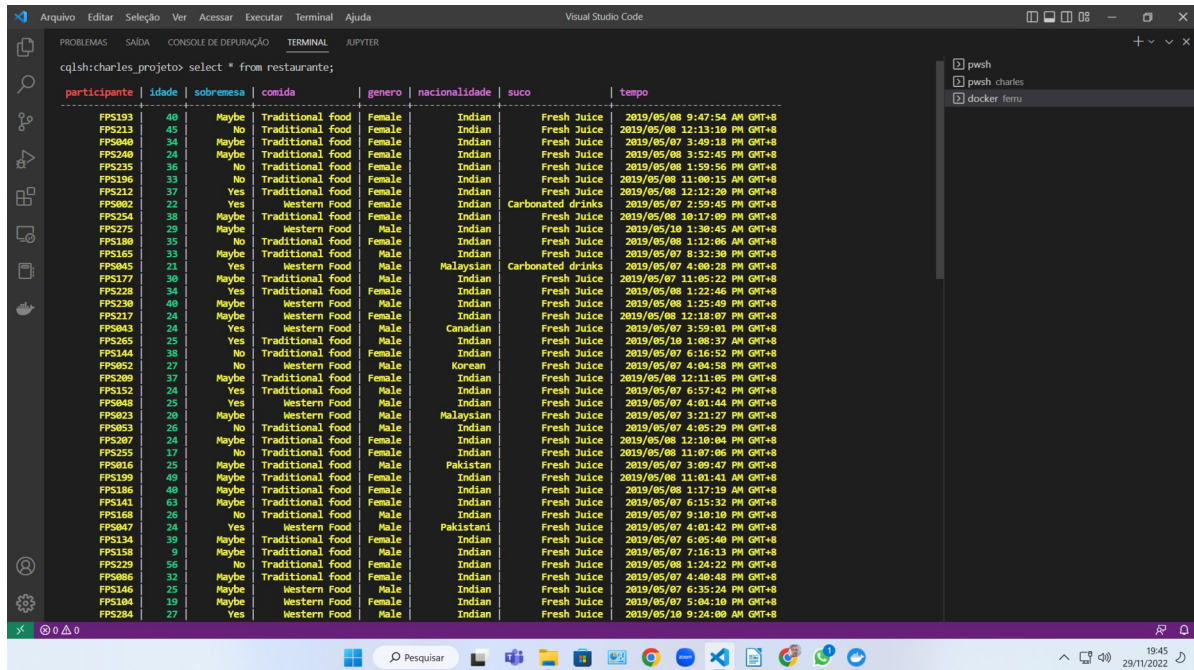
```
# cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.0.0 | Cassandra 4.0.7 | CQL spec 3.4.5 | Native protocol v5]
Use HELP for help.
cqlsh> use charles_projeto;
cqlsh:charles_projeto> COPY restaurante (tempo, participante, genero, nacionalidade, idade, comida, suco, sobremesa) FROM '/home/charles/comida_preferida.csv' WITH DELIMITER = ','
AND HEADER=TRUE;
Using 3 child processes

Starting copy of charles_projeto.restaurante with columns [tempo, participante, genero, nacionalidade, idade, comida, suco, sobremesa].
Processed: 288 rows; Rate: 240 rows/s; Avg. rate: 673 rows/s
288 rows imported from 1 files in 0.428 seconds (0 skipped).
cqlsh:charles_projeto>
```

select \* from restaurante where idade > 30 and sobremesa = 'Yes' allow filtering;

**1.6** Exporte a base tratada final para formato CSV. Apresente o código em Python que demonstre a execução desta operação.

COPY restaurante (tempo, participante, genero, nacionalidade, idade, comida, suco, sobremesa) FROM '/home/charles/comida\_preferida.csv' WITH DELIMITER = ',' AND HEADER=TRUE;



participante	idade	sobremesa	comida	genero	nacionalidade	suco	tempo
FPS193	40	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 9:47:54 AM GMT+8
FPS213	45	No	Traditional food	Female	Indian	Fresh Juice	2019/05/08 12:13:10 PM GMT+8
FPS040	34	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/07 3:40:18 PM GMT+8
FPS240	24	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 3:52:45 PM GMT+8
FPS235	36	No	Traditional food	Female	Indian	Fresh Juice	2019/05/08 1:59:56 PM GMT+8
FPS196	33	No	Traditional food	Female	Indian	Fresh Juice	2019/05/08 11:00:15 AM GMT+8
FPS212	37	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 12:12:20 PM GMT+8
FPS002	22	Yes	Western food	Female	Indian	Carbonated drinks	2019/05/07 2:59:45 PM GMT+8
FPS254	38	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 10:17:09 PM GMT+8
FPS275	29	Maybe	Western food	Male	Indian	Fresh Juice	2019/05/10 1:30:45 AM GMT+8
FPS180	35	No	Traditional food	Female	Indian	Fresh Juice	2019/05/08 1:12:06 AM GMT+8
FPS145	33	Maybe	Traditional food	Male	Indian	Fresh Juice	2019/05/07 8:32:30 PM GMT+8
FPS045	21	Yes	Western food	Male	Malaysian	Carbonated drinks	2019/05/07 4:00:28 PM GMT+8
FPS177	30	Maybe	Traditional food	Male	Indian	Fresh Juice	2019/05/07 11:05:22 PM GMT+8
FPS228	34	Yes	Traditional food	Female	Indian	Fresh Juice	2019/05/08 1:22:46 PM GMT+8
FPS230	40	Maybe	Western food	Male	Indian	Fresh Juice	2019/05/08 1:25:49 PM GMT+8
FPS217	24	No	Western food	Female	Indian	Fresh Juice	2019/05/08 12:18:07 PM GMT+8
FPS043	24	Yes	Western food	Male	Canadian	Fresh Juice	2019/05/07 3:59:01 PM GMT+8
FPS265	25	Yes	Traditional food	Male	Indian	Fresh Juice	2019/05/10 1:06:37 AM GMT+8
FPS144	38	No	Traditional food	Female	Indian	Fresh Juice	2019/05/07 6:16:52 PM GMT+8
FPS052	27	No	Western food	Male	Korean	Fresh Juice	2019/05/07 4:04:58 PM GMT+8
FPS200	37	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 12:11:05 PM GMT+8
FPS152	24	Yes	Traditional food	Male	Indian	Fresh Juice	2019/05/07 6:57:42 PM GMT+8
FPS048	25	Yes	Western food	Male	Indian	Fresh Juice	2019/05/07 4:01:44 PM GMT+8
FPS023	20	Maybe	Western food	Male	Malaysian	Fresh Juice	2019/05/07 3:21:27 PM GMT+8
FPS053	26	No	Traditional food	Male	Indian	Fresh Juice	2019/05/07 4:05:29 PM GMT+8
FPS207	24	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 12:10:04 PM GMT+8
FPS255	17	No	Traditional food	Female	Indian	Fresh Juice	2019/05/08 11:07:06 PM GMT+8
FPS016	25	Maybe	Traditional food	Male	Pakistan	Fresh Juice	2019/05/07 3:09:47 PM GMT+8
FPS199	40	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 11:01:41 AM GMT+8
FPS186	40	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/08 1:17:19 AM GMT+8
FPS141	63	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/07 6:15:32 PM GMT+8
FPS168	26	No	Traditional food	Male	Indian	Fresh Juice	2019/05/07 9:10:10 PM GMT+8
FPS047	24	Yes	Western food	Male	Pakistani	Fresh Juice	2019/05/07 4:01:42 PM GMT+8
FPS134	39	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/07 6:05:40 PM GMT+8
FPS158	9	Maybe	Traditional food	Male	Indian	Fresh Juice	2019/05/07 7:16:13 PM GMT+8
FPS229	56	No	Traditional food	Female	Indian	Fresh Juice	2019/05/08 1:24:22 PM GMT+8
FPS086	32	Maybe	Traditional food	Female	Indian	Fresh Juice	2019/05/07 4:40:48 PM GMT+8
FPS146	25	Maybe	Western food	Male	Indian	Fresh Juice	2019/05/07 6:35:04 PM GMT+8
FPS104	19	Maybe	Western food	Female	Indian	Fresh Juice	2019/05/07 5:04:10 PM GMT+8
FPS284	27	Yes	Western food	Male	Indian	Fresh Juice	2019/05/10 9:24:00 AM GMT+8

**1.7** Crie pelo menos uma visualização de dados que utilize um gráfico de barras e dê suporte à resposta para a(s) pergunta(s) de negócio estabelecida(s) na questão 3. Apresente a visualização gerada e como ela ajuda a responder a solução do problema proposto.

**1.8** Crie pelo menos uma visualização de dados que utilize um gráfico de dispersão e dê suporte à resposta para a(s) pergunta(s) de negócio estabelecida(s) na questão 3. Apresente a visualização gerada e como ela ajuda a responder a solução do problema proposto.

## Processando os dados e plotando alguns gráficos

Nesta etapa, vamos nos conectar as tabelas do Cassandra via Python

e faremos alguns processando nos dados com PySpark para respondermos

as perguntas iniciais sobre os dados. Também plotaremos alguns gráficos

para deixar a análise um pouco mais visual.



## Instalando bibliotecas não existentes na máquina

```
!pip install findspark
```

Collecting findspark Downloading findspark-2.0.1-py2.py3-none-any.whl (4.4 kB) Installing collected packages: findspark Successfully installed findspark-2.0.1

## Importando as bibliotecas

```
from pyspark import SparkContext, SparkConf  
  
from pyspark.sql import *  
  
from pyspark.sql.functions import *  
  
import os  
  
import sys  
  
import findspark  
  
import numpy  
  
import pandas as pd  
  
import matplotlib.pyplot as plt
```

## Configurando o SparkConf

```
findspark.init()  
  
os.environ['PYSPARK_SUBMIT_ARGS'] = '--packages com.datastax.spark:spark-cassandra'  
  
conf = SparkConf()  
  
conf.set('spark.cassandra.auth.username', 'cassandra')  
  
conf.set('spark.cassandra.auth.password', 'cassandra')  
  
conf.set('spark.cassandra.connection.host', 'localhost')  
  
conf.set('spark.cassandra.connection.local_dc', 'dc1')
```

## Criando o objeto SparkContext, através do objeto SparkConf configurado (conf)

```
sc = SparkContext().getOrCreate(conf) # Criando um objeto SQLContext do Spark também, que utiliza o objeto SparkContext  
sqlContext = SQLContext(sc)
```

```
import matplotlib.pyplot as plt
```

```
x = [42, 46, 47, 51, 54, 56, 59, 60, 65, 67]  
y1 = [1, 1, 2, 1, 1, 2, 2, 1, 1, 1]  
y2 = [1, 1, 2, 1, 1, 2, 2, 1, 1, 1]  
plt.plot(x, y1, label="line L")  
plt.plot(x, y2, label="line H")  
plt.plot()
```

```
plt.xlabel("x axis")  
plt.ylabel("y axis")  
plt.title("Line Graph Example")  
plt.legend()  
plt.show()
```

