Data Validity, Data Completeness, Data Consistency

Imports required :

```
import configparser
import pandas as pd
from sqlalchemy import create_engine
import mysql.connector as msql
from mysql.connector import Error
import csv,sys
import sqlite3
import re
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

Data Validity :

Verifying if the data is valid or not by verifying if the primary key is unique or not

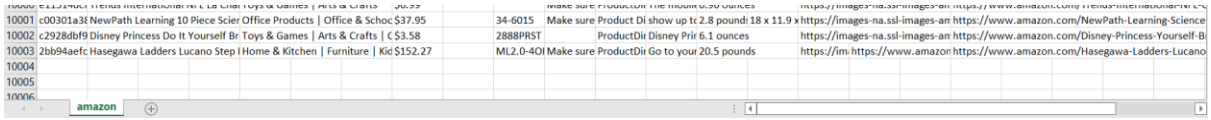
```
In [74]: walmart_df.nunique()

Out[74]: Uniq_Id              29988
         Crawl_Timestamp      25424
         Product_Url          29988
         Product_Name         29687
         Sale_Price            7599
         Brand                10755
         Item_Number           8872
         Gtin                 29988
         Product_inovation        0
         Category              3115
         Available                2
         dtype: int64
```

Audit Data Completeness

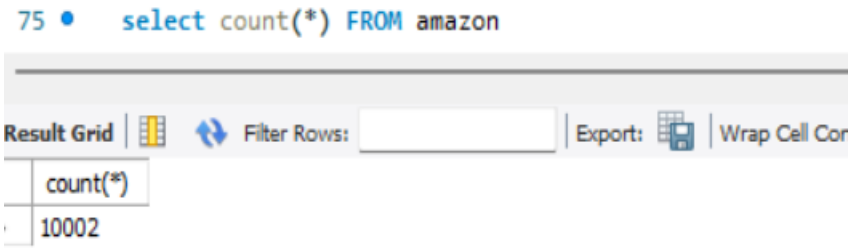
Comparing count before and after inserting data to table

Count in csv file –> amazon.csv

10003 – 1(header) = 10002

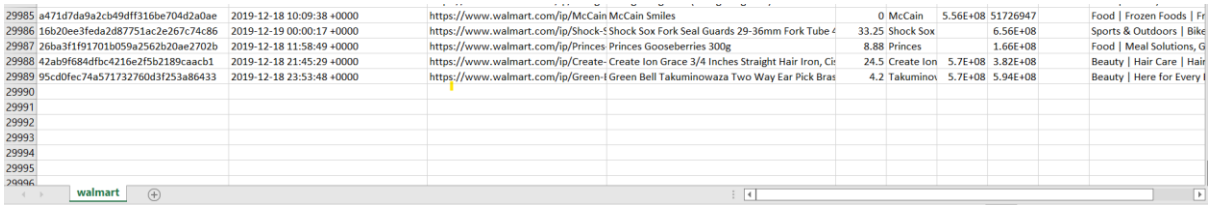


Count of data after inserting -> 10002



Count in csv file –> Walmart.csv

29989 – 1(header) = 29988



29985	a471d7da9a2cb49dff316be704d2a0ae	2019-12-18 10:09:38 +0000	https://www.walmart.com/ip/McCain	McCain Smiles	0	McCain	5.56E+08	51726947		Food | Frozen Foods | Fr
29986	16b20ee3feda2d87751ac2e267c74c86	2019-12-19 00:00:17 +0000	https://www.walmart.com/ip/Shock-S	Shock Sox Fork Seal Guards 29-36mm Fork Tube	33.25	Shock Sox		6.56E+08		Sports & Outdoors | Bike
29987	26ba3f1f91701b059a2562b20ae2702b	2019-12-18 11:58:49 +0000	https://www.walmart.com/ip/Princes	Princes Gooseberries 300g	8.88	Princes		1.66E+08		Food | Meal Solutions, G
29988	42ab9f684dfbc4216e2f5b2189caacb1	2019-12-18 21:45:29 +0000	https://www.walmart.com/ip/Create-	Create Ion Grace 3/4 Inches Straight Hair Iron, Ci	24.5	Create Ion	5.7E+08	3.82E+08		Beauty | Hair Care | Hair
29989	95cd0fec74a571732760d3f253a86433	2019-12-18 23:53:48 +0000	https://www.walmart.com/ip/Green-I	Green Bell Takuminowaza Two Way Ear Pick Bras	4.2	Takuminov	5.7E+08	5.94E+08		Beauty | Here for Every
29990										
29991										
29992										
29993										
29994										
29995										
29996										

walmart

Walmart

Count of data after inserting -> 29988

```
75 •    select count(*) FROM walmartdata
```

esult Grid | Filter Rows: | Export:

count(*)
29988

Audit Completeness –

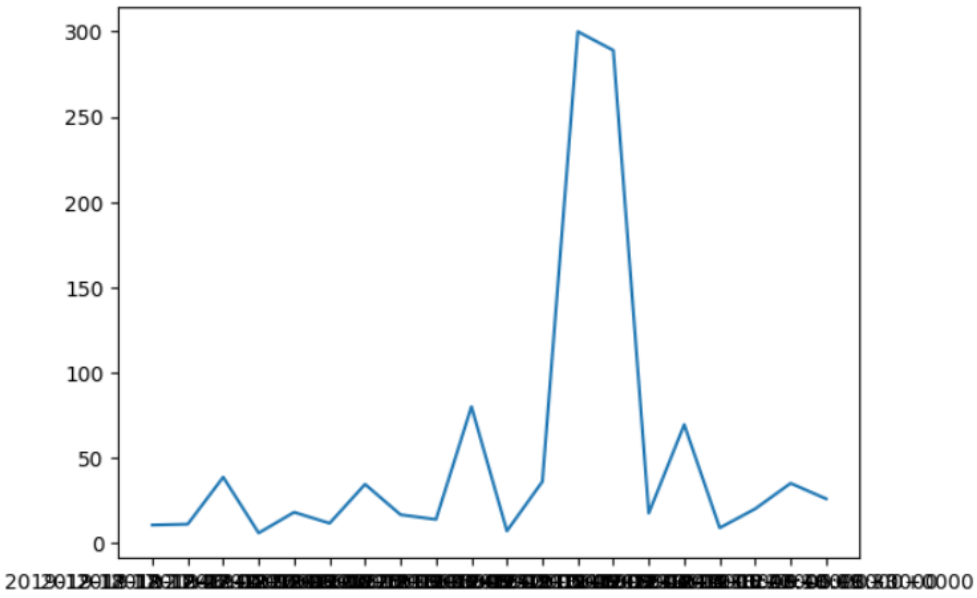
- consistancy = pd.read_csv('walmart.csv')

- plt.figure()

  con = consistancy.loc[1:20, ['Sale_Price', 'Crawl_Timestamp']]

  plt.plot(con.Crawl_Timestamp,con.Sale_Price)

```
In [210]: plt.figure()
          con = consistancy.loc[1:20, ['Sale_Price', 'Crawl_Timestamp']]
          plt.plot(con.Crawl_Timestamp,con.Sale_Price)

Out[210]: [<matplotlib.lines.Line2D at 0x207c3bd12a0>]
```



Comparing prices in US and India for Iphone-

- Code part –

  test = pd.read_csv('apple product price list from 26 countries.csv')

  us = test[test.country == 'United States']

  india = test[test.country == 'India']

In [ ]:
```
plt.plot(con.Gtin,con.Sale_Price)
```

[248]:
```
num = re.findall(r'\d+', str(us.price))
```

[256]:
```
num2 = re.findall(r'\d+', str(india.price))
```

[272]:
```
x= [num[0],num[1],num[2],num[3],num[4],num[5],num[6],num[7],num[8],num[9]]
y = [num2[0],num2[1],num2[2],num2[3],num2[4],num2[5],num2[6],num2[7],num2[8],num2[9]]
```

[270]:
```
print(x)
```

```
['431', '699', '29', '12', '24', '432', '399', '16', '62', '24']
```

In [ ]:

[249]:
```
print(num)
```

```
['431', '699', '29', '12', '24', '432', '399', '16', '62', '24', '433', '599', '24', '95', '24', '434', '179', '29', '83', '6',
'435', '249', '41', '50', '6', '436', '129', '21', '50', '6', '437', '549', '91', '50', '6', '438', '179', '00', '29', '83',
'6', '439', '279', '11', '62', '24', '440', '199', '8', '29', '24', '441', '49', '4', '08', '12', '442', '329', '27', '41', '1
2', '443', '799', '66', '58', '12', '444', '129', '10', '75', '12', '445', '1299', '108', '25', '12', '446', '999', '83', '25',
'12', '447', '79']
```

[257]:
```
print(num2)
```

In [74]:
```
walmart_df.nunique()
```

Out[74]:
```
Uniq_Id              29988
Crawl_Timestamp      25424
Product_Url          29988
Product_Name         29687
Sale_Price            7599
Brand                10755
Item_Number           8872
Gtin                 29988
Product_inovation        0
Category              3115
Available                2
dtype: int64
```

In [12]:
```
amazon_df.count()
```

Out[12]:
```
Uniq_Id                  10002
Product_Name             10002
Brand_Name                   0
Asin                         0
Category                  9172
Upc_Ean_Code                34
List_Price                   0
Selling_Price             9895
Quantity                     0
Model_Number              8232
About_Product             9729
Product_Specification     8370
Shipping_Weight           8864
Product_Dimensions         479
Variants                  2478
Sku                          0
```

In [10]:
```
amazon_df.isnull().sum()
```

Out[10]:
```
Uniq_Id                      0
Product_Name                 0
Brand_Name               10002
Asin                     10002
Category                   830
Upc_Ean_Code              9968
List_Price               10002
Selling_Price              107
Quantity                 10002
Model_Number              1770
About_Product              273
Product_Specification     1632
Shipping_Weight           1138
Product_Dimensions        9523
Variants                  7524
Sku                      10002
Product_Url                  0
Stock                    10002
Product_Details          10002
Dimensions               10002
Color                    10002
Ingredients              10002
Direction_To_Use         10002
Is_Amazon_Seller             0
Size_Quantity_Variant    10002
Product_Description      10002
dtype: int64
```