2020

Big Data Architecture & Governance



**Student Names:**

Tanushree Deshmukh

Varada Dharmadhikari

Saumil Shah

**Assignment Name:**

Marketing Analysis Group 1

# Contents

[2. Assignment 2](#_Toc35761211)

[2.1. Case 2](#_Toc35761212)

[2.2. Assignment Goals 2](#_Toc35761213)

[2.2.1. VeLelero Deliverables 2](#_Toc35761214)

[2.2.2. Assignment deliverables 2](#_Toc35761215)

[3. Documentation 5](#_Toc35761216)

[3.1. Vision Diagram 5](#_Toc35761217)

[3.2. Data Wrangling and Cleansing 6](#_Toc35761218)

[3.3. Database Installation 7](#_Toc35761219)

[3.4. Data Mapping and Integration 8](#_Toc35761220)

[3.5. Metadata 9](#_Toc35761221)

[3.5.1. Business Metadata 9](#_Toc35761222)

[3.5.2. Technical Metamodel 9](#_Toc35761223)

[3.6. Power BI installation 10](#_Toc35761224)

[3.7. Data Validation and Data Visualization 11](#_Toc35761225)

[3.8. System Integration and User Acceptance Testing 12](#_Toc35761226)

[3.9. Challenges Encountered 13](#_Toc35761227)

[3.10. End User Instructions 14](#_Toc35761228)

# Assignment

## Case

The Olist Marketing Funnel dataset contains information about Marketing Qualified Leads(MQLs) who were contacted between June 1st, 2017 and July 1st , 2018. The aim of this project is to analyze the Origin, Business Segment and Type of these MQLs. The other dataset used is the Brazilian E-commerce website dataset, to generate reports for the Sales and Marketing, Finance and Operations teams.

## Assignment Goals

Work with the assigned datasets to perform/create the following:

### Velero Deliverables

* Create Group Assignment/Project on Velero with below mentioned activities:
  + Project, Project Plan (Tasks/Milestones)
  + Issues-Risks
  + Assigned activities
* Use Timesheet to post the time spent on each task/activity

### Assignment deliverables

#### Documentation

* Vision Diagram – Drawn the vision diagram for your project
* Perform Data Wrangling and Cleansing - Pandas/Alteryx/XSV
  + Use the tools for joining datasets
  + Filtering and Aggregating
  + Missing value handling
  + Deriving additional columns from existing datasets
  + Cleaning (removing blank spaces, formatting dates, Capitalizing etc.)
* Database Installation (Configuring your Database and Environment Setup with Sample Data)
* Data Mapping and Integration to your Database for the Entire Dataset
* Metadata
  + **Business Metadata:** Provide a table that includes business terms and complete description for each business term related to your assignment.
  + **Technical Metamodel:** Complete technical metadata information regarding to the assignment and how is defined in the assigned database
* Power BI installation and Configuring and the connector for used for your Database
* Data Validation and Data Visualization using Power BI on your Dataset (Post Importing Data in the assigned Database)
* System Integration and User Acceptance Testing – define test cases used
* Risks/Issues of project and End Users of your Dashboards
* Challenges Encountered and work around for them
* End User Instructions - describe steps to run the Dashboard.

#### Dashboard

#### Documentation

* Vision Diagram – Drawn the vision diagram for your project
* Perform Data Wrangling and Cleansing - Pandas/Alteryx/XSV
  + - Use the tools for joining datasets
    - Filtering and Aggregating
    - Missing value handling
    - Deriving additional columns from existing datasets
    - Cleaning (removing blank spaces, formatting dates, Capitalizing etc.)
* Database Installation (Configuring your Database and Environment Setup with Sample Data)
* Data Mapping and Integration to your Database for the Entire Dataset
* Metadata
  + - **Business Metadata:** Provide a table that includes business terms and complete description for each business term related to your assignment.
    - **Technical Metamodel:** Complete technical metadata information regarding to the assignment and how is defined in the assigned database
* Power BI installation and Configuring and the connector for used for your Database
* Data Validation and Data Visualization using Power BI on your Dataset (Post Importing Data in the assigned Database)
* System Integration and User Acceptance Testing – define test cases used
* Risks/Issues of project and End Users of your Dashboards
* Challenges Encountered and work around for them
* End User Instructions - describe steps to run the Dashboard.

#### Dashboard

Once you wrangle/clean/join/integrate the data, import the data into Power BI and illustrate how to use the data in your database to create a dashboard that will illustrate various analytic aspects of the data. The dashboard should have a target audience that will be able to use the information to perform their job.

Questions to consider:

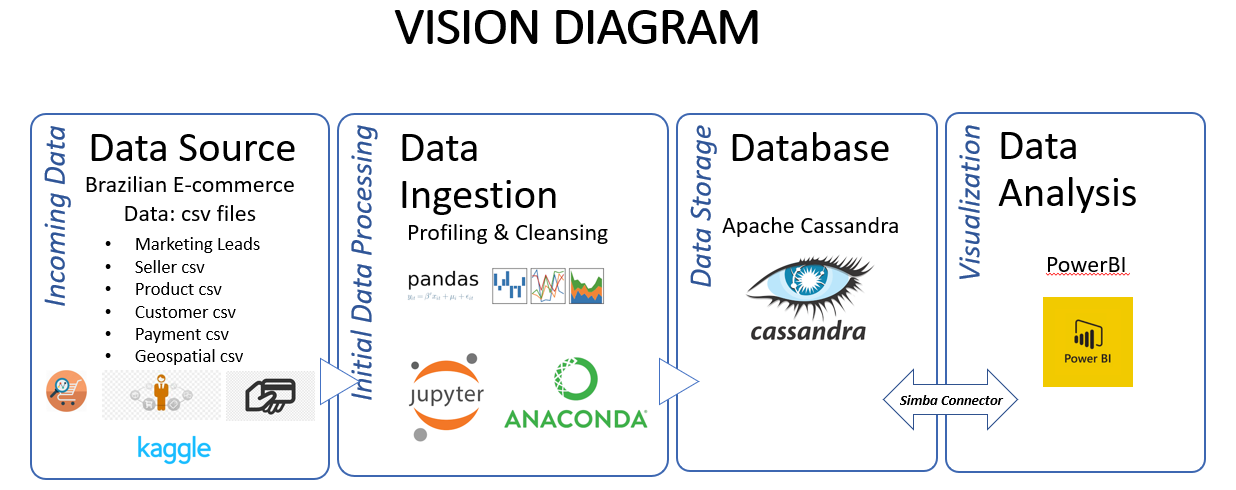
* **Slice and Dice Data:** Which columns are dimensions; which columns are measures?
* **Flexibility:** How would you generate new dimensions?
* **Data Aggregation**: What will you do to summarize measures?
* **Audience**: Who would use this dashboard?
* **Usability**: What dashboard will be used for?

#### Deliverables

* Assignment documentation: use this document and complete all the topics identified in 1.1.1.1.
* Presentation: your presentation should include the following topics
  + Velero
    - Project Plan
    - Risk & Issues
    - Time analysis
  + Metadata: overview of your business and technical metadata and how technical metadata is stored in the database
  + Dashboard: Complete explanation of the dashboard

# Documentation

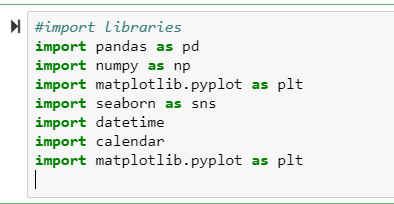
## 3.1 Vision Diagram



## Data Wrangling, Validation and Cleansing

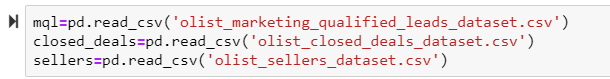
**Using Jupyter Notebook and Pandas**

First of all, we need to import the libraries to the Jupyter notebook so that we can use all of python’s functionality.



**Loading the csv files to the notebook**

For the first data set.

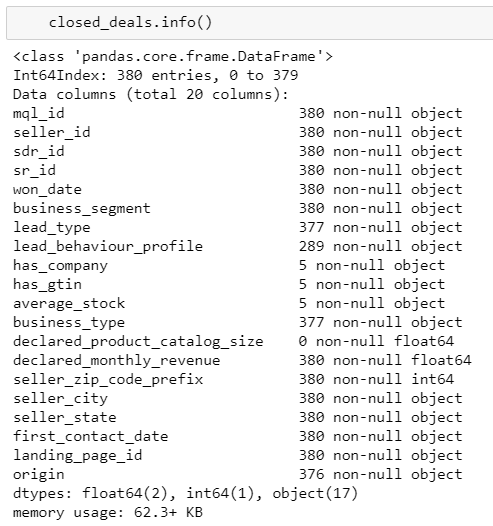
****

Merging some data sets and displaying top 5 rows of aggregated set using .head()

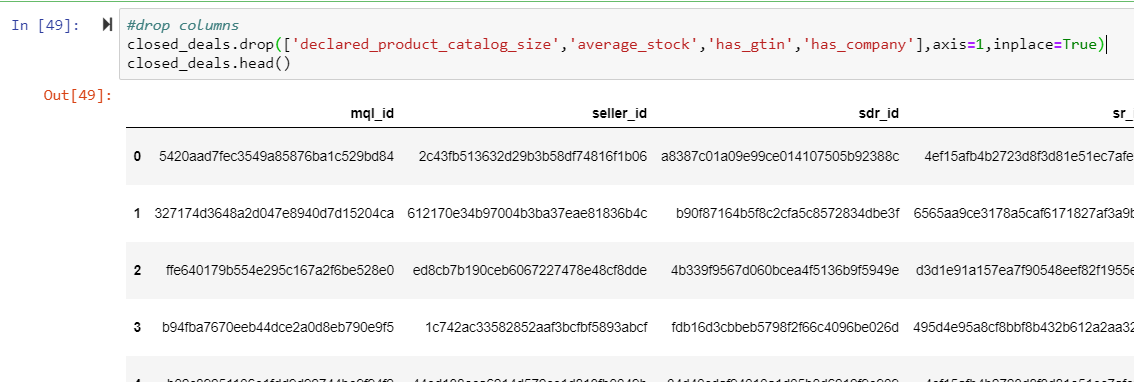
****

**Checking data types and missing values**

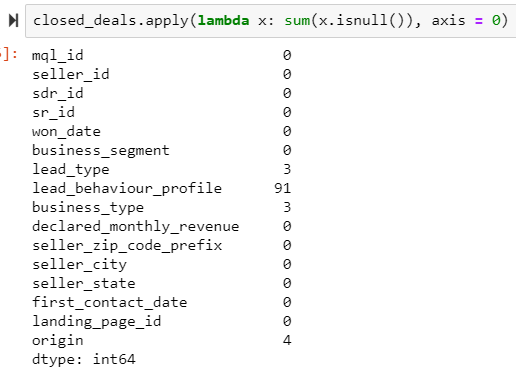
The CSVs which were loaded to the local variable needs to be checked for all the column’s data type and missing values which helps to do the data cleaning.



**Dropping Columns which we do not need**

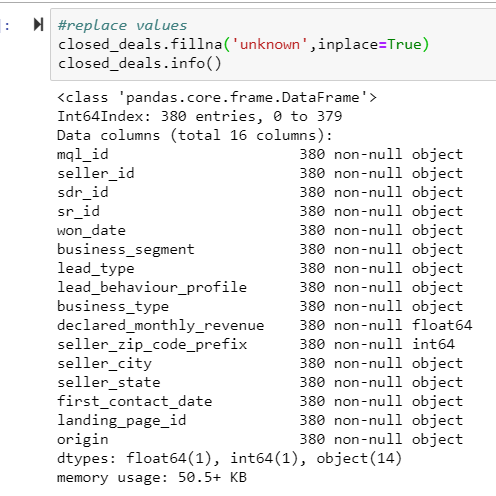


**Checking for NULL Values**

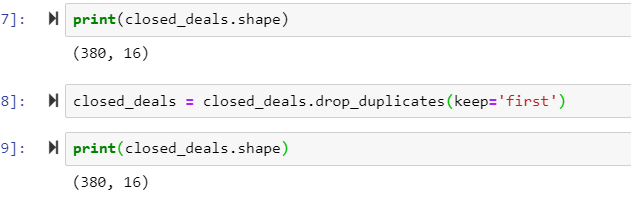


**Taking care of NULL values**

Replacing Nulls with ‘’unknown” . We followed this approach so that we do not miss any data since that might be the case if we remove the records with NULL values.

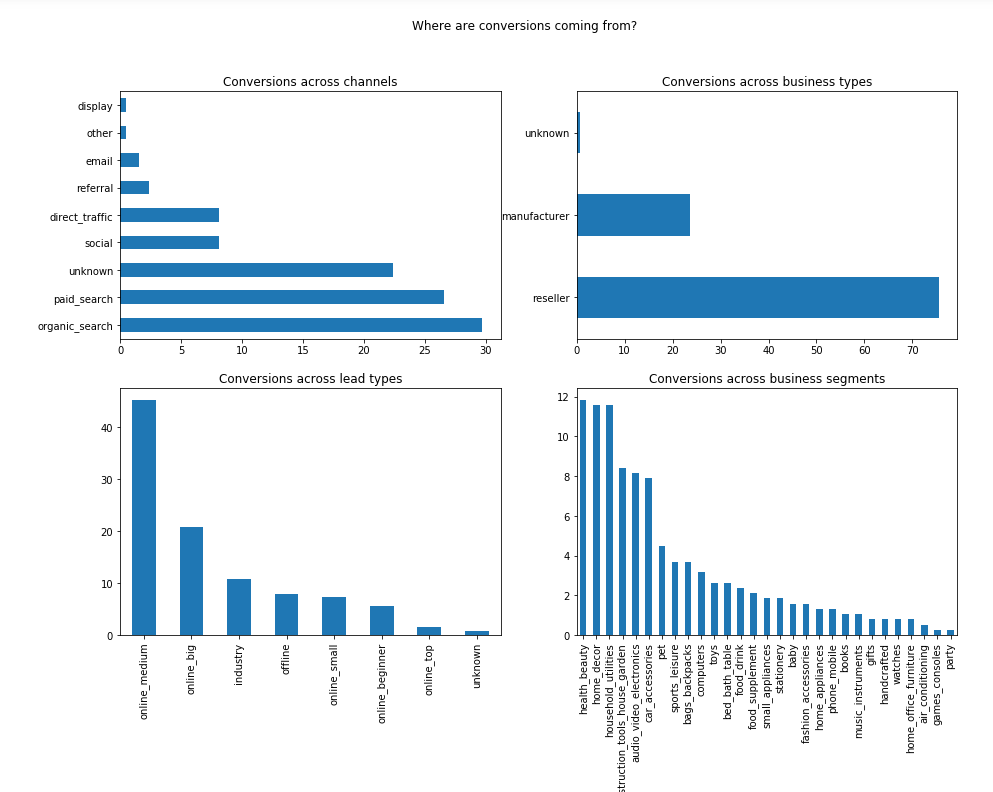


**Checking for duplicate values in data**

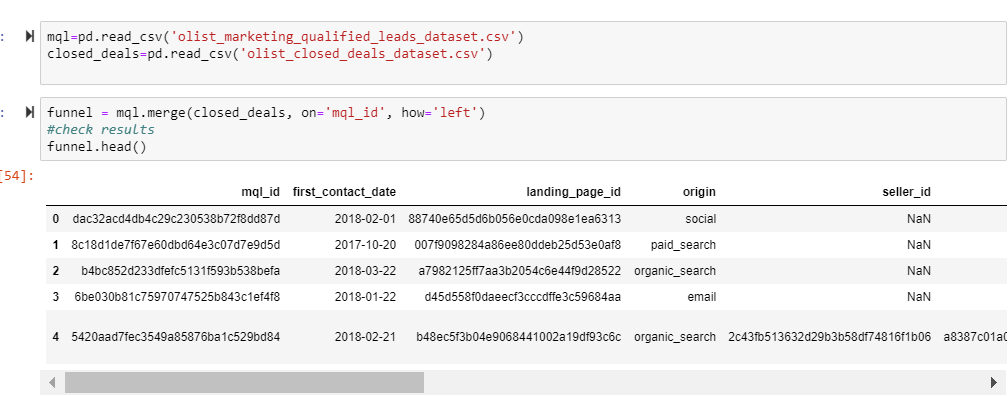


Some visualizations to see from where marketing lead conversions are coming from?





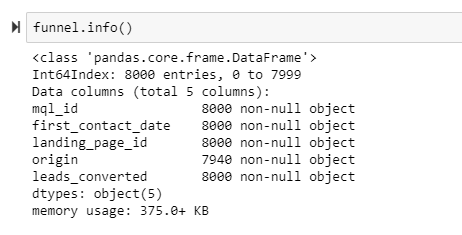
**Analysis on dataset : Leads conversion rate analysis**



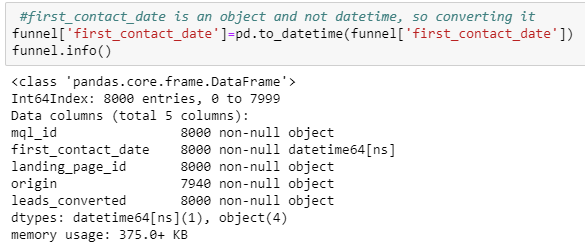
**Creating a new variable 'converted'** to establish which leads converted - Yes/No

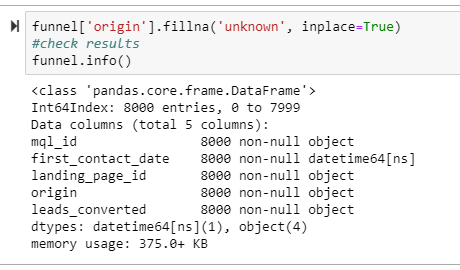


**Checking data types**

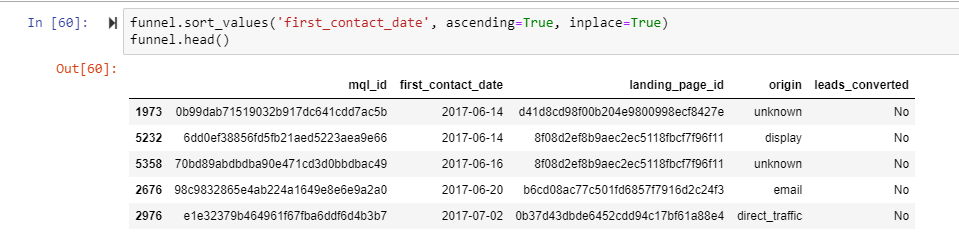


First\_contact\_date is an object and not datetime, so converting it to required format

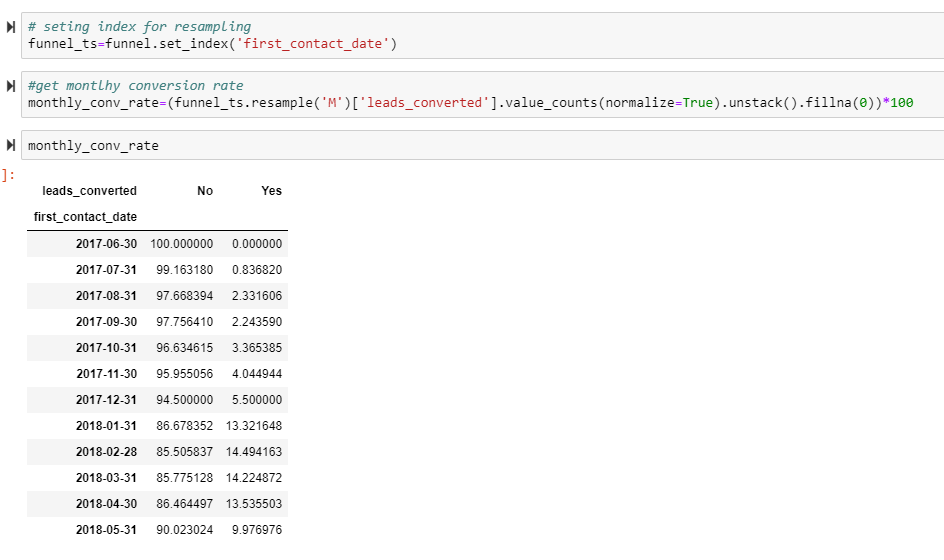




**Sorting** values in ascending order

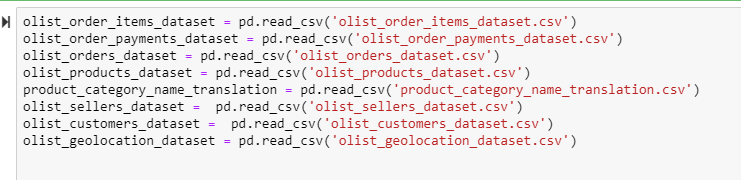


**Setting index for resampling** and calculating monthly conversion rate

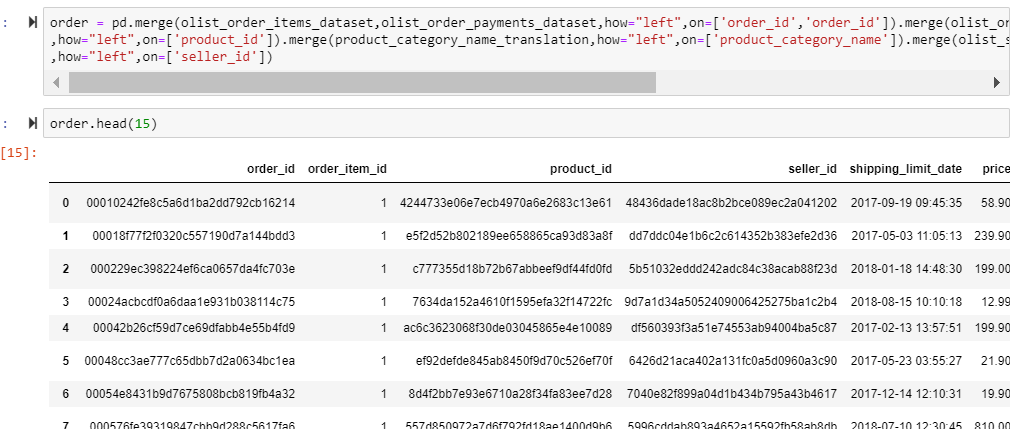


**Dataset 2:**

Similarly for this data set as well ,importing all libraries first and then reading all CSV files.



Merging all datasets into one :



Exporting the merged file ‘’order” into CSV



Thus, we have done the basic cleaning of data like removing the duplicates, taking care of NULL values etc. using pandas. The data is now clean and can be fed to database Cassandra.

Additional analysis on the data has also been done in PowerBI. Please refer to PowerBI visualization section for more insights, analysis and dashboards.

## Database Installation

# Installing Cassandra

# Step 1: Install JRE

 Install Latest JRE if you haven’t installed it already.

 To check if it is installed à go to cmd à type jre -version

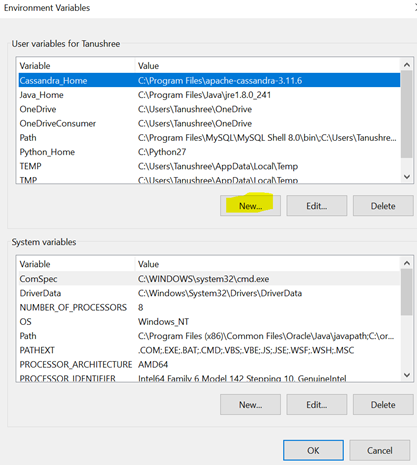
It should be 1.8 and above or else download and install from this link “<https://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>”

Choose Windows 64 if your system is windows 64 or else X86

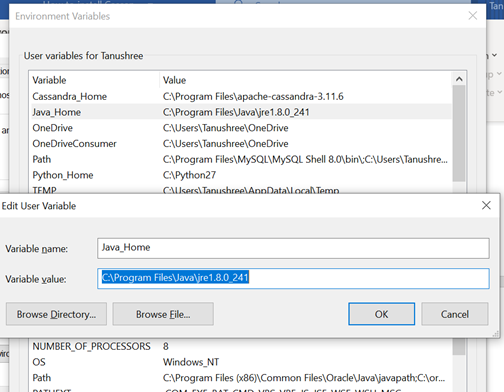
**Step 2: Setup the Path to the environment variable**

Once you install JRE, Add the path of the JRE to your path

To do this go to start >> edit environment variables for your account >>



Click on new>> and mention the path of your JRE directory and click ok.

****

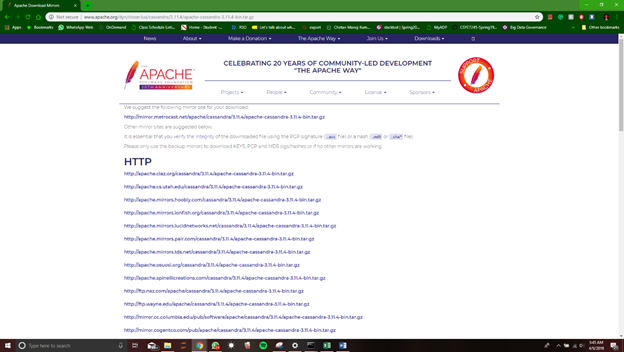
Once path is set, check the path by going to CMD and then running this command

**Echo %JAVA\_HOME%**

Step 3: Download and Install Cassandra

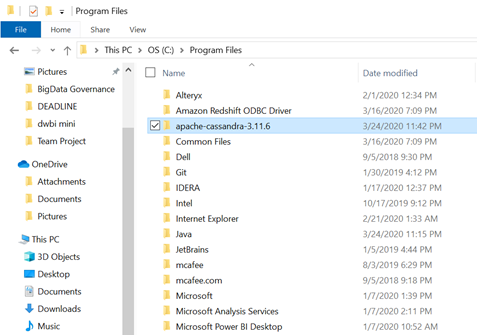
 Install Cassandra Database from this

**“**<http://www.apache.org/dyn/closer.lua/cassandra/3.11.4/apache-cassandra-3.11.4-bin.tar.gz>**”**

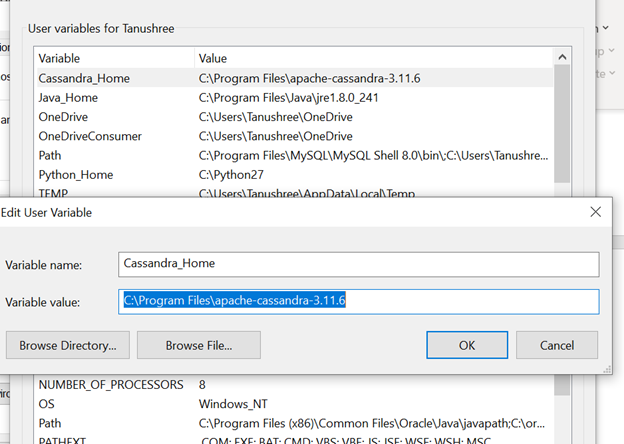
****

**Step 4:  Set up Cassandra**

# Once downloaded, open the folder where it is downloaded and extract it.

****

Once you copied the files to your C drive’s Program files ,add the path to the variable as you did for the JRE in Environment Variables

****

**Step 5: Install Python 2.7**

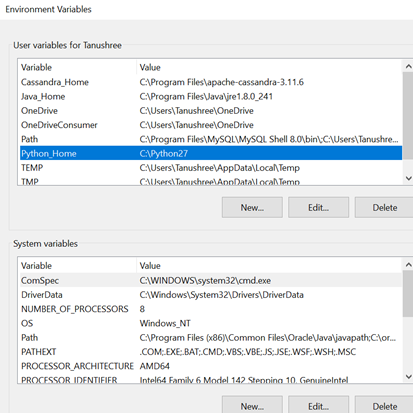
 Install **Python 2.7** , please make sure its not any higher version than this since only this version is compatible.

If you haven’t installed the python 2.7 then you need to install it and you can get the python 2.7 from this link

<https://www.python.org/downloads/release/python-2716/>

Once downloaded add the path of the python to the computer’s variable. go to start >> edit environment variables for your account >>

Click on path and then click on edit. >>  add the python’s directory as its path



**Step 6:  Install Apache Thrift**

 One of the additional component we need to install  in this process is Apache Thrift which you can get it from this link “<http://www.apache.org/dyn/closer.cgi?path=/thrift/0.12.0/thrift-0.12.0.exe>”

Download and install it. You can just follow along the instructions by reading while installing.

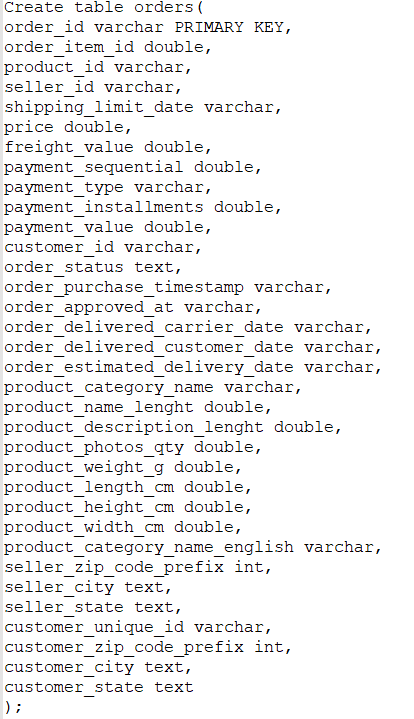
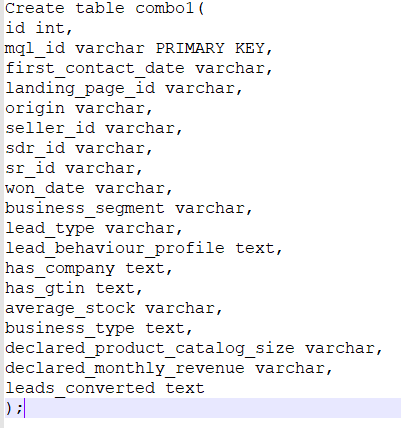
Finally our Cassandra is installed!!

## Data Mapping and Integration

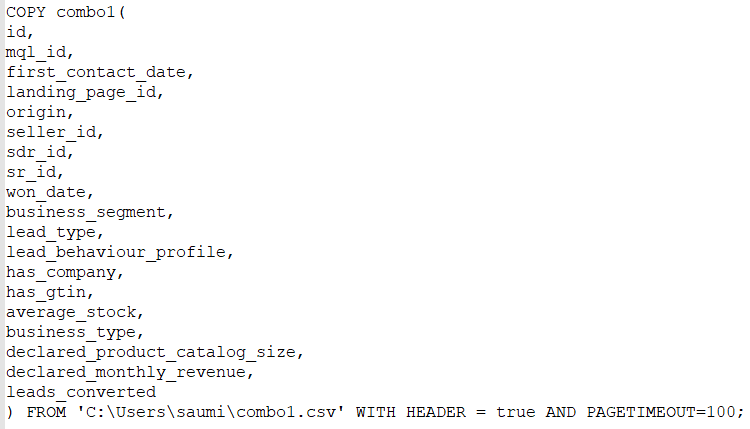
We got clean data from python and then we had to integrate with our cassandra, so we first had to create a keyspace using a command:

CREATE KEYSPACE test WITH replication = {'class': 'SimpleStrategy', 'replication\_factor' : 3};

After the key space is created, we need to create the tables where we can copy the csv files, so we create the table structures with all the columns same as the csv with accurate data type. So we write the create table query with all the columns from the csv and with all their respective datatype.

Once we create the table structure we write the copy code and give the path of where the csv is stored so that the csv can be copied into the table created.

Once the files are copied to the tables we can check if all the content is present by writing a select count query if the row counts are correct.

Now to connect the tables to PowerBI we need to connect with the Simba ODBC connector, select the keyspace that we created and select the table we want to use for visualization.

### Metadata

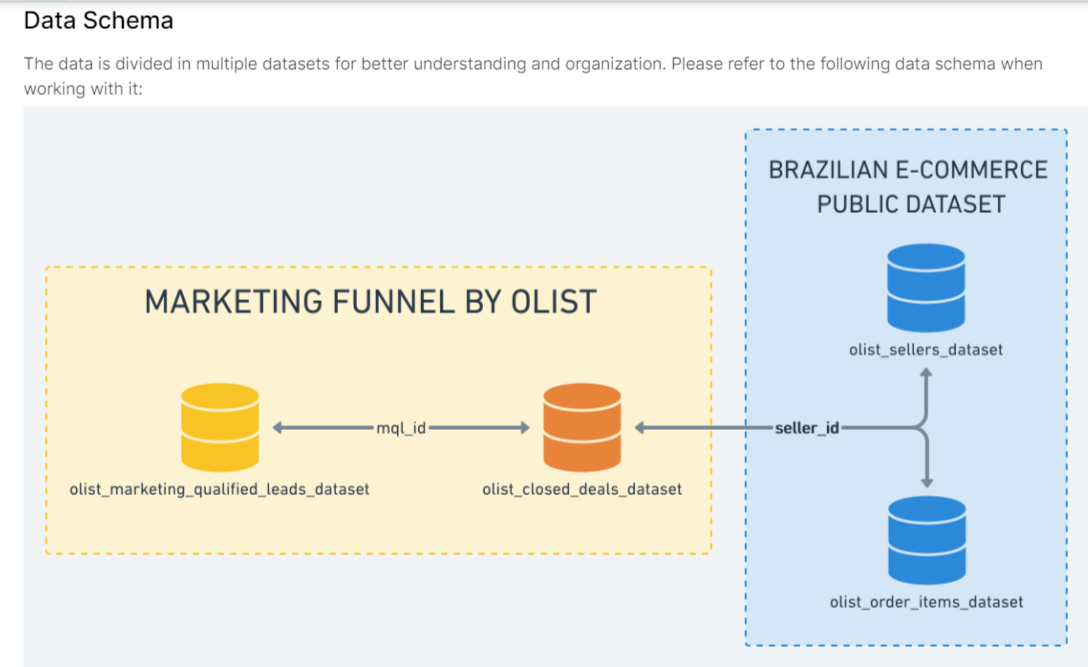
### 3.5.1 Business Metadata

**1.**   **Dataset Repository:**

* Marketing Funnel by Olist

*Dataset from sellers that filled-in requests of contact to sell* their products on [Olist Store](http://www.olist.com/)

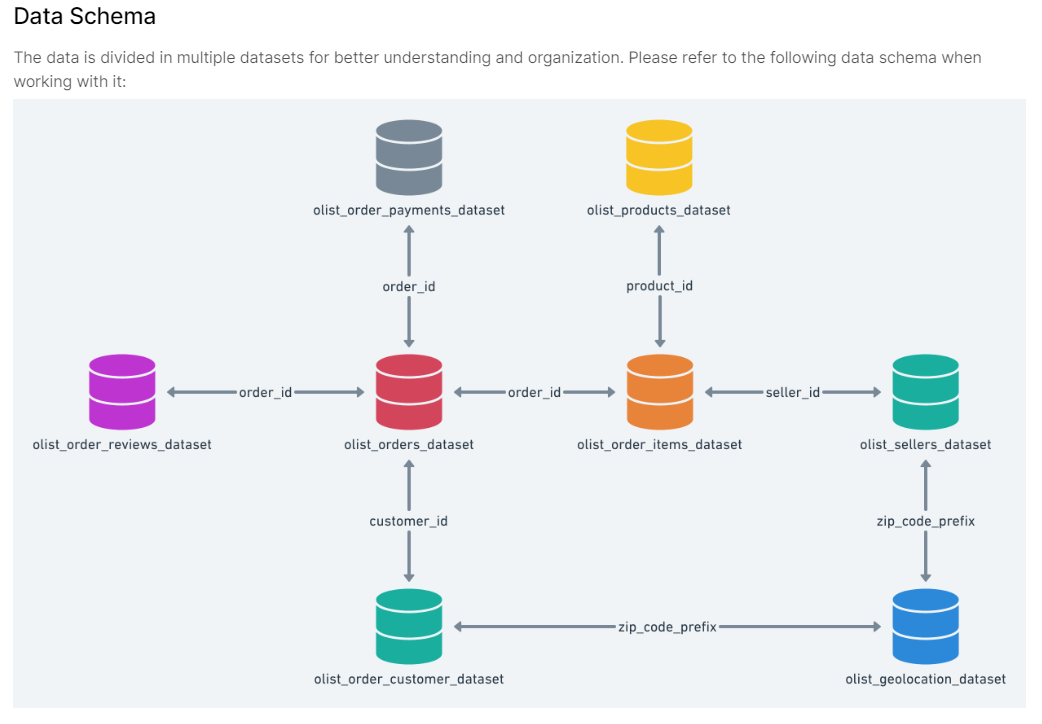
# <https://www.kaggle.com/olistbr/marketing-funnel-olist>



# Brazilian E-Commerce Public Dataset by Olist

# <https://www.kaggle.com/olistbr/brazilian-ecommerce>

*Dataset has information of 100k orders from 2016 to 2018 made at multiple marketplaces in Brazil. Its features allow viewing an order from multiple dimensions: from order status, price, payment and freight performance to customer location, product attributes and reviews written by customers*



**2.**   **Glossary:**

**Marketing Funnel by Olist**

**Olist\_closed\_deals\_dataset**

|  |  |
| --- | --- |
| mql\_id | Marketing Qualified Lead id |
| seller\_id | Seller id |
| sdr\_id | Sales Development Representative id |
| sr\_id | Sales Representative |
| won\_date | Date the deal was closed |
| lead\_type | Lead type. Informed on contact |
| business\_segment | ead business segment. Informed on contact |

**Olist\_marketing\_qualified\_leads\_dataset**

|  |  |
| --- | --- |
| mql\_id | Marketing Qualified Lead id |
| first\_contact\_date | Date of the first contact solicitation |
| landing\_page\_id | Landing page id where the lead was acquired |
| origin | Type of media where the lead was acquired |

# Custom Fields

|  |  |
| --- | --- |
| leads\_converted | Yes/No depending on if a lead converted or not |
| date\_diff | time taken to convert a lead |

# Brazilian E-Commerce Public Dataset by Olist

# olist\_order\_items\_dataset (Line Item)

|  |  |
| --- | --- |
| order\_id | order unique identifier |
| order\_item\_id | sequential number identifying number of items included in the same order |
| product\_id | product unique identifier |
| seller\_id | seller unique identifier |
| shipping\_limit\_date | Shows the seller shipping limit date for handling the order over to the logistic partner |
| price | item price |
| freight\_value | item freight value item (if an order has more than one item the freight value is splitted between items) |

**olist\_order\_payments\_dataset**

|  |  |
| --- | --- |
| order\_id | order unique identifier |
| payment\_sequential | customer may pay an order with more than one payment method. If he does so, a sequence will be created to accommodate all payments |
| payment\_type | method of payment chosen by the customer |
| payment\_installments | number of installments chosen by the customer |
| payment\_value | transaction value |

**olist\_orders\_dataset**

|  |  |
| --- | --- |
| order\_id | unique identifier of the order |
| customer\_id | key to the customer dataset. Each order has a unique customer\_id |
| order\_status | Reference to the order status (delivered, shipped, etc) |
| order\_purchase\_timestamp | Shows the purchase timestamp |
| order\_approved\_at | Shows the payment approval timestamp |
| order\_delivered\_carrier\_date | Shows the order posting timestamp. When it was handled to the logistic partner |
| order\_delivered\_customer\_date | Shows the actual order delivery date to the customer |
| order\_estimated\_delivery\_date | Shows the estimated delivery date that was informed to customer at the purchase moment |

**olist\_products\_dataset**

|  |  |
| --- | --- |
| product\_id | unique product identifier |
| product\_category\_name | root category of product, in Portuguese |
| product\_name\_lenght | number of characters extracted from the product name |
| product\_description\_lenght | number of characters extracted from the product description |
| product\_photos\_qty | number of product published photos |
| product\_weight\_g | product weight measured in grams |
| product\_length\_cm | product length measured in centimeters |
| product\_height\_cm | product height measured in centimeters |
| product\_width\_cm | product width measured in centimeters |

**olist\_sellers\_dataset**

|  |  |
| --- | --- |
| seller\_id | seller unique identifier |
| seller\_zip\_code\_prefix | first 5 digits of seller zip code |
| seller\_city | seller city name |
| seller\_state | seller state name |

**product\_category\_name\_translation**

|  |  |
| --- | --- |
| product\_category\_name | category name in Portuguese |
| product\_category\_name\_english | category name in English |

**olist\_customers\_dataset**

|  |  |
| --- | --- |
| customer\_id | key to orders dataset. Each order has a unique customer\_id |
| customer\_unique\_id | unique identifier of a customer |
| customer\_zip\_code\_prefix | first five digits of customer zip code |
| customer\_city | customer city name |
| customer\_state | customer state name |

1. **Business Content**

Our aim is to analyze data sets in detail and suggested recommendations for decision making in business. Some of the pointers are :

·      *To find total / monthly conversion rate for the leads*

·      *Number of customers by Product Category*

·      *Average price per Product / Product Category*

·      *Number of Products sold by seller / city*

·      *Revenue generated yearly / monthly*

**4.**   **End Users /Business Clients:**

# a.     Olist (E-commerce website)

# Marketing and Sales team

# Finance team

# b.     Seller

# Marketing and Sales team

# Advertising and Sales teams

### 3.5.2 Technical Metamodel

|  |  |
| --- | --- |
| File Name & Size | Marketing Funnel by Olist                                             855KB |
| Brazilian E-Commerce Public Dataset by Olist    120MB |
| Type of Compression | Zip |
| OS used to run software | Windows |
| Hardware Processor Name | Intel(R) Core i7 |
| Hardware RAM | 16 GB |
| Database Used | Cassandra |
| Tools Used | PowerBI, JupyterNotebook, Microsoft Excel,Velero |
| Connector Name (for connecting database to PowerBI ) | Simba |

## Power BI installation

**How to Install PowerBI**

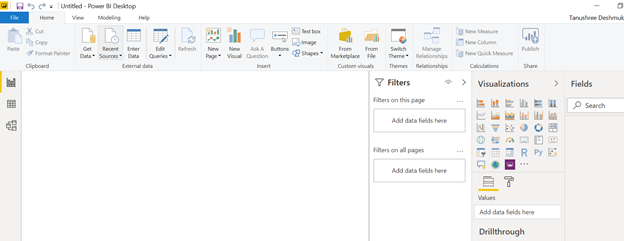
**Step 1: Download Microsoft PowerBI**

 Power BI can be downloaded from the Microsoft website :

“<https://powerbi.microsoft.com/en-us/downloads/>”

You can sign-up for a 30 day free trial.

## Step 2: Install Power BI



Follow the procedure according to the prompt and click finish once the setup is complete.

Open PowerBI and login to your Microsoft account.

You can connect different databases to PowerBI and create dashboards/reports and visualizations.

**Connecting Cassandra to the PowerBI**

**Step 1: Download Simba Cassandra ODBC Connector**

 Download the Simba Cassandra connector using the below provided link.

“<https://www.simba.com/drivers/cassandra-odbc-jdbc/>”

 Create an account by signing up so that the SIMBA will send you trial license key which you need to use the ODBC connector

**Step 2: Install Cassandra and set up to use**

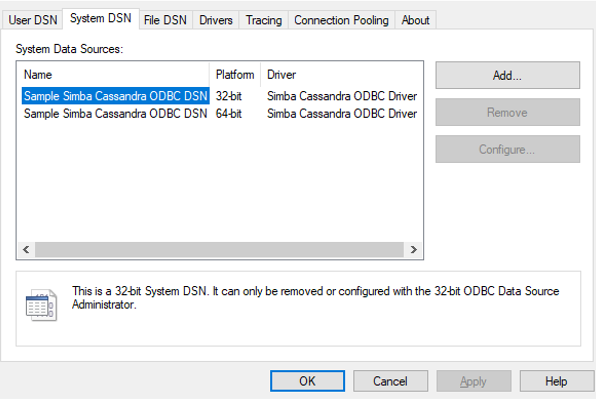
 Once downloaded Cassandra connector, follow the prompt and install the connector.

After installing, go to directory and in the bin, folder copy the license file which you will receive from the Simba after registering on their site.

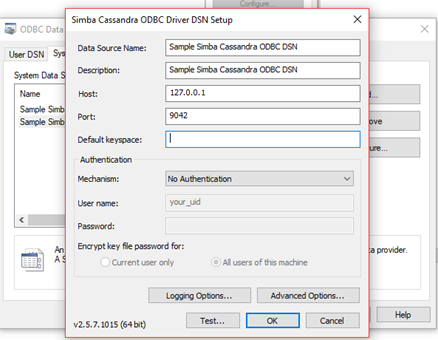
**Step 3: Create Connection**

 Once license file is added, Go to Start >> Search for ODBC

Open ODBC and navigate to system DSN.



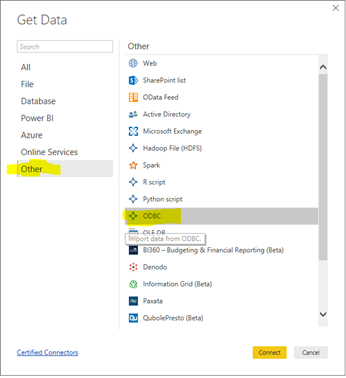
Click on the Sample Simba Cassandra ODBC DSN and then click on the configure. And provide the Cassandra server information.



After providing all the information, test the connection.

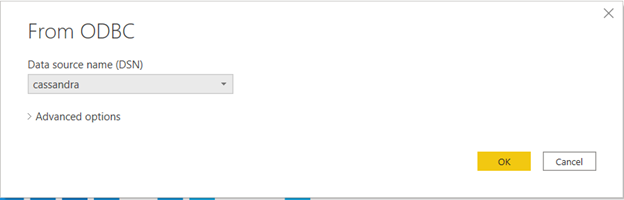
**Step 4: Connect PowerBI using Simba connector**

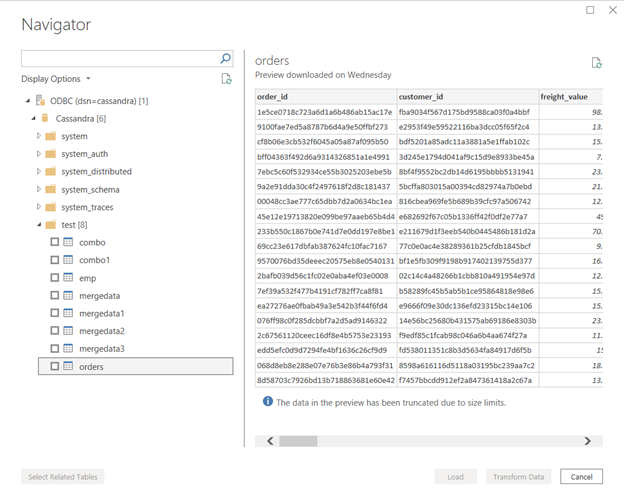
 Open PowerBI and then click on get Data and select ODBC.



Click on connect and select Simba connector name from the drop down and click on Ok.

 Select all the tables which we need for the analysis and then click on Load.



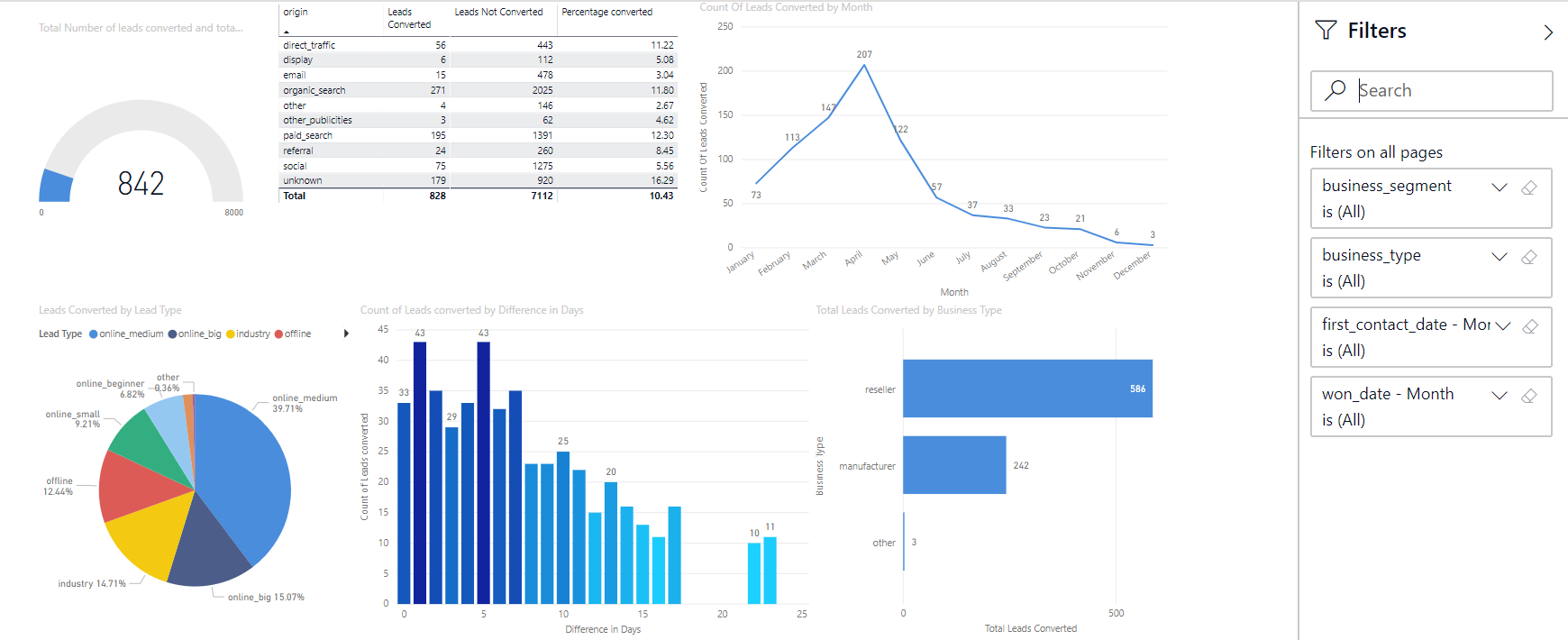


 Once loaded the Power BI has all the records.

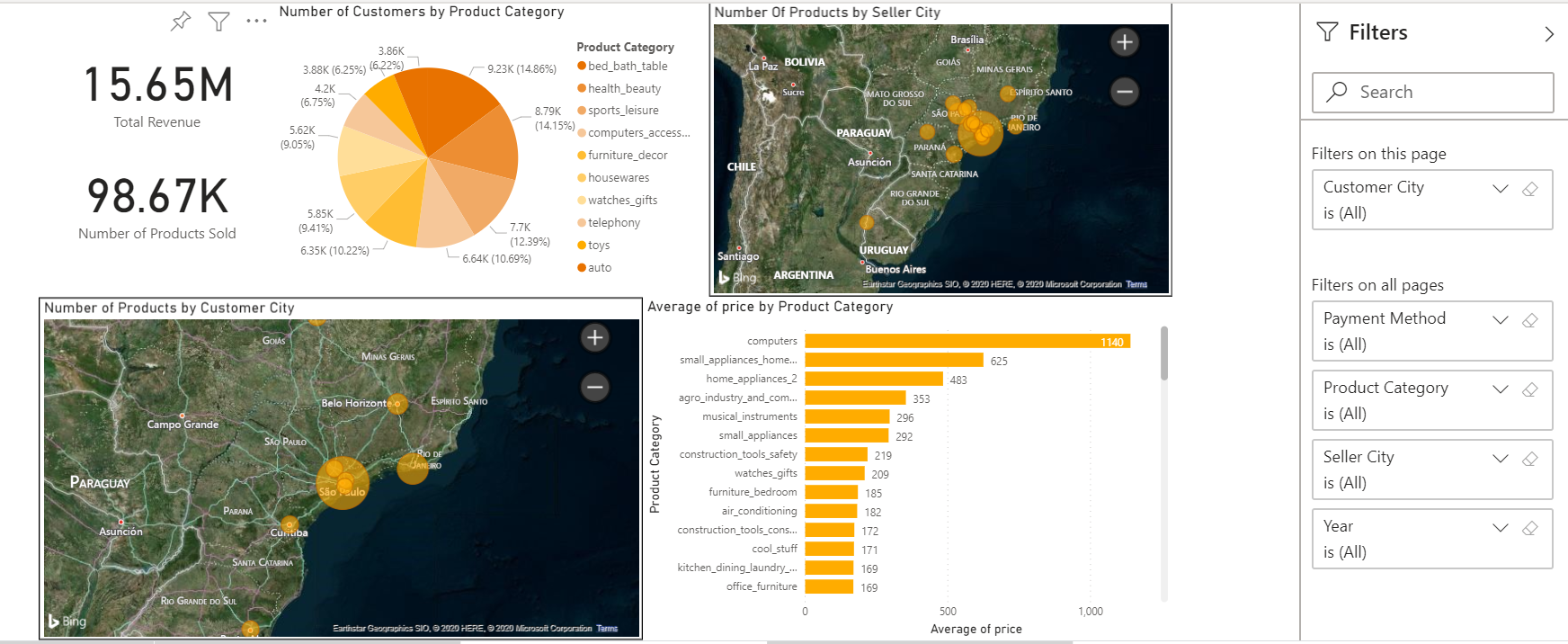
## 3.7 Data Validation and Data Visualization

We have created the following reports:

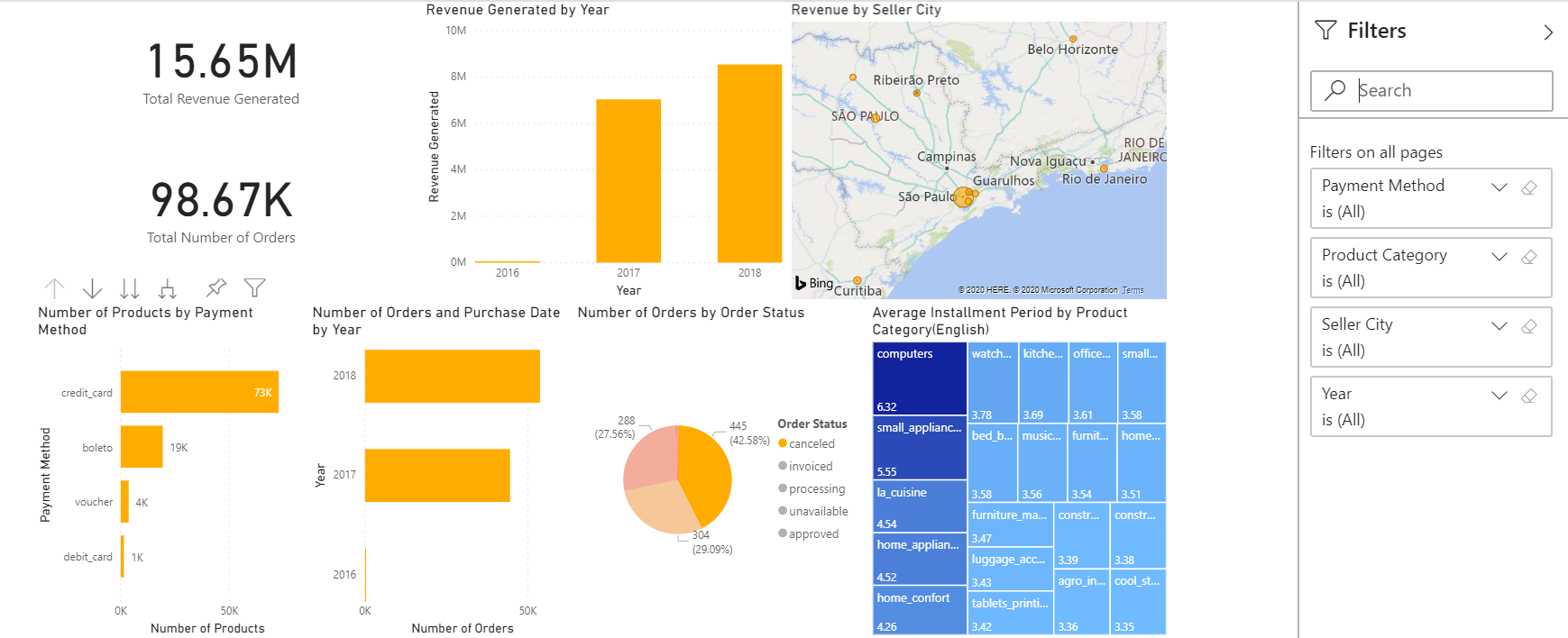
1. Marketing Leads Analysis Report:



2. Sales and Marketing Report



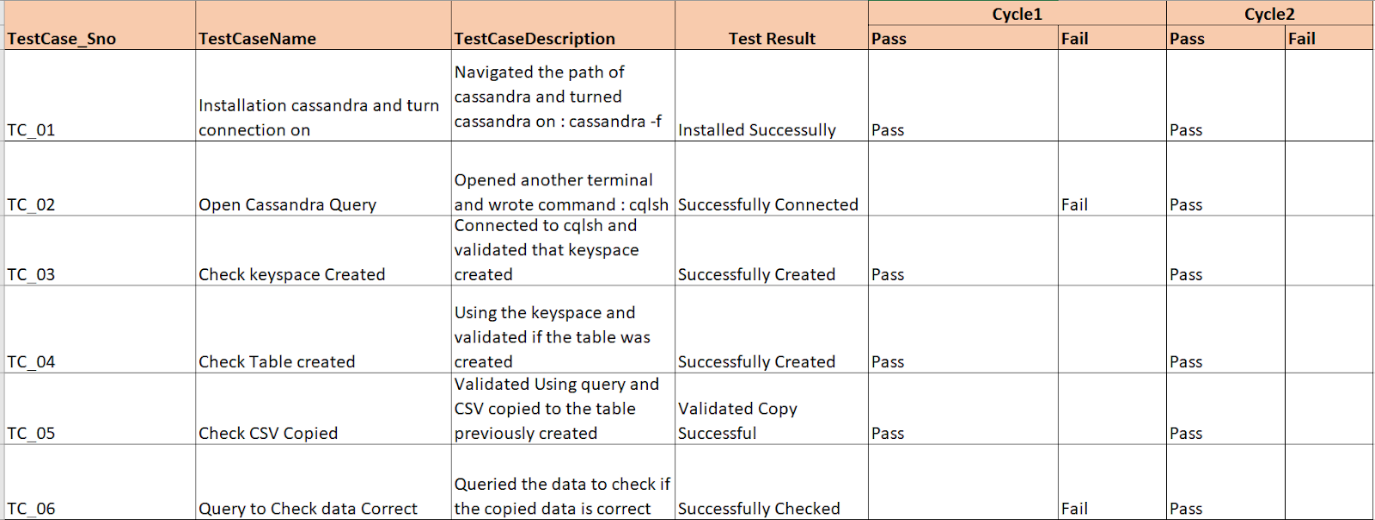
3. Finance and Operations Reports



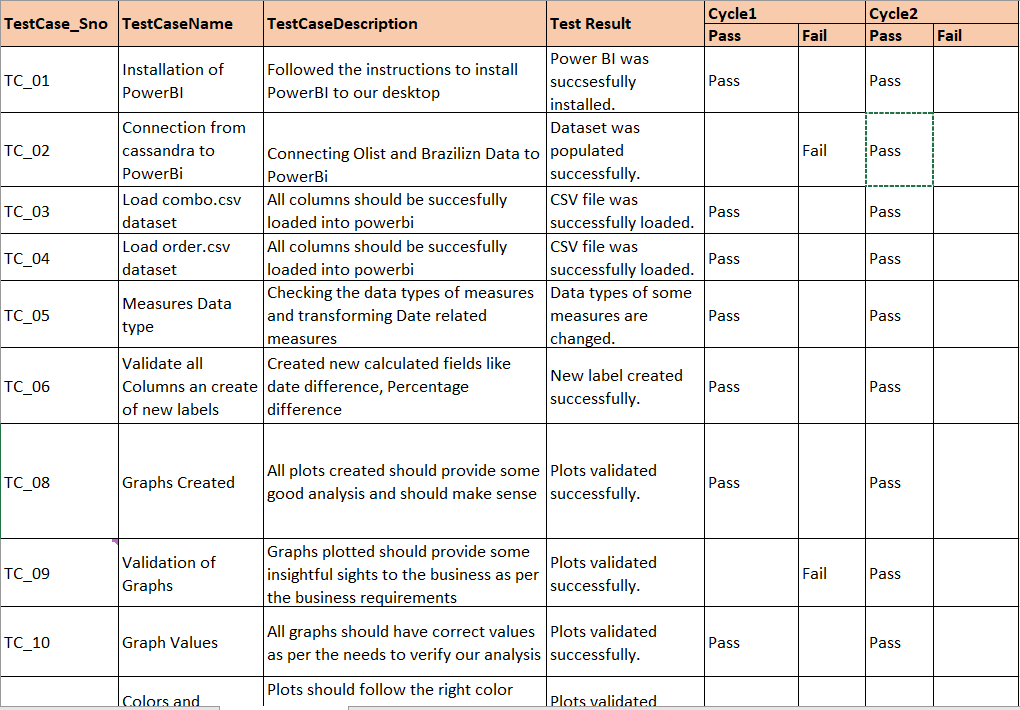
## System Integration and User Acceptance Testing

We made two testing scenarios one for Installing Cassandra and another for Testing the Power BI environment

For the Installation we created various test scenarios and ran it for two cycles and noted the final test results in the columns.



For the PowerBI test we ran different test scenarios ranging from establishing a connection to the working of the dashboard and did two cycles of tests and noted the results.



## Challenges Encountered

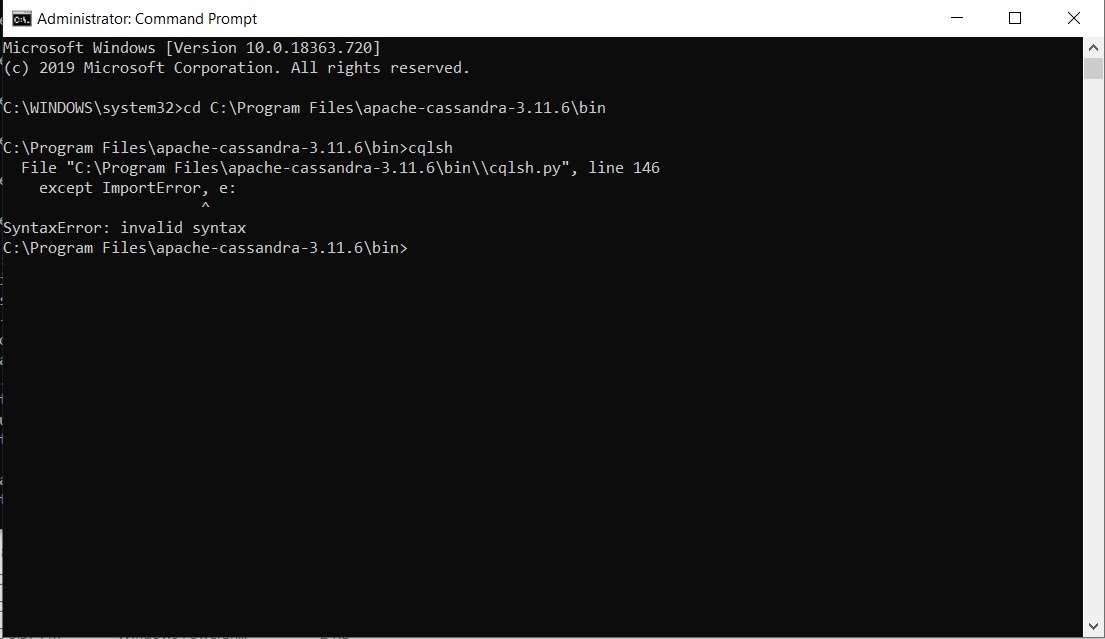
Risk is the possibility of losing something of value. Values can be gained or lost when taking risk resulting from a given action or inaction, foreseen or unforeseen (planned or not planned). Risk can also be defined as the intentional interaction with uncertainty. Uncertainty is a potential, unpredictable, and uncontrollable outcome; risk is a consequence of action taken despite uncertainty.

Issues are present focused, always negative and need to be fixed to be on the track of the project’s goal. Issues are when an event that has in fact has occurred and the team now must work on resolution.

 Here we will be discussing some of risks and issues we have faced while implementing our project

 Python Version Issue:

While installing Cassandra the Cassandra connection was successfully loaded but while starting the Cassandra query cqlsh there was a python error. First, we thought there might be some error in a file, but later we found that we needed python 2.7 and we had python 3.8 on our systems. So, we downgraded our python versions and reinstalled Cassandra and it worked.



Null values in the dataset:

We had a few Null and Unknow values in our dataset. We wanted a clean dataset for analysis, so we used pythons’ pandas library to clean and filter the unwanted records so that we are left with only useful data. We also used pandas to create useful new columns like ‘lead\_converted’ using some calculations for further analysis

 ODBC Connector:

This was the main issue and roadblock which we faced while doing this project. Choosing the ODBC connectors which connect Cassandra database with the PowerBI and the information must be input to make the connection was a task, must check quite a few connectors before we chose to go with SIMBA ODBC CONNECTOR.

 Joining two Datasets:

We wanted to join the Geographical dataset with the orders dataset, but we could not include the Latitude and longitude values because they were joining with other Rows and giving a huge row count of unwanted data. So, we Joined both the dataset on common fields like state and city.

 Data Loading:

Loading from the CSV was quite a task where we came with so many errors while loading. The error log had little or no information at all which had no help in debugging.

We resolved this by deleting all the empty rows and columns and then tried to load.

 Visualization:

The data when loaded to the power BI data visualization tool, there were a couple of columns where we needed to change the data type to get the better visualization for example: data type of ‘won\_date’ columns so that the visualization of date difference.

The data had to be aggregated and calculated manually on almost all the columns in the power BI tool to get the analysis done.

 Risk of losing important data:

In the given dataset there were some Null and unknown values. We needed to handle that so that we did not lose important and crucial data that could be used for the analysis. So, we made use of python’s pandas library to clean the data and get an optimal dataset.

## End User Instructions

* Download the following files:

1. olist\_closed\_deals.csv and olist\_marketing\_qualified\_leads.csv from [Olist Marketing Funnel Dataset](https://www.kaggle.com/olistbr/marketing-funnel-olist#olist_closed_deals_dataset.csv) and
2. The olist\_customers\_dataset.csv, olist\_order\_items\_dataset.csv, olist\_order\_payments.csv, olist\_product.csv, olist\_sellers\_dataset.csv [Brazilian E-commerce dataset](https://www.kaggle.com/olistbr/brazilian-ecommerce/home) from Kaggle.

* Merge and clean these files using Pandas
* Load the merged and cleaned dataset into Cassandra
* Connect Cassandra with Power BI to make visualizations

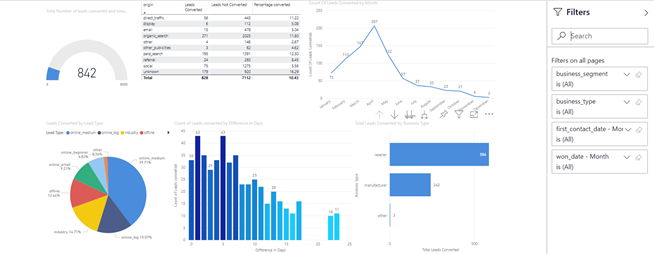
**Context of the datasets - what we did**

In this project, we have analyzed the Olist Marketing Funnel Dataset and Brazilian e-commerce datasets taken from Kaggle.

Marketing Leads Analysis Report

[Olist Marketing Funnel Dataset](https://www.kaggle.com/olistbr/marketing-funnel-olist#olist_closed_deals_dataset.csv) – The dataset contains information about 8k Marketing Qualified Leads that were contacted between June 1st 2017 and July 1st 2018, to sell their products on the Olist e-commerce website. Out of these 8000 MQLs, a number of leads were converted. This dataset gives information about the origin of the lead(seller), their business segment and business type.

A detailed report has been published about Marketing Leads Analysis on [Marketing Leads Analysis Report](https://app.powerbi.com/groups/me/reports/76229a3c-be0a-404a-ae7a-4b892948e5a5?ctid=a8eec281-aaa3-4dae-ac9b-9a398b9215e7)



The above report contains information about

• Total number of leads that were converted – hence onboarded as sellers on the website

• Percent of converted leads by origin

• Number of leads converted by month

• Leads converted by the type of lead – online/offline

• Number of days it took to convert a lead

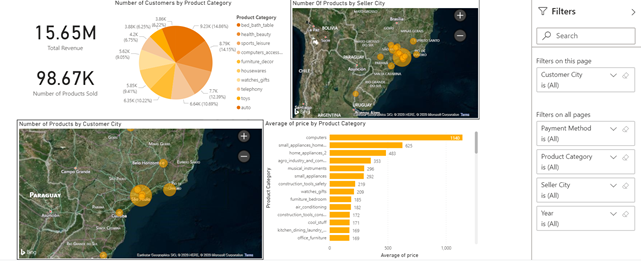
• Number of leads converted by business type and segment

To make optimal use of the dashboard, we have also enabled report wide filters, that enable the user to see lead conversion information based on business segment, business type, when the leads were first contacted and when they were onboarded.

Sales and Marketing Report

This report is based on [Brazilian E-commerce dataset](https://www.kaggle.com/olistbr/brazilian-ecommerce/home). The information used for this report has been taken from Brazilian E-commerce Dataset’s product, seller, payment, customer, order and geography csv files. This report can be useful for the Sales and Marketing teams of Olist and the sellers listed on Olist.

These files have been merged to create the Sales and Marketing Report – published on Power BI.



The above report contains information about

• Total Revenue generated by Olist

• Number of Products sold from June 1st 2017 to July 1st 2018

• Number of products sold by category

• Top 20 seller cities that sell the most number of products

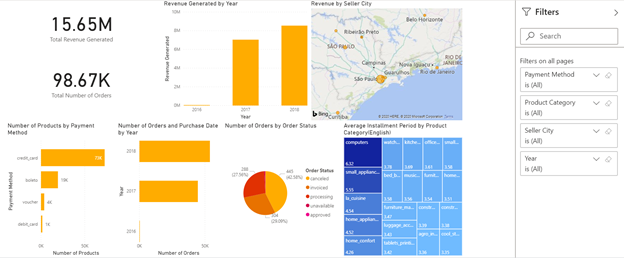
• Top 20 customer cities that buy the most number of products

• Average price of product category

To make optimal use of this report, we have enabled report wide filters, that enable the user to see Sales and Marketing information based on the Year, Seller City, Customer City and product category.

Finance and Operations Report

This report is based on [Brazilian E-commerce dataset](https://www.kaggle.com/olistbr/brazilian-ecommerce/home#olist_customers_dataset.csv). The information used for this report has been taken from Brazilian E-commerce Dataset’s product, seller, payment, customer, order and geography csv files. This report can be useful for the Finance and Operations teams of Olist and the sellers listed on Olist.

The files have been merged to create the [Finance and Operations Report](https://app.powerbi.com/groups/me/reports/0441a600-1d78-4d87-b734-97ce57397a14?ctid=a8eec281-aaa3-4dae-ac9b-9a398b9215e7) – published on Power BI  


The above report contains information about

• Total Revenue generated

• Total number of orders placed

• Revenue generated by year – that can be drilled down to day of month

• Revenue generated by seller cities – top 20

• Number of products paid for by payment method – that can be drilled down to show the product category of the payment method used

• Number of orders placed by year – that can be drilled down to day of month

• Number of orders that were not delivered – that can be drilled down to the product category

• Product categories for which the payment is made in installments and the average number of installments to complete the payment.

For optimal use of the report, we have also enabled report wide filters, that allow a more specific view of the information for the Finance & Operations teams. The report wide filters include filtering based on Payment method, product category, seller city and year.

In order to view and use these reports, the user must have a Microsoft Power BI account.

## Velero

We used Velero for our project planning, tracking and filling the timesheets. We created Milestones/Task based on our project requirements assigned to an individual and also assigned estimated hours required. As time progressed and the task were complete each participant filled a timesheet for respective tasks.

We also noted the issues and risks associated with the project and how to mitigate them.

