Budget Analytics and Dashboard Creation with SQL & Tableau

By: Mark Giannillo

Introduction:

At the time of creating this project, I work as a Financial Analyst supporting the Global Marketing department of Hasbro, Inc. In my day-to-day, I use my data analytics skills to help brand teams across the organization with budget tracking and management. This process involves using tools like Microsoft Excel and Allocadia to build custom budget trackers that communicate key insights to stakeholders.

As a passionate data analyst even outside of the workplace, I've studied many analytical tools that I often wish I had access to in my role at Hasbro. The purpose of this project is to demonstrate how a company like Hasbro could implement SQL and Tableau to make full use of their data and make better informed business decisions. I intend to showcase a combination of my expertise in working with corporate financial data and my self-taught technical skills by conducting an exploratory analysis and building an interactive dashboard from a simulated purchase order dataset called '*PO_Data.csv*'.

Guide:

Part 1: Simulating the Data

Part 2: Exploratory Analytics in Microsoft SQL Server Management Studio

Part 3: Creating Business Intelligence Dashboard in Tableau

Part 1 - Simulating the Data

After many attempts of searching for a public dataset that was suited to my ambitions for this project, I decided that I would use the Python programming language to simulate my own dataset. This data is meant to be an artificial copy of the datasets I interact with in my role at Hasbro. The dataset is called 'PO_Data.csv', and each row represents a purchase order. Each of these orders are associated with certain attributes such as brand, spending category and price. This dataset will be used for analysis in parts 2 and 3 of this project.

To create this dataset, I used the Python packages *pandas*, *random*, *time*, and *datetime*. The first thing I did was create an empty pandas DataFrame with the column headers that I wanted to include. I set the dimension so that the DataFrame would have 5000 rows.

```
In [8]: purchase_orders.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 5000 entries, 3427 to 3977
Data columns (total 8 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Vendor          5000 non-null   object
 1   Type_of_Spend   5000 non-null   object
 2   Brand           5000 non-null   object
 3   PO_Number       5000 non-null   int64
 4   PO_Amount       5000 non-null   int64
 5   Invoice_Amount  5000 non-null   int64
 6   Difference      5000 non-null   int64
 7   Date_Opened     5000 non-null   datetime64[ns]
dtypes: datetime64[ns](1), int64(4), object(3)
memory usage: 351.6+ KB
```

Column names and data types

My next step was to populate the columns. Each observation has a randomly selected vendor, brand, purchase order number, price, and invoice amount generated by functions from the random package. The categorical variables(*'Brand'*,*'Vendor'*,*'Type_of_Spend'*) were selected at random from predetermined lists created by me. The price amount column is a random number between 200 and 100,000 and the invoice amount column is a random number between 0 and the value in the price amount column for that row, ensuring the invoice amount never exceeds the price amount. I found a function online that selected a random date between two dates, which I used to populate the Date Opened column for the year 2022.

This resulted in data that had little fluctuation in purchase order spending across the year. In my experience as a Financial analyst in the Toy industry, spending tends to increase throughout the year and peak in Q4. To adjust the data to reflect this type of spending activity, I increased all rows in Q3

and Q4 by adding $25,000 and $50,000 respectively to each item these quarters. This simulated a higher volume of spending in these periods, which is closer to how the data would look in real life.

```
purchase_orders.loc[Q3, 'PO_Amount'] = purchase_orders['PO_Amount'] + 25000

purchase_orders.loc[Q4, 'PO_Amount'] = purchase_orders['PO_Amount'] + 50000
```

Adding to Q3 & Q4

Lastly, I subtracted the price amount from the invoice amount and populated the 'Difference' column with the difference. I tested the shape of the data one last time by calculating the sums of purchase order spending for each quarter, and then decided it was ready for analysis. I used a function to convert the DataFrame into a csv so that I could access the data in the next step.

```
purchase_orders.to_csv('PO_Data.csv')
```

Exporting as CSV

Part 2 - Exploratory Analysis with SQL

After exporting my simulated DataFrame as a csv file, I created a Database in Microsoft SQL Server Management Studio and imported the '*PO_Data.csv*'. From here, I was able to access the data by writing queries to retrieve certain information.

Column Name	Data Type
column1	smallint
Vendor	nvarchar(50)
Type_of_Spend	nvarchar(50)
Brand	nvarchar(50)
PO_Number	int
PO_Amount	int
Invoice_Amount	int
Difference	int
Date_Opened	date

Figure 1

Above includes a SQL Server generated breakdown of the column names and data types included in this project. For the query in Figure 1 I had to cast *'Date_Opened'* as a varchar data type in order to use the substring clause.

To get a look at spending from an annual perspective, I wrote a complex query that involves a sub-query and case-when sequence. Insights like this could be very useful for tracking spending against quarterly budgets, and help to prevent overspending and underspending.

```
SELECT
 Period,
 Month,
 FORMAT(SUM(PO_Amount), 'C') AS Total_PO_Spending
 FROM (SELECT
 *,
 CASE
     WHEN Date_Opened BETWEEN '01/01/2022' AND '03/31/2022' THEN 'Q1'
     WHEN Date_Opened BETWEEN '04/01/2022' AND '06/30/2022' THEN 'Q2'
     WHEN Date_Opened BETWEEN '07/01/2022' AND '09/30/2022' THEN 'Q3'
     WHEN Date_Opened BETWEEN '10/01/2022' AND '12/31/2022' THEN 'Q4'
     ELSE 'NA'
 END AS Period,
 SUBSTRING(CAST(Date_Opened AS VARCHAR), 6, 2) AS Month
 FROM po_data..PO_Data) AS case_table
 GROUP BY Month, Period
 ORDER BY Month, Total_PO_Spending
```

Vendor	Total_Spending	PO_Count
Google Inc.	$45,664,738.00	658
TikTok Inc	$43,530,532.00	635
Meta Platforms	$43,359,088.00	641
KYG Antler LLC	$43,047,853.00	617
Disney Studios	$42,051,186.00	612
Nintendo Technologies	$40,897,424.00	617
The Martingale Agency	$40,894,321.00	596
Pinterest Inc	$40,705,535.00	624

Figure 2

Every row in the simulated dataset is associated with certain categorical variables, such as brand and spending category, so I created several queries to explore the distribution of total spending between these variables. Queries of this type could help a company keep track of how much funding is being allocated to certain brands or spending categories within their organization.Note that the price increases gradually from the beginning of the year to the end of the year as intended in the simulation.

```
---- A look at Purchase Order volume by Spend Type ----
SELECT Type_of_Spend,
FORMAT(SUM(PO_Amount), 'C') AS Total_Spending,
COUNT(PO_Number) AS PO_Count
FROM po_data..PO_Data
GROUP BY Type_of_Spend
ORDER BY Total_Spending DESC
```

Period	Month	Total_PO_Spending
Q1	01	$22,006,885.00
Q1	02	$19,341,965.00
Q1	03	$22,557,727.00
Q2	04	$21,413,265.00
Q2	05	$19,817,357.00
Q2	06	$20,776,057.00
Q3	07	$32,522,619.00
Q3	08	$30,194,423.00
Q3	09	$32,083,608.00
Q4	10	$42,273,401.00
Q4	11	$38,991,580.00
Q4	12	$38,171,790.00

Figure 3

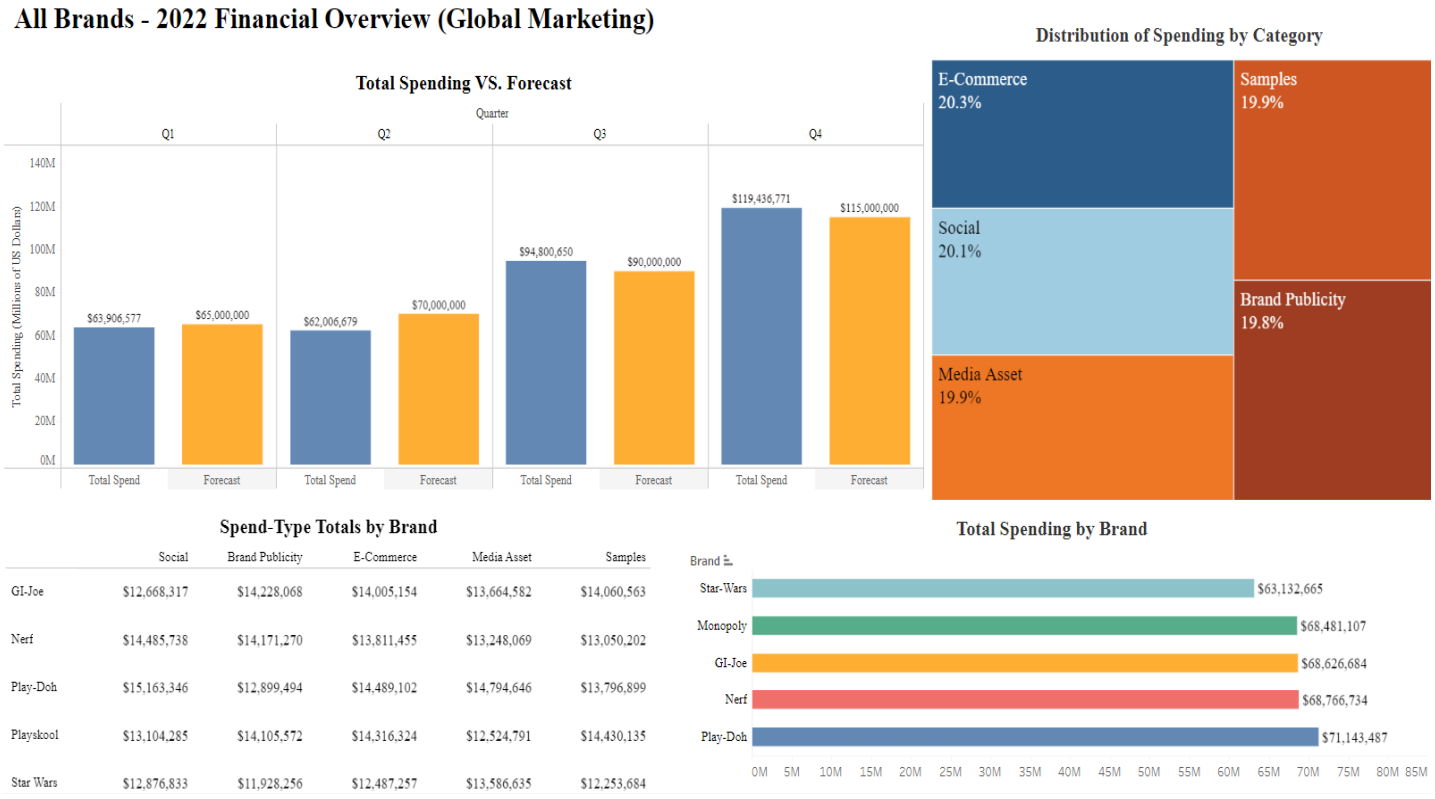
The query above and the table that it generated was one of several that I used to create the data visualization in my tableau dashboards in the next step.

Part 3 - Dashboard Visualizations with Tableau

The last step in this project was using the simulated dataset and the tables generated by some of my SQL queries to create a Tableau workbook for budget planning and tracking. Tableau is a business intelligence tool that helps users create interactive dashboards and compelling data visualizations. I set up the workbook so that it would have two dashboards: one to analyze the marketing budget as a whole and one that gives brand-specific budget information and can be filtered by the user according to their needs.

Dashboard 1

The first dashboard in this workbook contains a holistic view of budgets across all brands and types of spend. This includes a time-series bar chart that tracks total spending per fiscal quarter against a predetermined budget. It also includes visualizations that compare spending across brands and show how those funds are allocated among spending categories. Information like this could help the directors of the Global Marketing department understand how costs are broken down across brands and spending categories, making for better informed budgeting decisions.



Dashboard 2

The next dashboard in the workbook is meant to help users access more granular insights than Dashboard 1 provides. It has a slicer button where you can filter the visualizations based on their brand or view data visualizations for multiple brands at once. In the upper-left corner of the dashboard I created a stacked bar chart that shows spending levels by month. The colors of each bar represent a type of spend and their size is proportional to the amount of spending in each category. These colors and their corresponding spend types are detailed in the key below the chart. Each of these data visualizations provides a powerful look into spending activity and makes for a great tool for marketing teams to view insights at a brand-specific level.