Restaurant Sales Analysis Project

This project analyzes restaurant sales data over a period of 12 months with the goal of providing insights into sales trends, product category performance, and month-over-month sales growth. By examining the detailed sales data, our goal is to identify key patterns that can guide future business decisions, such as inventory management, marketing strategies, and product offerings.

Analysis Objectives:

1. **Evaluate total sales trends**: Understand fluctuations in sales over the year, identifying months with significant increases or decreases.
2. **Evaluate sales breakdown by product category**: Analyze the contribution of fast food and beverage categories to total sales, highlighting their relative importance.
3. **Identify top-selling products**: Highlight the most popular products at the restaurant and understand which contribute most to total sales.
4. **Examine month-over-month sales growth**: Observe and analyze changes in sales from one month to the next, identifying growth and decline patterns.

Conclusion:

Through this analysis, the restaurant can better understand sales dynamics, optimize product mix, and make data-driven decisions to enhance profitability and operational efficiency.

Step 1: Import the CSV File

This step imports the dataset from the provided file.

```
import delimited "/Users/jovanyvillegas/Desktop/Restaurant-Sales-
Analysis/data/Balaji Fast Food Sales.csv", clear
```

Step 2: Data Cleaning

This step checks for and handles missing and duplicate values.

2.1 Check for missing values in all variables

```
misstable summarize
```

2.2 Check for missing values in a specific column (e.g., transaction_amount)

```
tabulate transaction_amount, missing
```

2.3 Check for duplicates based on order_id

```
duplicates report order_id
```

Step 3: Create and Clean Date Variables

This step converts and creates date-related variables for future analysis.

3.1 Create the `date_stata` variable

```
gen date_stata = date(date, "MDY")  // Adjust format if necessary (e.g.,
"DMY")
```

3.2 Create the `month` variable from `date_stata`

```
gen month = month(date_stata)
```

Step 4: Data Aggregation

This step creates new aggregated variables such as total sales.

4.1 Create the `total_sales` variable using `transaction_amount`

```
gen total_sales = transaction_amount
```

4.2 Summarize data by month and calculate total sales

```
collapse (sum) total_sales, by(month)
```

4.3 Verify the aggregated data

```
list month total_sales
```

Step 5: Visualization of Total Sales by Month

This step generates a bar graph to visualize total sales by month.

5.1 View total sales by month

```
list month total_sales
```

5.2 Create a bar graph of total sales by month

```
graph bar total_sales, over(month) title("Total Sales by Month")
```

5.3 Save the graph as a PNG image

```
graph export "/Users/jovanyvillegas/Desktop/Restaurant-Sales-
Analysis/visualizations/total_sales_by_month.png", replace
```

Step 6: Breakdown of Sales by Product Type (for Pie Chart Visualization)

This step aggregates sales by product type and creates a pie chart for visualization.

6.1 Reimport the data

```
import delimited "/Users/jovanyvillegas/Desktop/Restaurant-Sales-
Analysis/data/Balaji Fast Food Sales.csv", clear
```

6.2 Verify the variables to confirm the correct product type variable name

```
describe
```

6.3 Aggregate total sales by product type

```
collapse (sum) total_sales = transaction_amount, by(item_type)
```

6.4 Calculate total sales across all categories

```
summarize total_sales
scalar total_sales_all = r(sum)
display "Total Sales Across All Categories: " total_sales_all
```

6.5 Calculate the percentage contribution of each product type

```
gen percentage = (total_sales / total_sales_all) * 100
```

6.6 View the results

```
list item_type total_sales percentage
```

6.7 Create a pie chart of sales by product type

```
graph pie total_sales, over(item_type) title("Sales Breakdown by Product
Type")
```

6.8 Save the pie chart as a PNG image

```
graph export "/Users/jovanyvillegas/Desktop/Restaurant-Sales-
Analysis/visualizations/sales_by_item_type.png", replace
```

Step 7: Generate Descriptive Statistics

This step generates descriptive statistics for key variables in the dataset.

7.1 Reimport the data (as before)

```
import delimited "/Users/jovanyvillegas/Desktop/Restaurant-Sales-
Analysis/data/Balaji Fast Food Sales.csv", clear
```

7.2 Recreate the `total_sales` variable

```
gen total_sales = transaction_amount
```

7.3 Descriptive statistics for total sales

```
summarize total_sales
```

7.4 Descriptive statistics for `transaction_amount` (original sales data)

```
summarize transaction_amount
```

Step 8: Visualization of Total Sales Trend (Line Graph)

This step visualizes the trend of total sales over time using a line graph.

8.1 Recreate `month` and `total_sales` variables if necessary

```
gen date_stata = date(date, "MDY")  // Adjust format if necessary
gen month = month(date_stata)
```

8.2 Collapse data by month and calculate total sales

```
collapse (sum) total_sales, by(month)
```

8.3 Create a line graph to visualize the trend of total sales by month

```
graph twoway (line total_sales month), title("Sales Trend by Month")
```

8.4 Save the line graph as a PNG image

```
graph export "/Users/jovanyvillegas/Desktop/Restaurant-Sales-
Analysis/visualizations/sales_trend_by_month.png", replace
```

Step 9: Visualization of Total Sales by Product Type (Bar Graph)

This step generates a bar graph to visualize total sales by product type.

9.1 Reimport the data

```
import delimited "/Users/jovanyvillegas/Desktop/Restaurant-Sales-
Analysis/data/Balaji Fast Food Sales.csv", clear
```

9.2 Recreate the `total_sales` variable

```
gen total_sales = transaction_amount
```

9.3 Summarize total sales by product type

```
bysort item_type: summarize total_sales
```

9.4 Create a bar graph to visualize total sales by product type

```
graph bar total_sales, over(item_type) title("Total Sales by Product Type")
```

9.5 Save the bar graph as a PNG image

```
graph export "/Users/jovanyvillegas/Desktop/Restaurant-Sales-
Analysis/visualizations/sales_by_item_type_bar.png", replace
```

Step 10: Analyze Total Sales by Day of the Week

This step aggregates sales by day of the week and visualizes the results.

10.1 Create the `date_stata` variable if needed

```
gen date_stata = date(date, "MDY")
```

10.2 Create the `day_of_week` variable

```
gen day_of_week = dow(date_stata)
```

10.3 Assign day names (Sunday to Saturday)

```
gen day_name = ""
replace day_name = "Sunday" if day_of_week == 0
replace day_name = "Monday" if day_of_week == 1
replace day_name = "Tuesday" if day_of_week == 2
replace day_name = "Wednesday" if day_of_week == 3
replace day_name = "Thursday" if day_of_week == 4
replace day_name = "Friday" if day_of_week == 5
replace day_name = "Saturday" if day_of_week == 6
```

10.4 Collapse the data to calculate total sales by day of the week

```
collapse (sum) total_sales = transaction_amount, by(day_name)
```

10.5 Create a bar graph to visualize total sales by day of the week

```
graph bar total_sales, over(day_name, sort(1)) title("Total Sales by Day of
the Week")
```

10.6 Save the bar graph as a PNG image

```
graph export "/Users/jovanyvillegas/Desktop/Restaurant-Sales-
Analysis/visualizations/sales_by_day_of_week.png", replace
```

Step 11: Analyze Month-over-Month Sales Growth

This step calculates and visualizes month-over-month sales growth.

11.1 Import the dataset

```
import delimited "/Users/jovanyvillegas/Desktop/Restaurant-Sales-
Analysis/data/Balaji Fast Food Sales.csv", clear
```

11.2 Describe the dataset

```
describe
```

11.3 Create the `month` variable

```
gen month = month(date_stata)
```

11.4 Collapse data by month to calculate total sales for each month

```
collapse (sum) total_sales = transaction_amount, by(month)
```

11.5 Declare the data as a time series

```
tsset month
```

11.6 Create the sales growth variable

```
gen sales_growth = (total_sales - L.total_sales) / L.total_sales * 100
```

11.7 Visualize month-over-month sales growth

```
graph twoway (line sales_growth month), title("Month-over-Month Sales
Growth")
```

Step 12: Identify Top-Selling Products

This step identifies the top-selling products and visualizes them.

12.1 Reimport the dataset

```
import delimited "/Users/jovanyvillegas/Desktop/Restaurant-Sales-
Analysis/data/Balaji Fast Food Sales.csv", clear
```

12.2 Aggregate sales by product name

```
collapse (sum) total_sales = transaction_amount, by(item_name)
```

12.3 Visualize the sales breakdown by product

```
graph bar total_sales, over(item_name) title("Sales Breakdown by Product")
```