**Data Wrangling Project Pacmann – Olist Ecommerce Data**

1. **Pendahuluan**

Salah satu peran seorag Data Analyst adalah melakukan pemrosesan data dari bentuk *raw* (data mentah) menjadi data yang bersih dan siap untuk dianalisa.

Dalam projek ini saya akan menganalisa Olist Ecommerce Data. Data disimpan dalam database relational dengan nama “olist.db”. Data ini yang kemudian akan saya analisis sesuai tujuan yang telah ditentukan.

Pengolahan data dimulai dengan mengakses data dari SQL untuk dioleh dengan Python. Kemudian melakukan melihat gambaran data, eksplorasi dan pemrosesan data, serta melakukan Analisis Data.

1. **Tujuan**

Tujuan projek kali ini adalah:

1. Mengetahui total penjualan perbulan?
2. Mengetahui kategori produk dengan pendapatan tertinggi?
3. Mengetahui 10 kota dengan penjualan tertinggi?
4. Mengetahui metode pembayaran yang paling diminati
5. **Gambaran Data**

Langkah pertama adalah mengakses data yang masih berupa database “olist.db” kedalam jupyter notebook agar bisa dilakukan pengolahan data dan kemudian di analisis. Disini saya menggunakan library sqlite3 dari Python.

[screenshoot code]

Kemudian mengakses semua table yang ada pada data. Olist Ecommerce Data terdari dari 9 tabel, yaitu:

1. Olist Customer
2. Olist Orders
3. Olist Order Review
4. Olist Order Payments
5. Olist Order Items
6. Olist Product
7. Olist Seller
8. Olist Geolocation
9. Product Category Name Translation
10. **Eksplorasi dan Pemrosesan Data**
11. Indentifikasi NaN
12. Indentifikasi Outlier
13. Indentifikasi Inconsistent Format
14. Indentifikasi Duplicate Data
15. Parsing Data
16. **Analisa Data**
17. Total Penjualan per Bulan
18. Kategori Produk dengan Pendapatan Tertinggi
19. 10 Kota dengan Penjualan Tertinggi
20. Metode Pembayaran yang Paling Diminati
21. **Penutup**

Olist Data Description

1. Olist Customer

* customers\_id: key to the orders dataset. Each order has a unique customer\_id.
* customers\_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

1. Olist Orders

* order\_id: order unique identifier
* customers\_id: key to the orders 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.

1. Olist Order Review

* review\_id: unique review identifier
* order\_id: order unique identifier
* review\_score: note ranging from 1 to 5 given by the customer on a satisfaction survey.
* review\_comment\_title: comment title from the review left by the customer, in Portuguese.
* review\_comment\_message: comment message from the review left by the customer, in Portuguese.
* review\_creation\_date: shows the date in which the satisfaction survey was sent to the customer.
* review\_answer\_timestamp: shows satisfaction survey answer timestamp

1. Olist Order Payments

* order\_id: order unique identifier
* payment\_sequential: a 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.

1. Olist Order Items

* order\_id: order unique identifier
* order\_item\_id: sequential number identifying number of items include 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)

1. Olist Product

* 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

1. Olist Seller

* 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

1. Olist Geolocation

* geolocation\_zip\_code\_prefix: first five digits of customer zip code
* geolocation\_lat: latitude
* geolocation\_lng: longitude
* geolocation\_city: city name
* geolocation\_state: state

1. Product Category Name Translation

* product\_category\_name: category name in Portuguese
* product\_category\_name\_english: category name in English

1. **Total Penjualan per Bulan**

* Menggunakan tabel "Olist Orders" dan "Olist Order Payments"
* Menghitung total pembayaran (payment\_value) dari setiap pesanan (order) berdasarkan tanggal pembelian (order\_purchase\_timestamp) per bulan.
* Melakukan pengelompokan dan agregasi data untuk mendapatkan total penjualan per bulan.
* Menampilkan visualisasi grafik atau tabel yang memperlihatkan tren penjualan bulanan.

1. **10 Kategori Produk dengan Pendapatan Tertinggi**

* Menggunakan tabel "Olist Order Items" dan "Olist Product"
* Menghitung total pendapatan (price) dari setiap kategori produk (product\_category\_name).
* Melakukan penggabungan (join) antara kedua tabel berdasarkan product\_id.
* Mengidentifikasi kategori produk dengan pendapatan tertinggi berdasarkan total pendapatan.
* Menampilkan visualisasi grafik atau tabel yang memperlihatkan kategori produk dengan pendapatan tertinggi.

1. **Analisis Penjualan per Kota**

* Menggunakan tabel "Olist Orders" dan "Olist Customer"
* Menghitung total penjualan (payment\_value) dari setiap pesanan berdasarkan kota pelanggan (customer\_city).
* Melakukan penggabungan (join) antara kedua tabel berdasarkan customer\_id.
* Mengidentifikasi kota dengan total penjualan tertinggi.
* Menampilkan visualisasi grafik atau tabel yang memperlihatkan kota dengan total penjualan tertinggi.

1. Analisis Tingkat Kepuasan Pelanggan

* Menggunakan tabel "Olist Order Reviews" dan "Olist Orders"
* Menghitung rata-rata nilai ulasan (review\_score) yang diberikan oleh pelanggan.
* Melakukan penggabungan (join) antara kedua tabel berdasarkan order\_id.
* Mengidentifikasi tingkat kepuasan pelanggan berdasarkan rata-rata nilai ulasan.
* Menampilkan visualisasi grafik atau tabel yang memperlihatkan tingkat kepuasan pelanggan.