**Analisis Data Transaksi Penjualan untuk Menentukan Pola Belanja Konsumen**

**FADILA SHELY AMALIA**

**22/501751//PPA/06396**

1. **PENDAHULUAN**
2. Latar Belakang

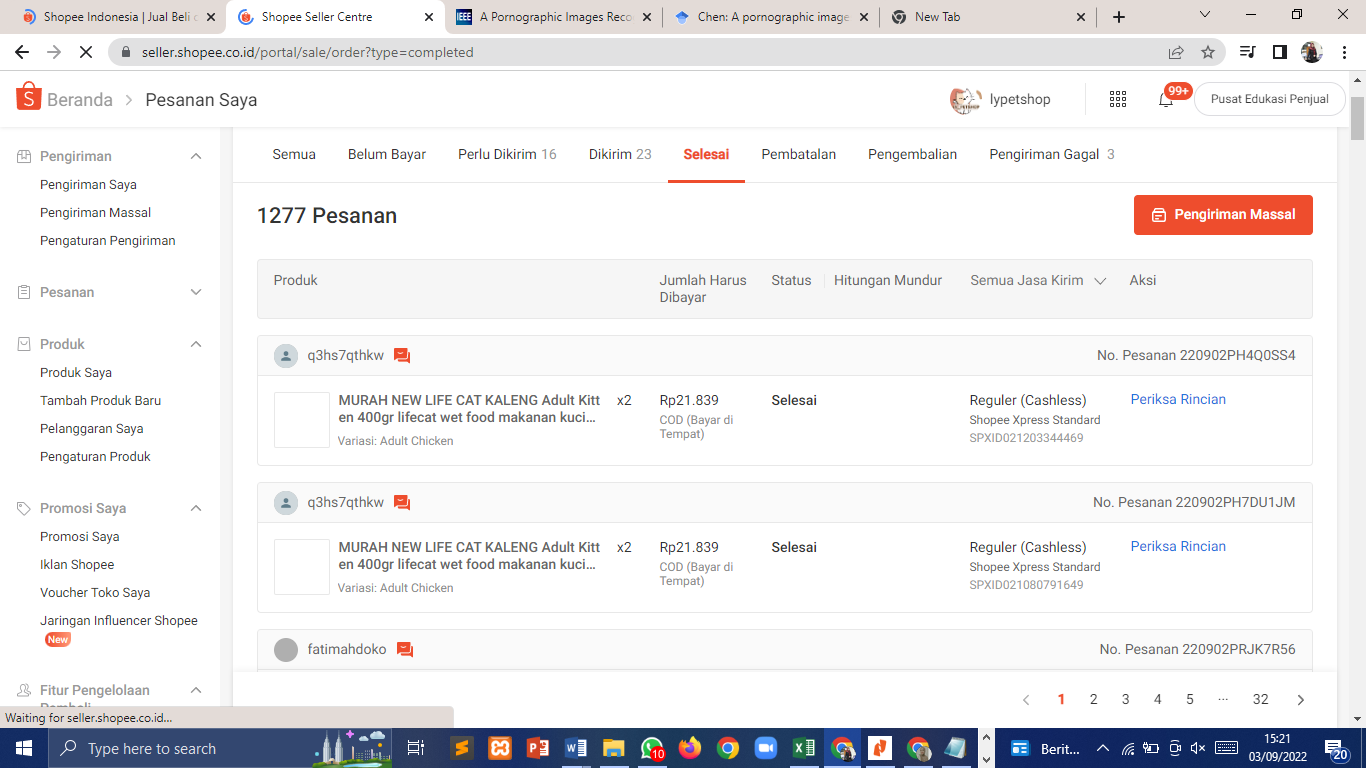
Pemanfaatan teknologi informasi dalam dunia bisnis telah menjadi kebutuhan oleh setiap pelaku bisnis. Dibutuhkan teknik pengumpulan data yang dapat digunakan untuk mengetahui pola belanja konsumen pada saat melakukan transaksi pembelian.

1. Ruang Lingkup

Ruang lingkup yang dibahas dibatasi pada data yang diperoleh dari transaksi penjualan toko *e-commerce* (Lypetshop – Shopee).

1. Tujuan dan Manfaat
2. Mengetahui pola pembelian konsumen pada Lypetshop.
3. Membantu *owner* dalam pengelolaan stok barang yang yang sering dibeli secara bersamaan.
4. Membantu *owner* dalam mengenbangkan strategi bisnis dari pola pembelian konsumen.
5. **METODE**
6. *Source Unstructured Data*

Data didapatkan dari laporan transaksi pengirimin barang yang telah selesai dilakukan oleh Lypetshop. Data diambil dari transaksi bulan Juli 2022 sampai September 2022. Produk yang dijual merupakan makanan dan perlengkapan hewan peliharaan.



1. Prosedur Analisis

Untuk mengubah data transaksi penjualan yang belum terstruktur dilakukan langkah sepeerti berikut :

1. Pengolahan data yaitu menentukan atribut pada data persediaan dan data transaksi penjualan.
2. Dilakukan *preprocessing* data dengan menentukan memfilter data berdasarkan atribut yang telah di tentukan sebelumnya sehingga didapatkan data dalam bentuk tabular.
3. Penyelesaian menggunakan metode apriori
4. Batasan-Batasan
5. Data yang digunakan adalah data transaksi pengiriman yang selesai dilakukan dari bulan Juli 2022 sampai September 2022.
6. Informasi berupa *Association rules*.
7. Nilai *support* dan *confidence* menyesuaikan.
8. **ANALISIS**

Data yang ada akan diesktrasi menjadi data yang lebih terstruktur dengan menghilangkan atribut-atribut yang tidak tiperlukan dalam proses analisis pola belanja konsumen. Pada data transaksi asli terdapat 46 atribut dan pada analisis untuk menemukan pola belanja konsumen menggunakan metode apriori ini hanya memerlukan tiga atribut yakni nomor *invoice* nama dan jumlah barang yang dibeli. Berikut sampel data transaksi yang dapat digunakan untuk analisis ini.

|  |  |  |
| --- | --- | --- |
| Nomor *Invoice* | Nama Produk | Qty |
| 220702DQE97W1V | SUSU TOP GROWTH 1 Murah Sachet 30 gr susu anak kucing / susu kucing | 1 |
| 220702DQE97W1V | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220702DQE97W1V | Makanan kucing ORI CAT Murah repack 1kg bentuk ikan | 1 |
| 220701AXHG8DH2 | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220704HCHHD9SR | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220704HCHHD9SR | Bolt kitten murah kemasan repack 1kg | 1 |
| 220705K46E4DQF | Cat Choize kitten tuna&salmon MURAH | 1 |
| 220705K46E4DQF | Cat Choize kitten tuna&salmon MURAH | 1 |
| 220701B8TX70HB | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220702CD4AW53Y | Cat Choize kitten tuna&salmon MURAH | 1 |
| 220702CD4AW53Y | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220705KQ39UTMB | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220705KQ39UTMB | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220705KQ39UTMB | LIFE CAT POUCH murah Baim Wong 85gr Wet Food makanan kucing | 1 |
| 220702BNA1T6UX | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220705M1HK7JHC | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220705M1HK7JHC | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220701BBC9PGXU | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220705M1AXMXD0 | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220705M1AXMXD0 | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220705M1AXMXD0 | LIFE CAT POUCH murah Baim Wong 85gr Wet Food makanan kucing | 1 |

Berdasarkan Tabel diatas maka mendapatkan data yang sudah di sortir berdasarkan quantity barang yang terjual sesuai nama barang sebagai berikut.

|  |  |  |
| --- | --- | --- |
| Nomor Invoice | Nama Produk | Qty |
| 220702DQE97W1V | SUSU TOP GROWTH 1 Murah Sachet 30 gr susu anak kucing / susu kucing | 1 |
| NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| Makanan kucing ORI CAT Murah repack 1kg bentuk ikan | 1 |
| 220701AXHG8DH2 | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220704HCHHD9SR | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| Bolt kitten murah kemasan repack 1kg | 1 |
| 220705K46E4DQF | Cat Choize kitten tuna&salmon MURAH | 2 |
| 220701B8TX70HB | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220702CD4AW53Y | Cat Choize kitten tuna&salmon MURAH | 1 |
| NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220705KQ39UTMB | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 2 |
| LIFE CAT POUCH murah Baim Wong 85gr Wet Food makanan kucing | 1 |
| 220702BNA1T6UX | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220705M1HK7JHC | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 2 |
| 220701BBC9PGXU | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 1 |
| 220705M1AXMXD0 | NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah | 2 |
| LIFE CAT POUCH murah Baim Wong 85gr Wet Food makanan kucing | 1 |

Maka berdasarkan data yang tertera pada tabel dapat dilihat pada transaksi dengan invoice 220702DQE97W1V terjual dengan tiga kombinasi barang yaitu SUSU TOP GROWTH 1 Murah Sachet 30 gr susu anak kucing / susu kucing, NEW LIFE CAT KALENG Murah Adult Kitten 400gr lifecat wet food makanan kucing kaleng basah, dan Makanan kucing ORI CAT Murah repack 1kg bentuk ikan.

1. ***Data Design***

Setelah melakukan analisis maka dari data yang akan digunakan dirubah dalam bentuk tabular dan merubah nama barang dalam bentuk angka 0 dan angka 1. Dimana jika barang kurang dari sama dengan 0 maka sistem perhitungan keranjang tersebut akan bernilai 0 dan jika nilai lebih dari 1 maka sistem perhitungan kerjang akan memberi *output* (nilai) 1, sehingga jika sebuah nota membeli barang A sebanyak 10 buah maka hanya akan dihitung 1.

Untuk mempermudah pembuatan tabel maka saya akan menyingkat masing-masing nama barang yang dijadikan contoh. Berikut merupakan contoh data tabular yang siap digunakan untuk langkah selanjutnya dalam menentukan nilai terjualnya barang supaya sistem dapat memahami informasi dataset.

Berikut ini data transaksi menggunakan nama yang sudah dipersingkat :

|  |  |  |
| --- | --- | --- |
| Nomor Invoice | Nama Produk | Qty |
| 220702DQE97W1V | Susu Kucing | 1 |
| Wet Food 400gr | 1 |
| Ori Cat 1KG | 1 |
| 220701AXHG8DH2 | Wet Food 400gr | 1 |
| 220704HCHHD9SR | Wet Food 400gr | 1 |
| Bolt 1KG | 1 |
| 220705K46E4DQF | Kitten Food | 2 |
| 220701B8TX70HB | Wet Food 400gr | 1 |
| 220702CD4AW53Y | Kitten Food | 1 |
| Wet Food 400gr | 1 |
| 220705KQ39UTMB | Wet Food 400gr | 2 |
| Wet Food 800gr | 1 |
| 220702BNA1T6UX | Wet Food 400gr | 1 |
| 220705M1HK7JHC | Wet Food 400gr | 2 |
| 220701BBC9PGXU | Wet Food 400gr | 1 |
| 220705M1AXMXD0 | Wet Food 400gr | 2 |
| Wet Food 800gr | 1 |

Setelah itu nama barang di rubah posisinya menjadi bentuk baris dan keterangan terjual dengan *quantity* >= 1 maka ditulis 1 dan tidak terjual adalah 0 terlihat di baris selanjutnya berdasarkan nomor *invoice* seperti berikut.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Susu Kucing | Wet Food 400gr | Ori Cat 1KG | Wet Food 400gr | Bolt 1KG | Kitten Food | Kitten Food | Wet Food 800gr |
| 220702DQE97W1V | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 220701AXHG8DH2 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 220704HCHHD9SR | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 220705K46E4DQF | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 220701B8TX70HB | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 220702CD4AW53Y | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 220705KQ39UTMB | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 220702BNA1T6UX | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 220705M1HK7JHC | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 220701BBC9PGXU | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 220705M1AXMXD0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

Setelah data terbentuk seperti yang ditampilkan pada tabel diatas maka data siap digunakan untuk menentukan *associations rules* atau aturan assosiatif antara suatu kombinasi item.

1. **Implementasi** **Coding**

Data yang diperoleh diperoleh melalui proses pengambilan data yaitu data transaksi dan data persediaan barang berupa file excel. Dalam Data transaksi penjualan terdapat 46 atribut yaitu Invoice, Status Pesanan, Status Pembatalan, No. Resi, Opsi Pengiriman, Antar ke counter/ pick-up, Pesanan Harus Dikirimkan Sebelum (Menghindari keterlambatan), Waktu Pengiriman Diatur, Waktu Pesanan Dibuat, Waktu Pembayaran Dilakukan, SKU Induk, Description, Nomor Referensi SKU, Nama Variasi, Price, Harga Setelah Diskon, Jumlah, Total Harga Produk, Total Diskon, Diskon Dari Penjual, Diskon Dari Shopee, Berat Produk, Quantity, Total Berat, Voucher Ditanggung Penjual, Cashback Koin, Voucher Ditanggung Shopee, Paket Diskon, Paket Diskon (Diskon dari Shopee), Paket Diskon (Diskon dari Penjual), Potongan Koin Shopee, Diskon Kartu Kredit, Ongkos Kirim Dibayar oleh Pembeli, Estimasi Potongan Biaya Pengiriman, Ongkos Kirim Pengembalian Barang, Total Pembayaran, Perkiraan Ongkos Kirim, Catatan dari Pembeli, Catatan, UserID, Nama Penerima, No. Telepon, Alamat Pengiriman, Kota/Kabupaten, Provinsi, InvoiceDate.

Berikut proses yang dilakukan dalam mengubah data tidak terstruktur dalam bentuk yang terstruktur menggunakan bahasa pemrograman python.

1. Menambahkan Library

Library yang digunakan untuk memproses data yaitu numpy dan pandas. Dimana numpy digunakan untuk …. dan pandas digunakan untuk …. Berikut ini code yang diimplementasikan :

#Langkah 1 (Import Library yang diperlukan)

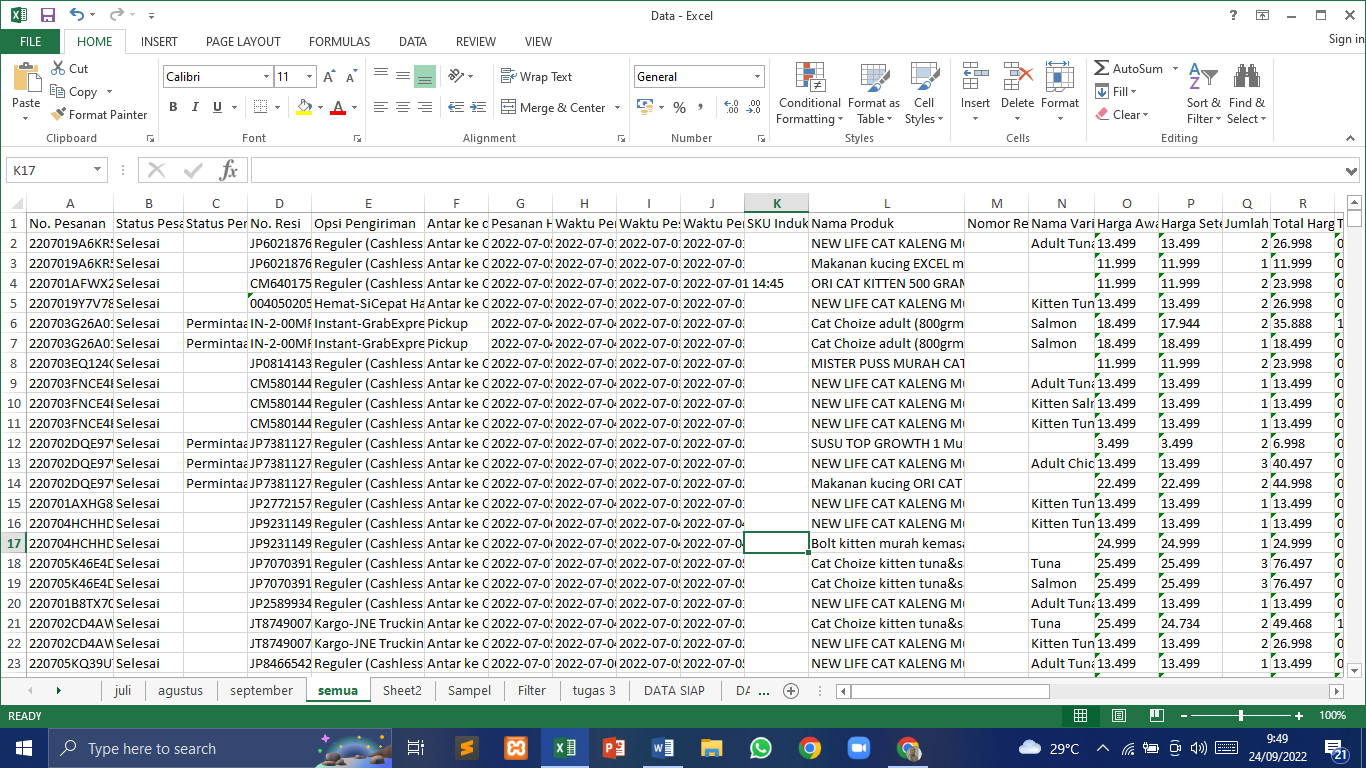
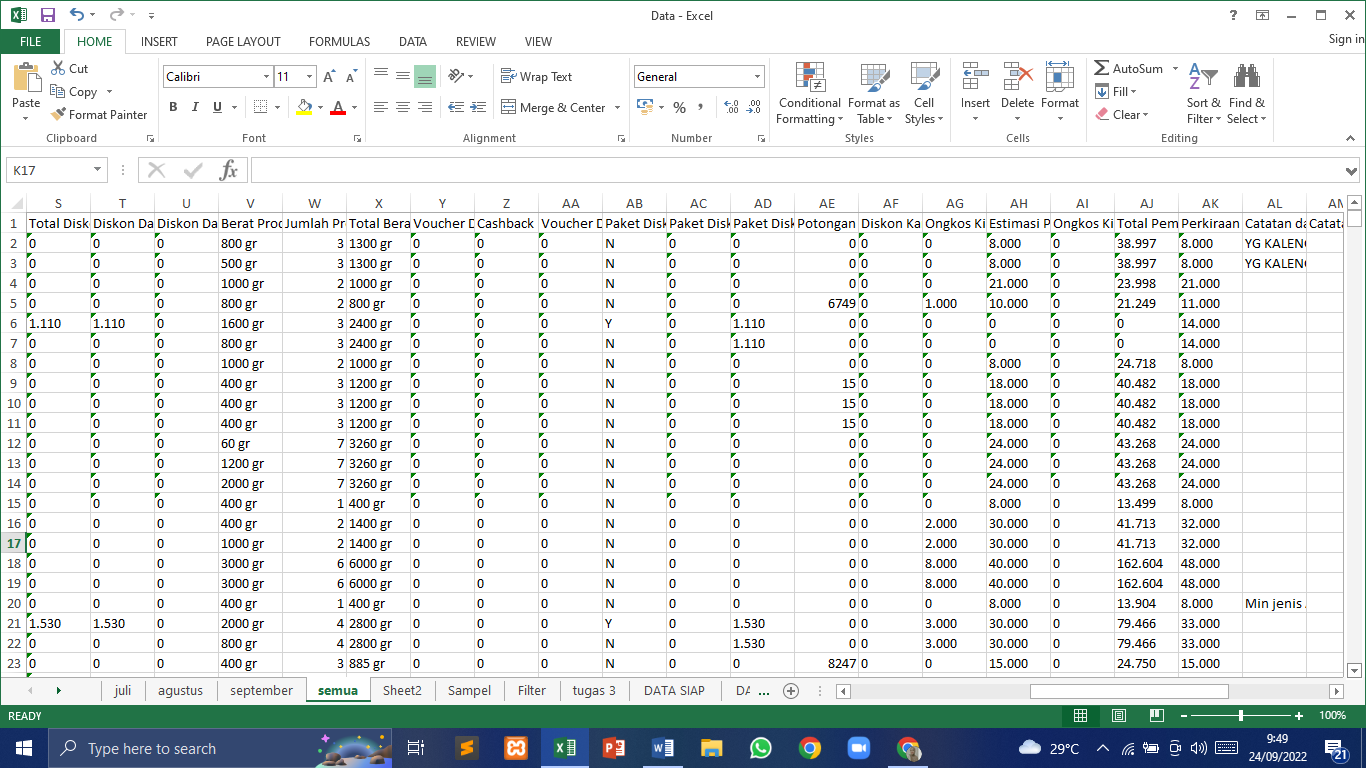
import numpy as np #komputasi numerik

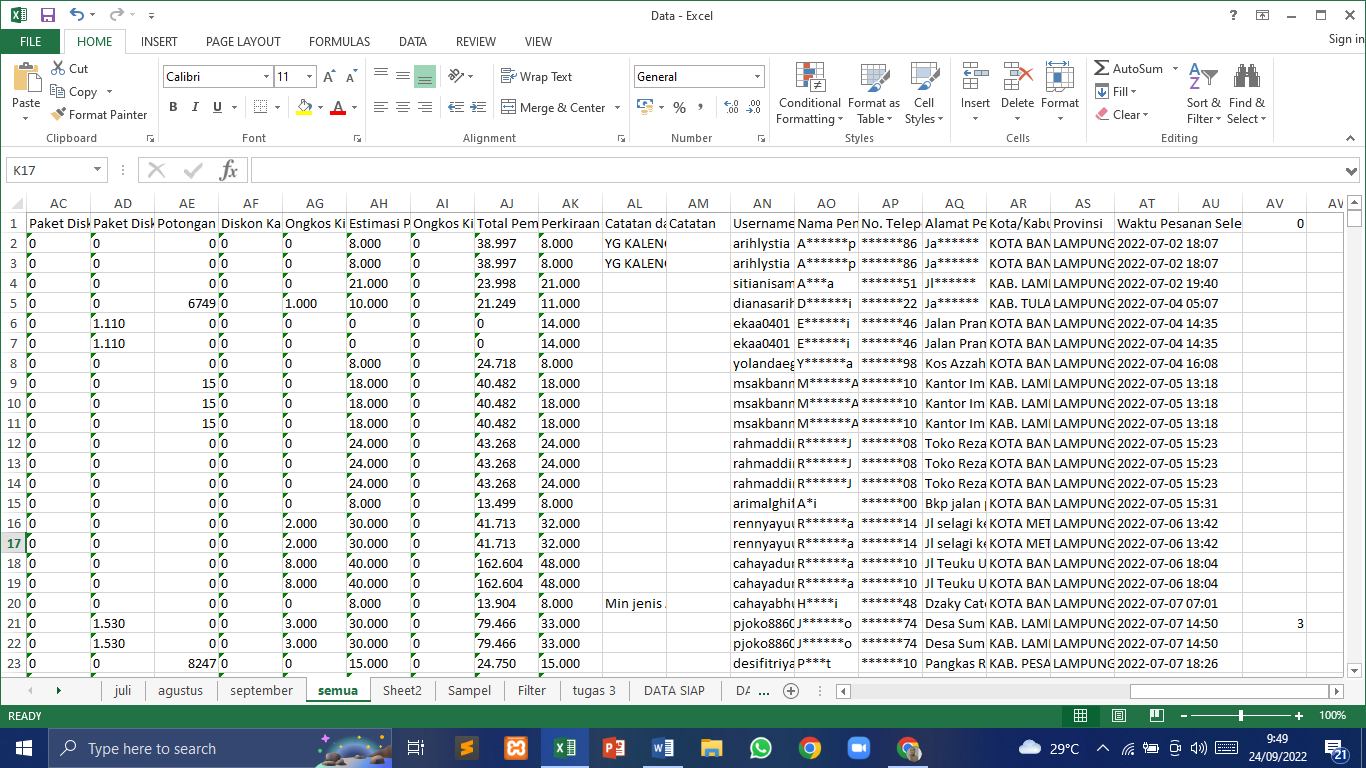
import pandas as pd #membaca data excel

from mlxtend.frequent\_patterns import apriori, association\_rules #import fungsi apriori dan association rules

1. Upload Data Excel

Proses selanjutnya yaitu mengimport data transaksi mentah dalam bentuk excel belum tertruktur untuk selanjutnya di proses agar berubah menjadi data yang terstruktur. Data mentah dapat dilihat pada gambar berikut.

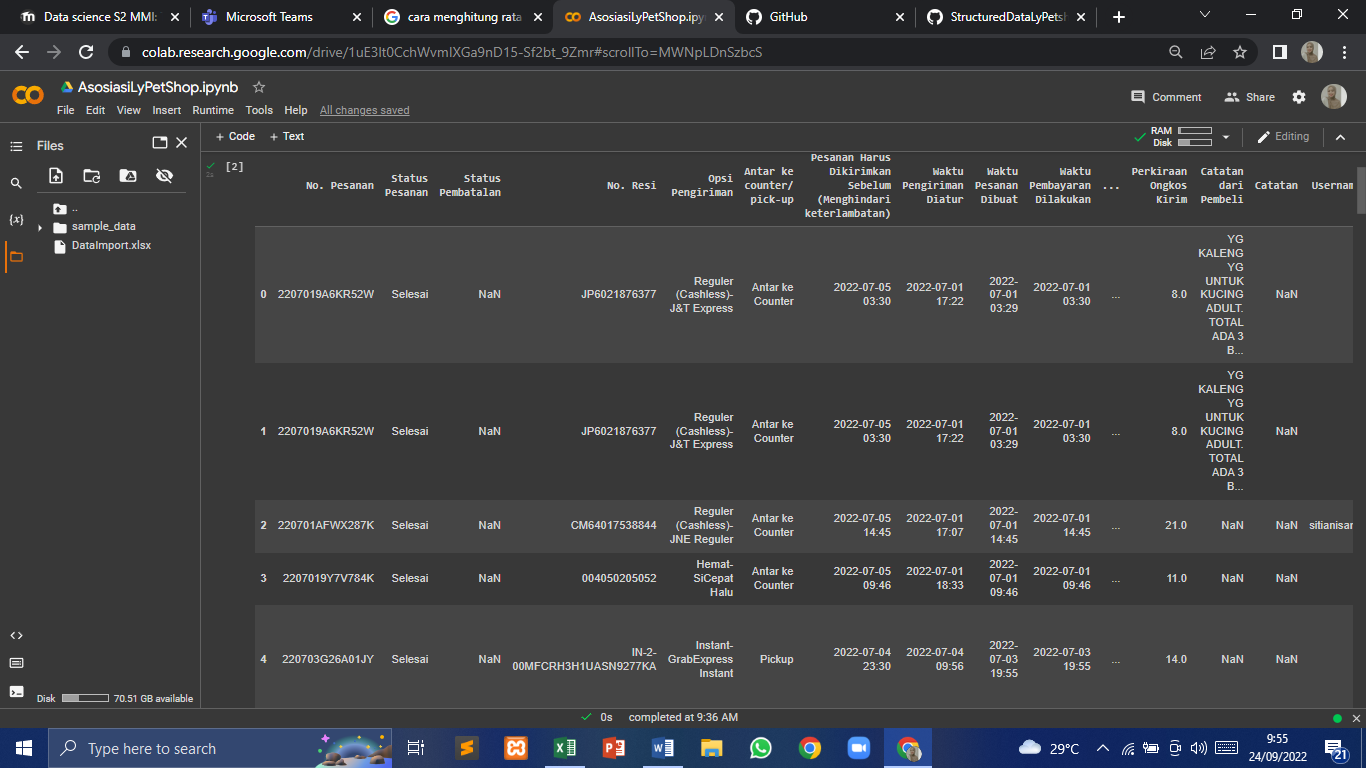


Selanjutnya data mentah tersebut diimport ke dalam text editor dengan menggunakan code berikut ini :

#Langkah 2 (Meload dan Mengeksplor Data / Memuat Data)

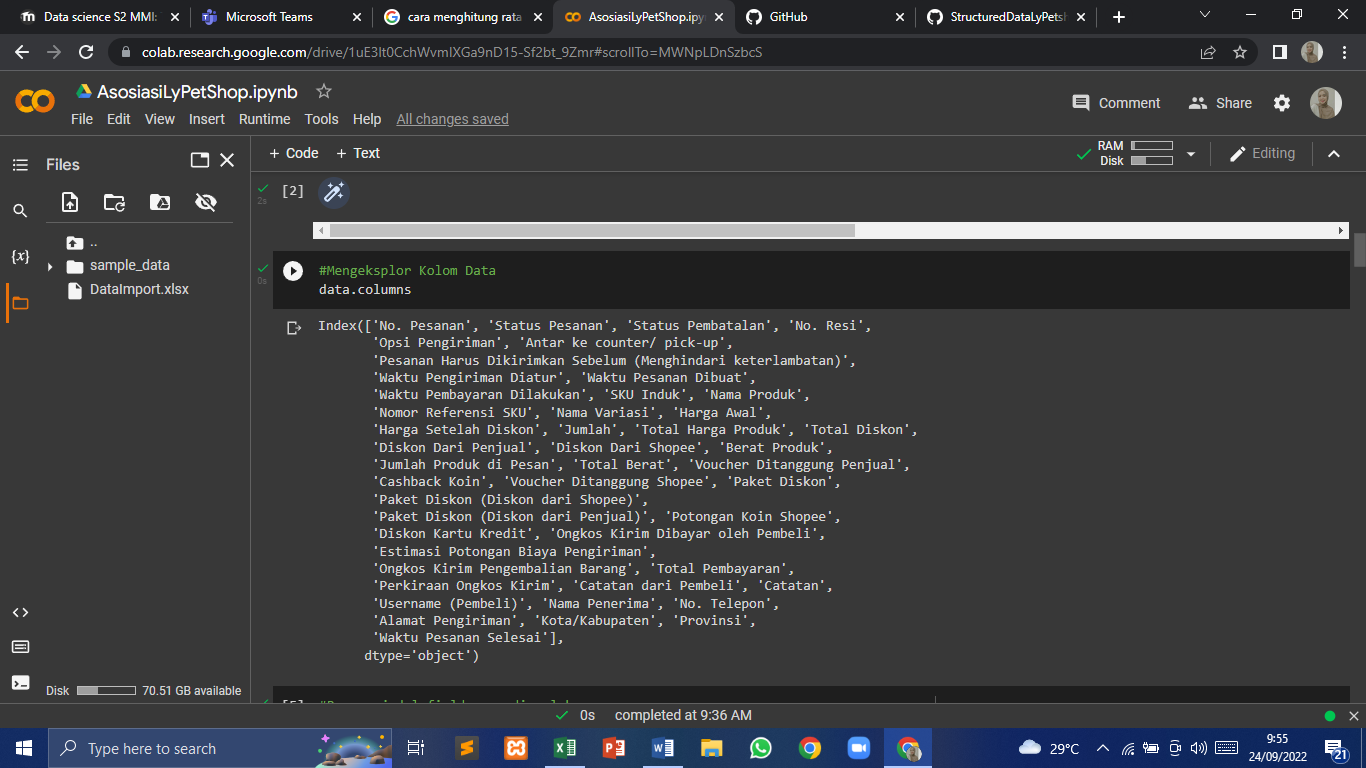
data = pd.read\_excel("DataImport.xlsx")

data.head()



#Mengeksplor Kolom Data

data.columns



1. Rename Field/Kolom

Untuk mempermudah proses selanjutnya maka saya mengubah nama masing-masing field yang akan digunakan. Seperti No. Pesanan dirubah menjadi Invoice, Nama Produk menjadi Description, Harga awal menjadi Price, Username (Pembeli) menjadi UserID dan Waktu Pesanan Selesai menjadi InvoiceDate.

#Rename judul field yang diperlukan

data.rename(columns = {'No. Pesanan' : 'Invoice'}, inplace = True)

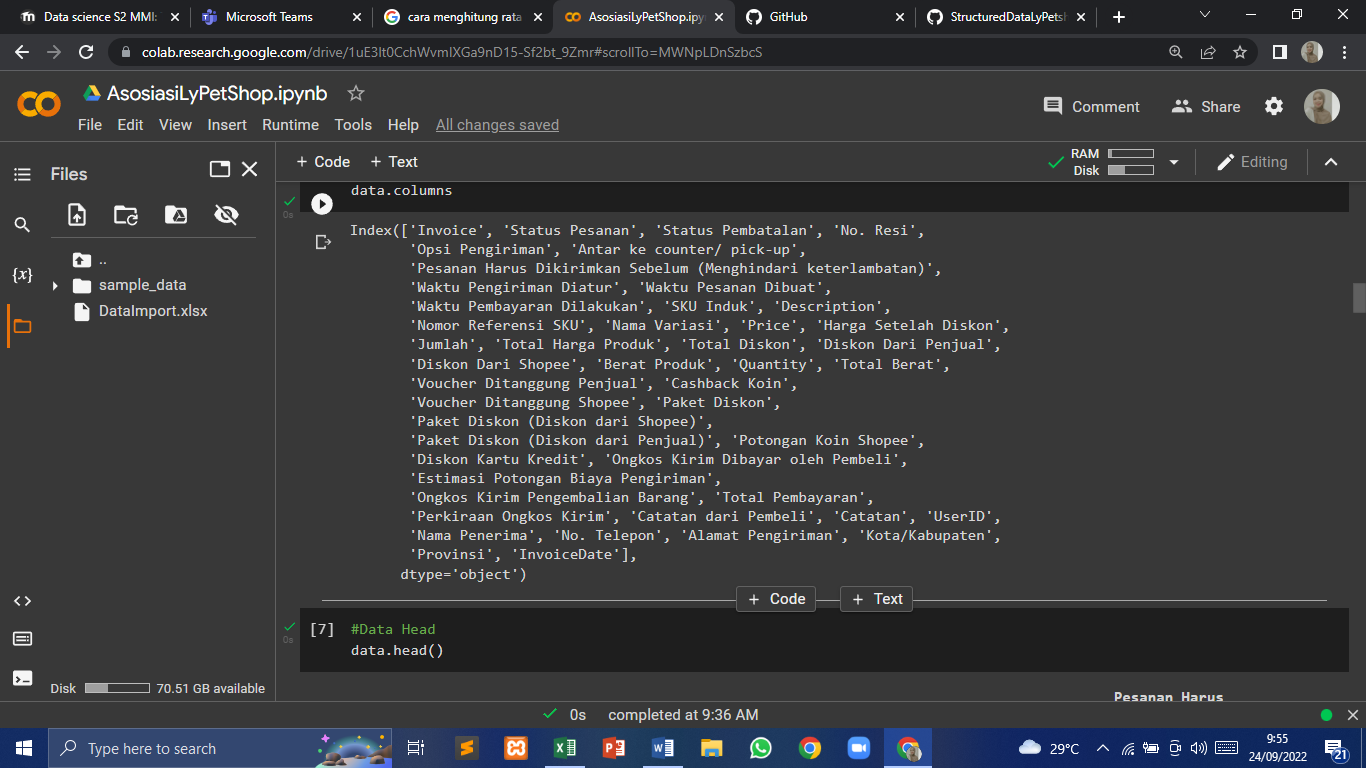
data.rename(columns = {'Nama Produk' : 'Description'}, inplace = True)

data.rename(columns = {'Harga Awal' : 'Price'}, inplace = True)

data.rename(columns = {'Jumlah Produk di Pesan' : 'Quantity'}, inplace = True)

data.rename(columns = {'Username (Pembeli)' : 'UserID'}, inplace = True)

data.rename(columns = {'Waktu Pesanan Selesai' : 'InvoiceDate'}, inplace = True)



Selanjutnya untuk mengetahui jumlah record yang kosong pada masing masing kolom dapat menggunakan perintah berikut.

#Melihat Data yang kosong pada record

print(data.isnull().sum())

Berikut merupakan total record yang kosong pada masing-masing kolom, untuk invoice = 0, itu artinya tidak ada record yang kosong atau data terisi secara keseluruhan, sedangkan pada status pembatalan = 1547, maka ada 1547 record data yang kosong atau NULL.

Invoice 0

Status Pesanan 0

Status Pembatalan 1547

No. Resi 0

Opsi Pengiriman 0

Antar ke counter/ pick-up 0

Pesanan Harus Dikirimkan Sebelum (Menghindari keterlambatan) 0

Waktu Pengiriman Diatur 2

Waktu Pesanan Dibuat 0

Waktu Pembayaran Dilakukan 0

SKU Induk 1547

Description 0

Nomor Referensi SKU 1547

Nama Variasi 446

Price 0

Harga Setelah Diskon 0

Jumlah 0

Total Harga Produk 0

Total Diskon 0

Diskon Dari Penjual 0

Diskon Dari Shopee 0

Berat Produk 0

Quantity 0

Total Berat 0

Voucher Ditanggung Penjual 0

Cashback Koin 0

Voucher Ditanggung Shopee 0

Paket Diskon 0

Paket Diskon (Diskon dari Shopee) 0

Paket Diskon (Diskon dari Penjual) 0

Potongan Koin Shopee 0

Diskon Kartu Kredit 0

Ongkos Kirim Dibayar oleh Pembeli 0

Estimasi Potongan Biaya Pengiriman 0

Ongkos Kirim Pengembalian Barang 0

Total Pembayaran 0

Perkiraan Ongkos Kirim 0

Catatan dari Pembeli 1452

Catatan 1547

UserID 0

Nama Penerima 0

No. Telepon 0

Alamat Pengiriman 0

Kota/Kabupaten 0

Provinsi 0

InvoiceDate 0

1. Menghapus Field Tidak Diperlukan

Dikarenakan hanya beberapa kolom saja yang diperlukan untuk proses selanjutnya, disini saya mengambil kolom invoice, description, price, quantity , customerID, dan invoice date. Maka, selain daripada kolom tersebut dihapus menggunakan fungsi drop.

#Hapus Kolom yang tidak diperlukan

data.drop('Status Pesanan', axis = 1, inplace = True)

data.drop('Status Pembatalan', axis = 1, inplace = True)

data.drop('No. Resi', axis = 1, inplace = True)

data.drop('Opsi Pengiriman', axis = 1, inplace = True)

data.drop('Antar ke counter/ pick-up', axis = 1, inplace = True)

data.drop('Pesanan Harus Dikirimkan Sebelum (Menghindari keterlambatan)', axis = 1, inplace = True)

data.drop('Waktu Pengiriman Diatur', axis = 1, inplace = True)

data.drop('Waktu Pesanan Dibuat', axis = 1, inplace = True)

data.drop('Waktu Pembayaran Dilakukan', axis = 1, inplace = True)

data.drop('SKU Induk', axis = 1, inplace = True)

data.drop('Nomor Referensi SKU', axis = 1, inplace = True)

data.drop('Nama Variasi', axis = 1, inplace = True)

data.drop('Harga Setelah Diskon', axis = 1, inplace = True)

data.drop('Jumlah', axis = 1, inplace = True)

data.drop('Total Harga Produk', axis = 1, inplace = True)

data.drop('Total Diskon', axis = 1, inplace = True)

data.drop('Diskon Dari Penjual', axis = 1, inplace = True)

data.drop('Diskon Dari Shopee', axis = 1, inplace = True)

data.drop('Berat Produk', axis = 1, inplace = True)

data.drop('Total Berat', axis = 1, inplace = True)

data.drop('Voucher Ditanggung Penjual', axis = 1, inplace = True)

data.drop('Cashback Koin', axis = 1, inplace = True)

data.drop('Voucher Ditanggung Shopee', axis = 1, inplace = True)

data.drop('Paket Diskon', axis = 1, inplace = True)

data.drop('Paket Diskon (Diskon dari Shopee)', axis = 1, inplace = True)

data.drop('Paket Diskon (Diskon dari Penjual)', axis = 1, inplace = True)

data.drop('Potongan Koin Shopee', axis = 1, inplace = True)

data.drop('Diskon Kartu Kredit', axis = 1, inplace = True)

data.drop('Ongkos Kirim Dibayar oleh Pembeli', axis = 1, inplace = True)

data.drop('Catatan', axis = 1, inplace = True)

data.drop('Nama Penerima', axis = 1, inplace = True)

data.drop('No. Telepon', axis = 1, inplace = True)

data.drop('Alamat Pengiriman', axis = 1, inplace = True)

data.drop('Provinsi', axis = 1, inplace = True)

data.drop('Estimasi Potongan Biaya Pengiriman', axis = 1, inplace = True)

data.drop('Ongkos Kirim Pengembalian Barang', axis = 1, inplace = True)

data.drop('Total Pembayaran', axis = 1, inplace = True)

data.drop('Perkiraan Ongkos Kirim', axis = 1, inplace = True)

data.drop('Catatan dari Pembeli', axis = 1, inplace = True)

data.drop('Kota/Kabupaten', axis = 1, inplace = True)

Selanjutnya mengecek kembali jumlah record yang kosong pada masing-masing kolom.

#Melihat Data yang kosong pada record

print(data.isnull().sum())

Berikut merupakan total record yang kosong pada masing-masing kolom yang sudah final. Dimana dapat dilihat bahwa ada enam kolom yang saya gunakan dan masing-masing sudah bernilai 0. Maka, sudah tidak ada lagi data dengan record yang kosong atau NULL.

Invoice 0

Description 0

Price 0

Quantity 0

CustomerID 0

InvoiceDate 0

1. Membersihkan Data

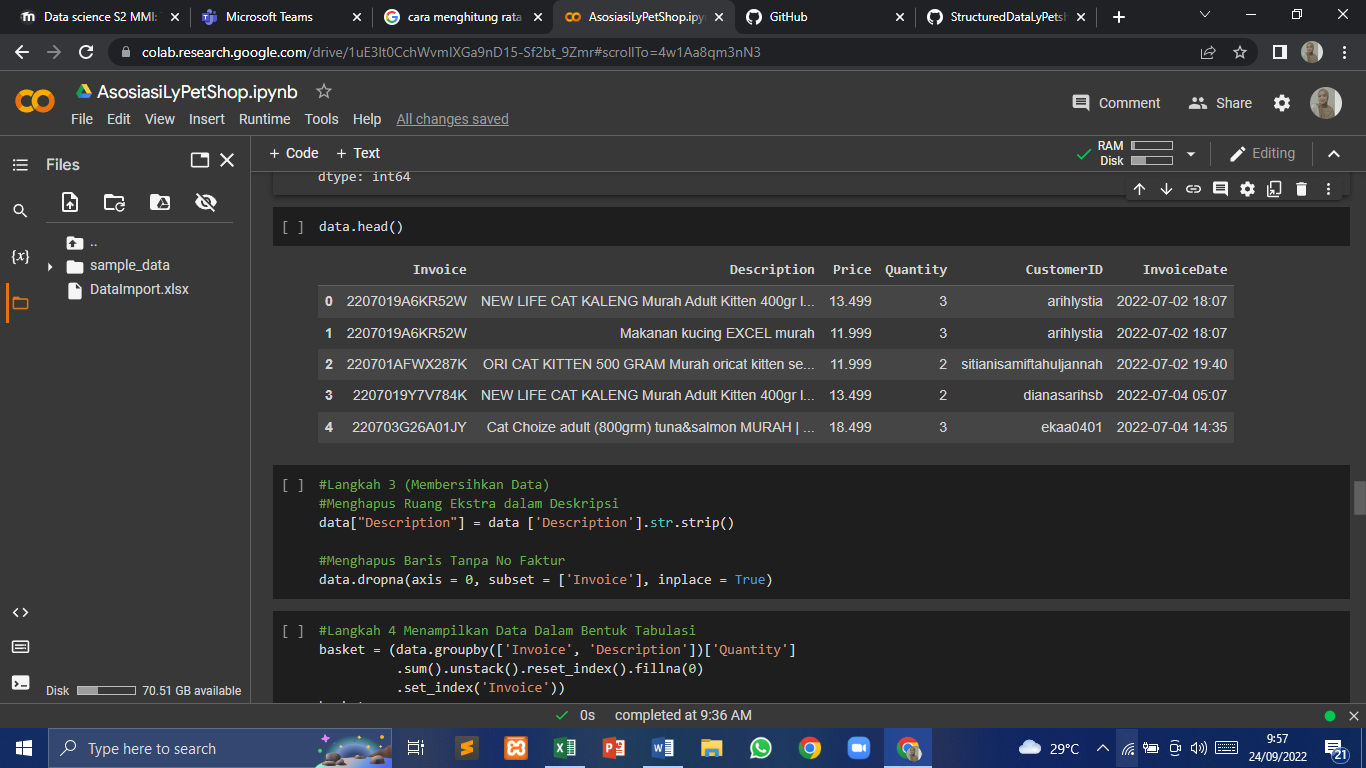
#Langkah 3 (Membersihkan Data)

#Menghapus Ruang Ekstra dalam Deskripsi

data["Description"] = data ['Description'].str.strip()

#Menghapus Baris Tanpa No Faktur

data.dropna(axis = 0, subset = ['Invoice'], inplace = True)



1. Data Tabulasi (jumlah terjual)

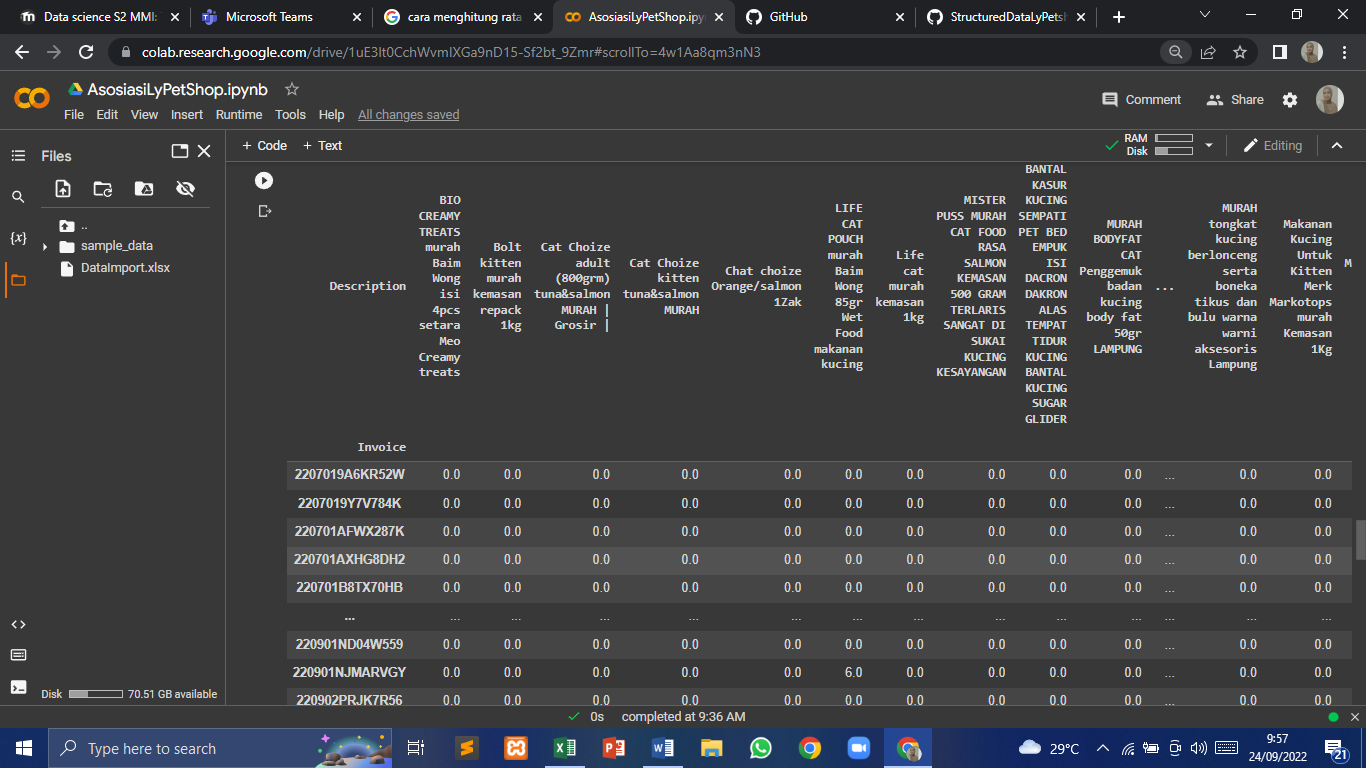
#Langkah 4 Menampilkan Data Dalam Bentuk Tabulasi

basket = (data.groupby(['Invoice', 'Description'])['Quantity']

          .sum().unstack().reset\_index().fillna(0)

          .set\_index('Invoice'))

basket



1. Tabulasi (0 dan 1)

#Langkah 5 (Pengkodean Data dengan Hot Encoding)

#Mendefinisikan Hot Encoding untuk Membuat Data Sesuai dengan Library yang Digunakan

def hot\_encode(x):

  if(x<=0):

    return 0

  if(x>=1):

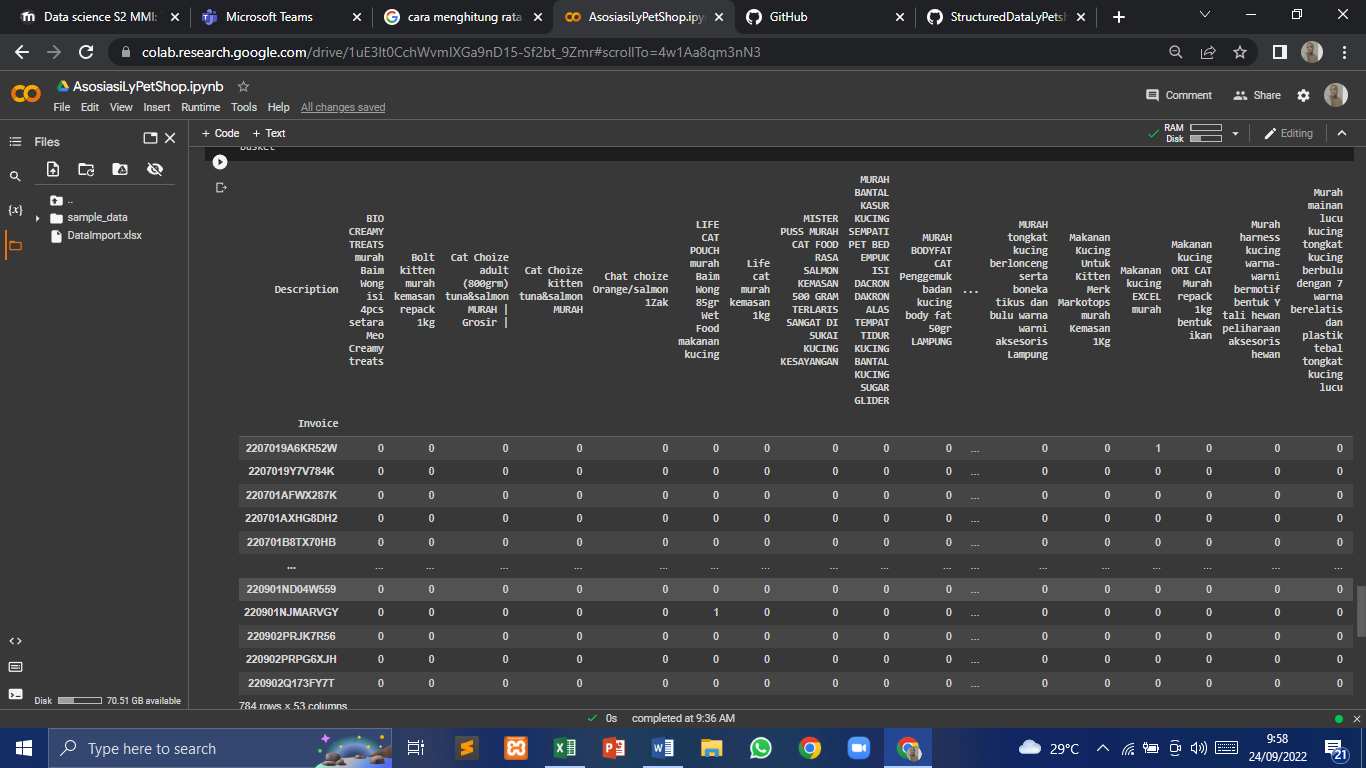
    return 1

#Pengkodean Dataset

basket\_encoded = basket.applymap(hot\_encode)

basket = basket\_encoded

basket



<https://github.com/fshellya/StructuredDataLyPetshop>

1. **Evaluasi**
2. Algoritma Apriori

Implementasi algoritma apriori untuk mengetahui barang apa saja yang sering dibeli dalam waktu yang bersamaan.

#Langkah 6 (Membangun Model dan Menganalisis Hasilnya)

#Membangun Model

frq\_items = apriori(basket, min\_support = 0.001, use\_colnames = True)

#Mengumpulkan aturan yang disimpulkan dalam dataframe

rules = association\_rules(frq\_items, metric ="lift", min\_threshold = 1)

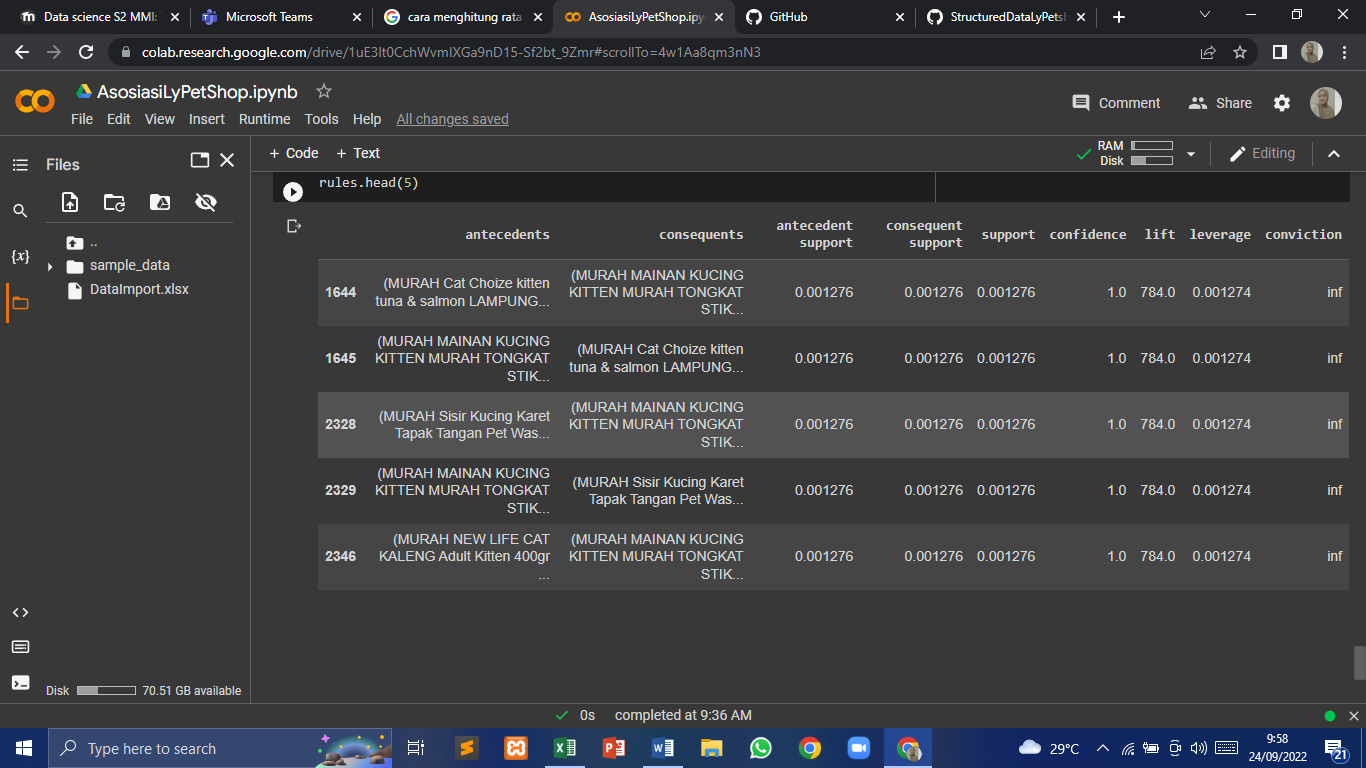
rules = rules.sort\_values(['confidence', 'lift'], ascending = [False, False])

#print (rules.head)

rules.head(5)

1. Hasil

Setelah melakukan pembersihan data dan implementasi algoritma apriori maka didapatkan hasil aturan asosiasi sebagai berikut.



Berdasarkan hasil dari penerapan algoritma apriori diatas dapat dilihat bahwa saya menggunakan minimal support 2 0.001 untuk menentukan 5 barang teratas yang paling banyak dibeli secara bersamaan dengan masing masing hasil sebagia berikut :

1. Apabila customer membeli Catchoize kitten maka akan membeli juga tongkat mainan kucing dengan nilai support 0.001276 dan nilai confidence 1 (100%).
2. Apabila customer membeli tongkat mainan kucing maka akan membeli juga cat choize kitten nilai support 0.001276 dan nilai confidence 1 (100%).
3. Apabila customer membeli sisir kucing maka akan membeli juga tongkat mainan kucing dengan nilai support 0.001276 dan nilai confidence 1 (100%).
4. Apabila customer membeli tongkat mainan kucing maka akan membeli juga sisir kucing dengan nilai support 0.001276 dan nilai confidence 1 (100%).
5. Apabila customer membeli Life cat kaleng adult maka akan membeli juga tongkat mainan kucing dengan nilai support 0.001276 dan nilai confidence 1 (100%).