



# WIREFRAME DOCUMENT

Amazon Sales Data Analysis

BY



VIKASH KUMAR MAHAPATRA

Revision Number & Date :- 1.3 27/09/2022

## Document Version Control

Date Issued	Version	Description	Author
20 SEPT 2022	1.0	Created Document And Add Document Version Control	Vikash
21 SEPT 2022	1.1	Created Content Page ,Introduction Page & Dataset Information Page	Vikash
26 SEPT 2022	1.2	Created Exploratory Analysis, Power-Bi dashboard, Add Page Number in Content Section	Vikash
27 SEPT 2022	1.3	Final Revision	Vikash

# CONTENTS

ABSTRACT	3
INTRODUCTION	4
 WHY THIS WIREFRAME DOCUMENT	
 PROBLEM STATEMENT	
DATASET INFORMATION	5-6
EXPLORATORY ANALYSIS	6-13
POWER-BI DASHBOARD	13-14

## ABSTRACT

Organizations under the E-commerce industry seek to attain core competence by creating and sustaining a unique process to collect personal information about customers and their purchasing trends. The report critically evaluates how a service-based organization Amazon uses management information systems as a vibrant tool in attaining competitive advantage through efficient management and acquisition of information. As in today's market without proper sales management, it's very hard to predict how the business is running and how it will be in future. Many companies with proper sales management have shown better growth as they already know which item they have to focus on, which product needs some improvement etc. Sales management helps in maintaining its customer base for a longer time by providing them attractive offers, as they already have the information's like who are their top customers, whom they have to focus on etc. Sales management so helps in minimizing the losses. Also, the competition is increasing day by day as many new companies are coming with better management systems and giving tough competition due to that it is now very important to have a proper sales management to run any business and to compete with these companies.

## INTRODUCTION

### WHY THIS WIREFRAME DOCUMENT

This document represent the complex tasks involved in a project in an easy-to-understand visual format. By these we can also communicate any changes to all stakeholders quickly and efficiently. This also assist in keeping project on track and help to reduce misunderstanding between stakeholders.

### PROBLEM STATEMENT

The problem statement aims to analyze amazon sales from 2017-2019 to know more in depth information about the items that are in high demand, high profit so that ways are find out to improve methods of distribution to reduce cost ,stay ahead in the competition, have progressive growth as well as further increase in the sales. As sale management today is the most important function in a commercial and business enterprise. To achieve the goal, we used a data set that is given and analyze most important parameter that are responsible for it. Also get some meaningful insights from the given information regarding Sales Data. Find key metrics and factors and show the meaningful relationship between attributes. Do your own research and come up with findings.

## DATASET INFORMATION

The file name salesdata.xls contains different Sales parameters data of the selective items. Sample data below.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	P	Q	R	S	T
1	CustKey	DateKey	Discount	Invoice Da	Invoice Numbr	Item Class	Item Nr	Item	Line Numbr	List Price	Order Number	Promised Deliv	Sales Amount	Sales A	Sales Margin A	Sales Price	Sales	Sales Rep	U/M
2	1000481	30-04-17	-237.91	2017-04-30	100012			Urban Large Eggs	2000	0	200015	30-04-17	237.91	0	237.91	237.91	1	184	EA
3	10002220	14-07-17	368.79	2017-07-14	100233	P01	20910	Moms Sliced Turkey	1000	824.96	200245	14-07-17	458.17	824.96	458.17	458.17	1	127	EA
4	10002220	17-10-17	109.73	2017-10-17	116165	P01	38076	Cutting Edge Foot-Long Hot Dogs	1000	548.66	213157	16-10-17	438.93	548.66	438.93	438.93	1	127	EA
5	10002489	03-06-17	-211.75	2017-06-03	100036			Kiwilox	1000	0	200107	03-06-17	211.75	0	211.75	211.75	1	160	EA
6	10004516	27-05-17	96627.9	2017-05-27	103341	P01	60776	High Top Sweet Onion	1000	408.52	203785	28-05-17	89248.66	185877	89248.66	196.1509	455	124	SE
7	10004516	30-05-17	-1950	2017-05-30	103610			Best Choice Fudge Brownies	2000	0	203785	30-05-17	1950	0	1950	1950	1	124	EA
8	10007866	03-09-17	371.014	2017-09-03	100403	P01	20910	Moms Sliced Turkey	2000	795.314	200436	03-09-17	424.3	795.314	424.3	424.3	1	149	EA
9	10009356	18-06-17	608.08	2017-06-18	105481	P01	62550	Tell Tale Garlic	29000	575	205213	18-06-17	541.92	1150	541.92	270.96	2	103	EA
10	10009356	18-06-17	424.8	2017-06-18	105481	P01	60794	High Top Walnuts	18000	51.88	205213	18-06-17	353.4	778.2	353.4	23.56	15	103	EA
11	10009356	18-06-17	13492.8	2017-06-18	105481	P01	36001	Big Time Frozen Cheese Pizza	9000	412.03	205213	18-06-17	11229	24721.8	11229	187.15	60	103	EA
12	10009356	18-06-17	10481.1	2017-06-18	105481	P01	38076	Cutting Edge Foot-Long Hot Dogs	13000	548.66	205213	18-06-17	8722	19203.1	8722	249.2	35	103	EA
13	10009356	18-06-17	404.147	2017-06-18	105481	P01	61484	Super Creamy Peanut Butter	37000	50.5051	205213	18-06-17	353.43	757.577	353.43	23.562	15	103	EA
14	10009606	16-09-17	1287.35	2017-09-16	100445	P01	17801	Better Fancy Canned Sardines	3000	1379.79	200478	16-09-17	1472.24	2759.59	1472.24	736.12	2	118	EA
15	10009606	16-09-17	4764.33	2017-09-16	100445	P01	48500	Red Spade Low Fat Cole Slaw	1000	1134.77	200478	16-09-17	5448.6	10212.9	5448.6	605.4	9	118	EA
16	10009633	10-06-17	-526.64	2017-06-10	104708		38631	Jumbo Large Eggs	1000	0	204575	06-06-17	526.64	0	526.64	526.64	1	127	EA
17	10009633	20-08-17	0	2017-08-20	100379	P01	60443	High Top Corn on the Cob	1000	346.53	200397	18-08-17	346.53	346.53	346.53	346.53	1	127	EA
18	10009645	17-11-17	0	2017-11-17	100684		61801	Moms Low Fat Bologna	2000	258.01	200733	14-11-17	3096.12	3096.12	3096.12	258.01	12	155	EA
19	10009652	14-05-17	91	2017-05-14	102256		63113	Landslide Columbian Coffee	1000	5.61	202389	14-05-17	1872.5	1963.5	1872.5	5.35	350	145	EA
20	10009652	14-05-17	-700	2017-05-14	102256			High Top Dried Mushrooms	2000	0	202389	14-05-17	700	0	700	2	350	145	EA
21	10009907	06-11-17	786.6	2017-11-06	117815	P01	38050	Gorilla Chocolate Milk	2000	301.23	200706	06-11-17	418.32	1204.92	418.32	104.58	4	167	EA
22	10010884	19-06-17	499.44	2017-06-19	105615	P01	38007	Gorilla Jack Cheese	1000	1103	205323	19-06-17	603.56	1103	603.56	603.56	1	105	EA
23	10010884	08-07-17	409.77	2017-07-08	107052	P01	28401	Ebony Prepared Salad	10000	968.44	205041	13-06-17	556.67	966.44	556.67	556.67	1	105	EA
24	10010884	08-07-17	186.43	2017-07-08	107052	P01	26361	Bravo Canned Yams	7000	439.7	205041	13-06-17	253.27	439.7	253.27	253.27	1	105	EA
25	10010884	08-07-17	349.78	2017-07-08	107052	P01	20910	Moms Sliced Turkey	4000	824.96	205041	13-06-17	475.18	824.96	475.18	475.18	1	105	EA
26	10010884	08-07-17	325.53	2017-07-08	107052	P01	45880	Red Spade Low Fat Bologna	17000	767.75	205041	13-06-17	442.22	767.75	442.22	442.22	1	105	EA
27	10010884	08-07-17	313.27	2017-07-08	107052	P01	29394	Pearl Chardonnay	12000	123.14	205041	13-06-17	425.57	738.84	425.57	70.92833	6	105	EA
28	10010884	08-07-17	174.7	2017-07-08	107052	P01	36001	Big Time Frozen Cheese Pizza	13000	412.03	205041	13-06-17	237.33	412.03	237.33	237.33	1	105	EA

## EXPLORATORY ANALYSIS

```
#Extracting CSV Data From System using Pandas Library
AmazonSales_Df = pd.read_excel('SALESDATA.xls')
```

```
# Shape of dataset i.e. no of rows and columns
AmazonSales_Df.shape

(65282, 20)
```

In the above figure we have load dataset read the data with help of pandas. We created a data frame name as AmazonSales\_Df. Also got shape of the dataset i.e. 65282\* 20.

```
#Statistics about the numerical columns in a dataset
AmazonSales_Df.describe().T
```

	count	mean	std	min	25%	50%	75%	max
CustKey	65282.0	1.001770e+07	7176.148147	10000453.00	1.001272e+07	1.001966e+07	1.002351e+07	10027583.00
Discount Amount	65280.0	1.855575e+03	9037.140888	-255820.80	2.460375e+02	4.417600e+02	9.997600e+02	343532.86
Invoice Number	65282.0	2.162237e+05	94892.281866	100012.00	1.179310e+05	2.228695e+05	3.143188e+05	332842.00
Line Number	65282.0	2.371385e+04	32664.024053	1000.00	3.000000e+03	1.200000e+04	3.200000e+04	344000.00
List Price	65282.0	5.146834e+02	449.189182	0.00	1.815800e+02	3.251900e+02	8.038600e+02	2760.70
Order Number	65282.0	1.805831e+05	67593.871116	100838.00	1.153210e+05	2.037020e+05	2.185760e+05	321532.00
Sales Amount	65282.0	2.852038e+03	15164.342107	200.01	3.083875e+02	5.539400e+02	1.280043e+03	555376.00
Sales Amount Based on List Price	65282.0	4.707474e+03	20696.443785	0.00	5.610400e+02	9.981600e+02	2.315040e+03	632610.16
Sales Cost Amount	65282.0	1.660979e+03	9556.485250	0.00	1.677900e+02	3.045000e+02	6.873200e+02	366576.00
Sales Margin Amount	65282.0	1.191059e+03	5860.787502	-3932.93	1.299500e+02	2.464900e+02	5.795300e+02	188800.00
Sales Price	65281.0	2.836159e+02	252.876719	-5000.00	1.000700e+02	1.837575e+02	4.482200e+02	6035.00
Sales Quantity	65282.0	4.508431e+01	429.661793	-1.00	2.000000e+00	3.000000e+00	8.000000e+00	16000.00
Sales Rep	65282.0	1.374224e+02	26.643936	103.00	1.130000e+02	1.340000e+02	1.600000e+02	185.00

```
#Data Types of Columns
AmazonSales_Df.dtypes
```

```
CustKey          int64
DateKey          datetime64[ns]
Discount Amount  float64
Invoice Date     datetime64[ns]
Invoice Number   int64
Item Class       object
Item Number      object
Item             object
Line Number      int64
List Price       float64
Order Number     int64
Promised Delivery Date datetime64[ns]
Sales Amount     float64
Sales Amount Based on List Price float64
Sales Cost Amount float64
Sales Margin Amount float64
Sales Price      float64
Sales Quantity   int64
Sales Rep        int64
U/M             object
dtype: object
```

In the above figure we read some statistical value of the dataset as well as datatypes present in the dataset. We have use of AmazonSales\_Df.describe() for statistical value and AmazonSales\_Df.dtypes for datatypes.

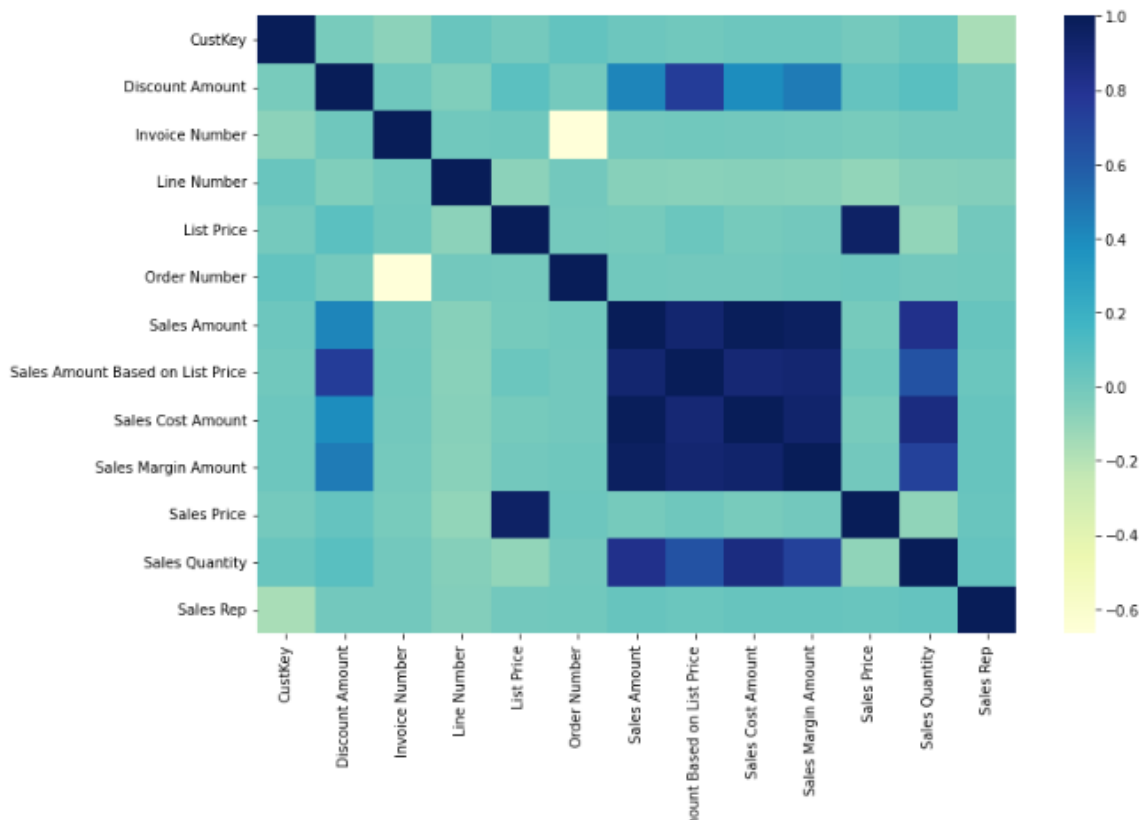
```
# Checking no. of Null Values
AmazonSales_Df.isnull().sum()
```

```
CustKey          0
DateKey          0
Discount Amount  2
Invoice Date     0
Invoice Number   0
Item Class       8289
Item Number      41
Item             0
Line Number      0
List Price       0
Order Number     0
Promised Delivery Date  0
Sales Amount     0
Sales Amount Based on List Price  0
Sales Cost Amount  0
Sales Margin Amount  0
Sales Price      1
Sales Quantity   0
Sales Rep        0
U/M             0
dtype: int64
```

In the above figure we find out if there any null values present in the dataset or not. We have use of AmazonSales\_Df.isna.sum() check null value.

```
# Checking the correlation
plt.figure(figsize=(12,8))
sns.heatmap(AmazonSales_Df.corr(method='pearson'),cmap='YlGnBu')
```

<AxesSubplot:>





In the above figure we find out correlation between variables or features of a dataset. We have use of AmazonSales\_Df.corr() so that any NaN values be automatically excluded & non-numeric data type or columns in the dataframe is ignored.

```
# Copying Dataset
AmazonSal_AnDf = AmazonSales_Df.copy()

AmazonSal_AnDf.dropna(subset=['Discount Amount', 'Sales Price', 'Item Number'], inplace=True)

AmazonSal_AnDf['Item class'].value_counts()

P01    56977
P01     16
Name: Item class, dtype: int64

AmazonSal_AnDf.groupby('Item').sum()['Sales Quantity']

Item
American Beef Bologna      230
American Chicken Hot Dogs  135
American Corned Beef      2722
American Pimento Loaf     216
American Potato Salad     200
...
Washington Diet Cola      826
Washington Diet Soda     1423
Washington Mango Drink    478
Washington Orange Juice   2104
Washington Strawberry Drink 134
Name: Sales Quantity, Length: 650, dtype: int64
```

In the above figure we have copied the dataset to new dataframe that is AmazonSal\_AnDf to make some necessary changes without messing around with original dataset. In second step in same figure we drop columns that nan values with AmazonSal\_AnDf.dropna. In third step we calculate columns and item class and item using AmazonSal\_AnDf['Item Class'].value\_counts() and AmazonSal\_AnDf.groupby('Item').sum()['Sales Quantity'].

```
# Creating Month, Year, Month, Quarter, Day Columns
AmazonSal_AnDf['Billed_Year'] = AmazonSal_AnDf['Invoice Date'].dt.year
AmazonSal_AnDf['Billed_Month_Year'] = AmazonSal_AnDf['Invoice Date'].dt.strftime('%y-%m')
AmazonSal_AnDf['Billed_Month'] = AmazonSal_AnDf['Invoice Date'].dt.month
AmazonSal_AnDf['Billed_Quarter'] = AmazonSal_AnDf['Invoice Date'].dt.quarter
AmazonSal_AnDf['Billed_Day'] = AmazonSal_AnDf['Invoice Date'].dt.day

# Only Required Columns to display
AmazonSal_AnDf = AmazonSal_AnDf[['CustKey', 'Item', 'Invoice Date', 'Billed_Year', 'Billed_Month_Year', 'Billed_Quarter', 'Billed_Month', 'Billed_Day', 'Sales Quantity', 'Sales Amount Based on List Price', 'Discount Amount', 'Sales Amount', 'Sales Margin Amount', 'Sales Cost Amount', 'Sales Rep', 'U/M', 'List Price', 'Sales Price']]

AmazonSal_AnDf.isnull().sum()

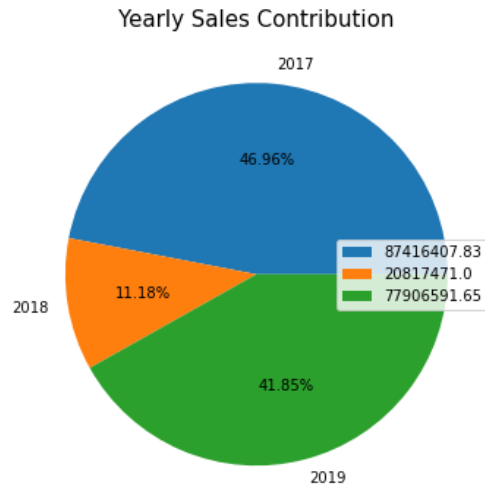
CustKey      0
Item          0
Invoice Date  0
Billed_Year   0
Billed_Month_Year  0
Billed_Quarter  0
Billed_Month  0
Billed_Day    0
Sales Quantity  0
Sales Amount Based on List Price  0
Discount Amount  0
Sales Amount  0
Sales Margin Amount  0
Sales Cost Amount  0
Sales Rep     0
U/M           0
List Price    0
Sales Price   0
dtype: int64
```

In the above figure we created new columns like month\_year, month, quarter, year, day and display only columns that are

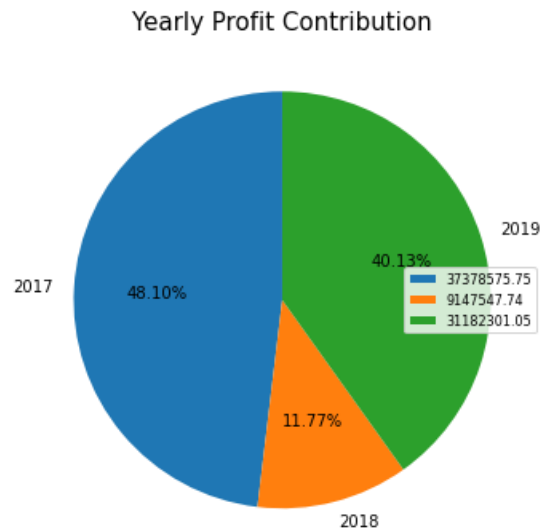
```
AmazonSal_AnDf.head(10)
```

	CustKey	Item	Invoice Date	Billed_Year	Billed_Month_Year	Billed_Quarter	Billed_Month	Billed_Day	Sales Quantity	Sales Amount Based on List Price	Discount Amount	Sales Amount
1	10002220	Moms Sliced Turkey	2017-07-14	2017	17-07	3	7	14	1	824.9600	368.7900	456.17
2	10002220	Cutting Edge Foot-Long Hot Dogs	2017-10-17	2017	17-10	4	10	17	1	548.6600	109.7300	438.93
4	10004516	High Top Sweet Onion	2017-05-27	2017	17-05	2	5	27	455	185876.6000	96627.9400	89248.66
6	10007866	Moms Sliced Turkey	2017-09-03	2017	17-09	3	9	3	1	795.3140	371.0140	424.30
7	10009356	Tell Tale Garlic	2017-06-18	2017	17-06	2	6	18	2	1150.0000	608.0800	541.92
8	10009356	High Top Walnuts	2017-06-18	2017	17-06	2	6	18	15	778.2000	424.8000	353.40
9	10009356	Big Time Frozen	2017-06-18	2017	17-06	2	6	18	60	24721.8000	13492.8000	11229.00

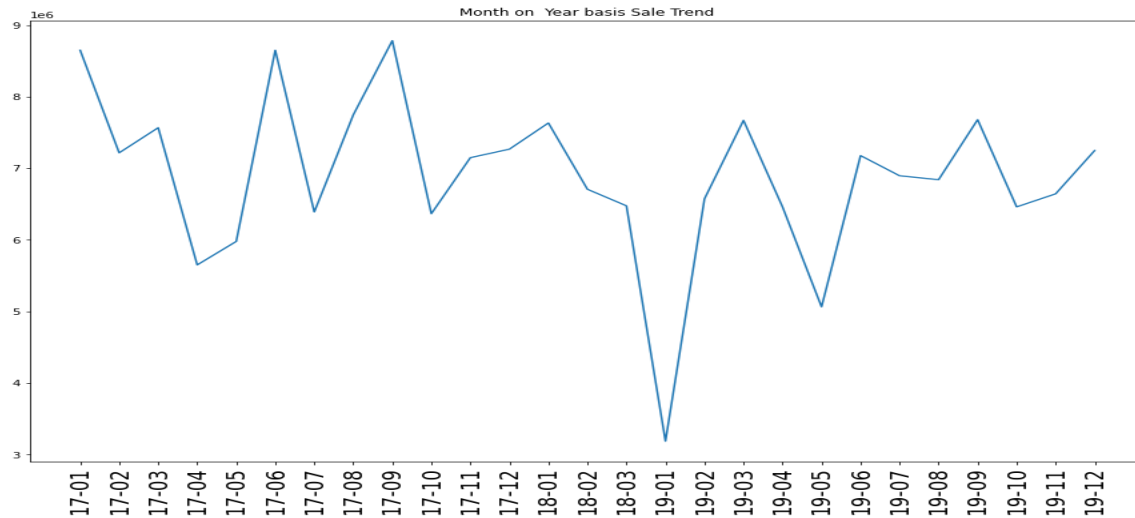
In the above figure we read the top ten records after modifying it with new dataframe that is AmazonSal\_AnDf.



In the above figure we can see that in the year 2017 have 46.98% of sales then second comes the year 2019 which have 41.85% of sales and least happen in the year 2018 with only 11.18%.



In the above figure we can see that in the year 2017 have 48.10% of profit then second comes the year 2019 which have 40.13% of profit and least happen in the year 2018 with only 11.71%.



From the above figure Highest Selling Month in the Year 2017 is Sept and Lowest Selling is April where as in year 2018 is Jan and Lowest Selling is March and lastly in the year 2019 Highest Selling Month is Sept and Lowest Selling is Jan.

Item	Billed_Month_Year	
Better Large Canned Shrimp	19-03	38546
	19-09	38458
	19-08	32617
	17-08	32613
	17-02	32613
	19-02	32529
High Top Dried Mushrooms	18-01	31340
	17-06	30140
	19-04	30087
Better Large Canned Shrimp	18-02	30056
	17-09	29964
	17-03	29964
	18-01	29868
Better Large Canned Shrimp	18-03	25212
	19-05	24208
	17-01	24036
Landslide Hot Chocolate	17-07	24031
	19-06	22684

In above figure showing max item billed in month year basis that is Better Large Canned Shrimp in Year 2019 and next item is High Top Dried Mushrooms in Year 2019.

Cutting Edge Pimento Loaf	19-12	1
Fabulous Berry Juice	17-06	1
	19-09	1
	19-03	1
Landslide Vegetable Oil	18-03	1
	17-09	1
	17-03	1
	19-05	1
Best Corn Puffs	17-05	1
Landslide Sesame Oil	19-01	1
	17-12	1
Excellent Strawberry Drink	17-11	1
Discover Spaghetti	19-01	1

In above figure showing least item billed in month year basis that is Landslide Vegetable Oil in Year 2019, Cutting Edge Pimento Leaf, Best Corn Puffs and Excellent Strawberry Drink around the year 2017 and 2019.

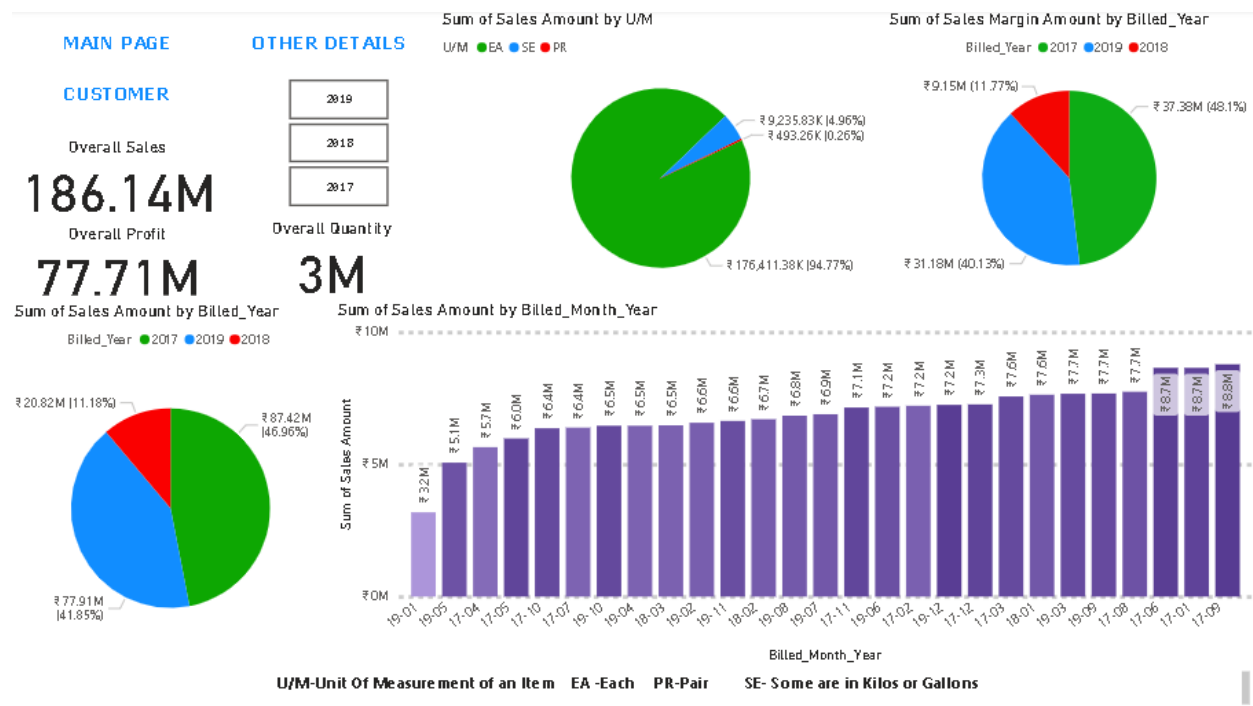
Sales Margin Amount		
Item	Billed_Month_Year	
Fast Mini Donuts	19-07	410520.45
High Top Dried Mushrooms	18-01	369730.26
Fast Mini Donuts	17-12	328386.96
	19-03	324178.73
Better Large Canned Shrimp	19-09	323438.89
	18-01	322267.57
High Top Dried Mushrooms	19-04	312223.88
Fast Mini Donuts	17-01	310527.12
	17-08	309201.71
	17-02	309201.71
Better Large Canned Shrimp	18-02	308579.99
	17-06	301970.74

In above figure showing max profit item in month year basis that is Fast Mini Donuts in Year 2019, High Top Dried Mushrooms in 2018, Fast Mini Donuts year 2017 and Better Large Canned Shrimp in year 2019.

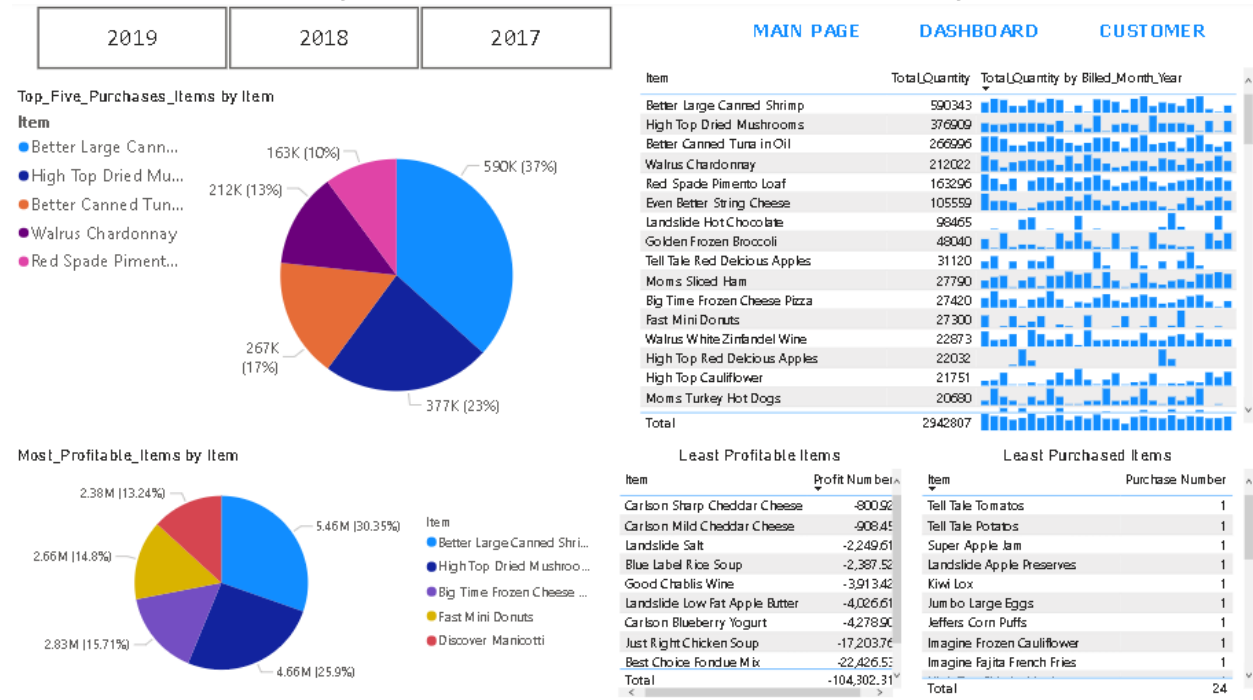
Fast Lemon Cookies	19-03	-2321.54
	19-09	-2321.54
	17-06	-2450.44
	17-05	-2450.44
	19-06	-2476.20
Best Choice Fondue Mix	18-02	-2488.89
	17-03	-2508.21
	17-09	-2508.21
	19-03	-2533.97
	19-09	-2533.97
Just Right Chicken Soup	17-12	-2879.53
	19-12	-3036.22
	19-10	-3036.22
	17-09	-3378.71
	19-11	-3932.93
Fast Lemon Cookies	17-12	-4335.59
	19-10	-4724.99
	19-12	-4724.99
	17-10	-7944.40

In above figure showing least profit item in month year basis that is Fast Lemon Cookies in Year 2019, Best Choice Fondue Mix in 2018 and Just Right Chicken Soup in year 2019.

## POWER-BI DASHBOARD



In the above figure in dashboard contain criteria for selection on the basis of year. The three number cards contains total sales, total quantity and total profit. Whereas top pie chart sale items on basis of unit of measurement and second one is profit percentage. The bottom pie chart contains the total sales percentage. Then the stack bar chart display sales on the basis of month year.



In above figure criteria selection on the basis of year. The top left side pie chart display top most selling items over 3 years. The second bottom side pie chart display highest profit items over the 3 years. The side table shows that quantity of items with detail years. The bottom two tables shows least profitable items and least quantity purchased items.