TASK 2: Data Visualization & Storytelling

This document provides the process used to create a dynamic sales analysis dashboard for the online grocery app, blinkit, using Microsoft Power BI. The primary objective of this project is to transform raw sales data into a powerful, interactive tool that delivers actionable business insights into sales performance, customer satisfaction, and operational efficiency.

1.0 Project Overview and Business Requirements

The entire development process is driven by a clear set of business requirements that serve as the blueprint for the dashboard. This section outlines the dashboard's core purpose, its key analytical features, and the specific metrics and visualizations requested to meet the project's goals.

1.1 Dashboard Purpose and Key Features

The primary objective of the blinkit dashboard is to conduct a comprehensive analysis of sales performance, customer satisfaction, and inventory distribution. It is designed to provide stakeholders with a clear and interactive view of the business, enabling data-driven decision-making.

The key features of the completed dashboard include:

- **Main KPIs:** A top-level summary of four critical performance indicators: Total Sales, Average Sales, Number of Items Sold, and Average Customer Rating.
- **Product-Focused Analysis:** A dedicated section that allows users to explore performance metrics broken down by product attributes such as fat content and item type.
- **Outlet-Focused Analysis:** A second analytical section focused on outlet performance, visualizing sales trends by establishment year, size, location, and type.
- **Interactive Filters:** A suite of user-friendly slicers that enable dynamic filtering of the entire dashboard by outlet location, size, and item type, allowing for deep-dive analysis.
- **Dynamic Metric Selection:** A Field Parameter that empowers users to switch the metric displayed across multiple charts, seamlessly changing the analysis from Total Sales to Average Sales, Number of Items, or Average Rating.

1.2 Stated Business Requirements

The dashboard's design and functionality are directly mapped to a predefined set of business requirements. These requirements ensure that the final product addresses the core analytical needs of the business.

Requirement Type	Specification
KPI Requirements	<ul><li>**Total Sales:** The overall revenue generated from all items sold.</li><li>**Average Sales:** The average revenue generated per sale transaction.</li><li>**Number of Items:** The total count of all items sold.</li><li>**Average Rating:** The average rating provided by customers for the items sold.</li></ul>
Chart Requirements	<ol><li>Total sales by fat content using a **Donut Chart**.</li><li>Total sales by item type using a **Bar Chart**.</li><li>Fat content by outlet for total sales using a **Clustered Bar Chart**.</li><li>Total sales by Outlet establishment year using a **Line Chart**.</li><li>Total sales by Outlet size using a **Donut Chart**.</li><li>Total sales by Outlet location using a **Funnel Chart**.</li><li>All key metrics (sales, items, rating) by Outlet type using a **Matrix Visual**.</li></ol>

With these requirements established, the first step in the development process is to connect to the source data and prepare it for analysis.

2.0 Data Preparation and Transformation

The data preparation stage is a critical foundation for any reliable analysis. This phase involves connecting to the raw data source, performing necessary cleaning and transformation to ensure accuracy and consistency, and verifying the overall quality of the dataset before it is loaded into the Power BI data model for visualization.

2.1 Data Connection

The analysis is based on a single dataset contained within an Excel workbook.

- **Data Source:** An Excel file named blinkit grocery data.
- **Connection Process:** The data was imported into Power BI using the **Get Data** > **Excel workbook** functionality from the Home ribbon.
- **Dataset Size:** The imported dataset contains approximately 8,523 rows and 12 columns of transaction-level sales data.

2.2 Data Cleaning in Power Query

Upon initial inspection, inconsistencies were identified in the raw data, necessitating a cleaning process within the Power Query Editor. The Item Fat Content column, in particular, contained non-standardized entries that required correction to ensure accurate grouping and analysis.

The following transformations were performed to standardize the values in this column:

Original Value	Cleaned Value
LF	Low Fat
reg	Regular
low fat	Low Fat

These changes were executed using the following steps in the Power Query Editor:

- The Power Query Editor was opened by selecting the **Transform data** button in the Power BI Desktop Home ribbon.
- The Item Fat Content column was selected.
- The **Replace Values** feature was used sequentially to replace each non-standard value with its correct, standardized equivalent (e.g., replacing "LF" with "Low Fat").
- Each of these transformation steps was automatically recorded in the **Applied Steps** pane in Power Query, ensuring the cleaning process is transparent and repeatable.

2.3 Data Quality Verification

After performing the cleaning operations, the overall quality of the dataset was verified using the built-in data profiling tools in Power Query.

- The **Column Quality** feature, accessible from the **View** tab in the Power Query Editor, was enabled to check for errors or empty values.
- This verification confirmed that, after cleaning, the primary dimension fields (such as Item Fat Content and Item Type) were **100% valid**, with **0% errors** and **0% empty values**.
- It was noted that the Item Weight column contained **16% empty values**. However, this was deemed acceptable for the project, as this field was not a critical component of the stated business requirements for the dashboard.

Once the data was confirmed to be clean and reliable, the **Close & Apply** function was used to load the transformed dataset into the Power BI model, making it ready for the design phase.

3.0 Dashboard Canvas and Layout Design

Before creating visualizations, the Power BI canvas was configured and a foundational layout was designed. A well-structured and visually appealing layout is essential for enhancing readability, guiding the user's attention, and creating a professional user experience.

3.1 Configuring the Report Canvas

The default report page was customized to create a wider, more accommodating canvas for the dashboard components.

- **Canvas Settings:**
    - Type: Custom
    - Height: 800 pixels
    - Width: 1400 pixels
- **Canvas Background:**
    - Color: White (#FFFFFF)
    - Transparency: 40%

3.2 Designing the Layout Components

The main layout structure was built using shape visuals to create distinct panels and content areas.

- A vertical panel on the left-hand side was created to house the dashboard title and slicers. This was achieved by inserting a shape visual (specifically, a 'Rounded tab both top' shape rotated 90 degrees to function as a vertical rounded rectangle) with the following key properties:
    - **Rotation:** 90 degrees
    - **Size:** Height 750, Width 205
    - **Style:** Fill Color #FFD200 (yellow), Border Off, Shadow On

- The dashboard title **"blinkit"** and the tagline **"India's last minute app"** were added using text boxes. The title was formatted with a font size of 36, and the tagline with a font size of 10, both in a green font color to align with the brand.
- Two large, rounded rectangles were added to the main content area to serve as placeholders for the product and outlet analysis sections. These were styled with a white fill color and a subtle 10% transparency to create a layered effect.

This structured layout provides a clean and organized canvas for the subsequent development of DAX measures and data visualizations.

4.0 Core Metric Development (DAX)

Data Analysis Expressions (DAX) are used to create the analytical engine of the dashboard. DAX formulas define calculated measures that go beyond simple aggregations, enabling dynamic calculations that respond to user interactions with filters and slicers. These measures form the core of the KPIs and chart-based analyses.

4.1 Creating Key Performance Indicator (KPI) Measures

The four primary KPIs identified in the business requirements were created as new measures using straightforward DAX formulas.

- **Total Sales:** Calculates the sum of all sales revenue from the Item_Outlet_Sales column.
- **Average Sales:** Calculates the average revenue per transaction.
- **Number of Items:** Counts the total number of transactions by counting the rows in the table.
- **Average Rating:** Calculates the average customer rating for all items sold.

4.2 Implementing Dynamic Metric Selection with Field Parameters

To meet the requirement of analyzing multiple metrics (Total Sales, Average Sales, etc.) within the same set of charts, a Field Parameter was implemented. This feature allows end-users to dynamically switch the measure being visualized.

- **Creation:** The parameter was created by navigating to the **Modeling** tab and selecting **New parameter** > **Fields**.
- **Configuration:** The parameter was named Matrix and configured to include the four core DAX measures: Total Sales, Average Sales, Number of Items, and Average Rating.

- **Functionality:** This action automatically generated a new table and a slicer visual. When added to the report canvas, this slicer allows users to select a single metric, which then dynamically populates the values in all charts connected to the Matrix parameter.

With the core DAX measures and the dynamic parameter in place, the project moved to the visualization stage.

5.0 Building Dashboard Visualizations

This section details the construction and formatting of each visual element on the dashboard. Every chart and KPI card was carefully designed to answer a specific business question outlined in the requirements phase, transforming the calculated DAX measures into clear and insightful visualizations.

5.1 Main KPI Visual

The four main KPIs were displayed prominently at the top of the dashboard using the modern **Card (new)** visual.

- **Configuration:** The visual was configured with a grid layout of **2 rows and 2 columns** to neatly organize the four metrics.
- **Formatting:** A **rounded rectangle shape** was applied to the cards along with a green **accent bar** to create visual separation and emphasis.
- **Data Display:** Callout values were formatted to display currency symbols for sales figures and appropriate decimal places for the average rating.
- **Enhancements:** To improve visual appeal and provide immediate context, custom **background images and icons** were added to each of the four KPI cards. The background images were set to **40% transparency** to create a subtle, layered effect.

5.2 Product Analysis Section

The top-right placeholder was dedicated to product-centric analysis, featuring three charts powered by the interactive Matrix Field Parameter slicer.

- **Fat Content (Donut Chart):** This chart visualizes the user-selected metric (e.g., Total Sales) broken down by the two main product categories: 'Low Fat' and 'Regular'.

- **Item Type (Stacked Bar Chart):** This visual displays the performance of different product categories, such as 'Fruits and Vegetables' and 'Snack Foods,' based on the metric selected in the Matrix slicer.
- **Fat by Outlet (Clustered Bar Chart):** This chart provides a more granular view, breaking down the selected metric by outlet location tier (Tier 1, Tier 2, Tier 3) and further segmenting the data by fat content.

5.3 Outlet Analysis Section

The bottom-right placeholder houses visuals focused on outlet-specific performance.

- **Outlet Establishment (Line Chart):** This chart plots Total Sales against the year each outlet was established, revealing performance trends over time. The visual was enhanced with a smooth line, markers, and a shaded area to improve readability.
- **Outlet Size (Donut Chart):** This chart provides a simple but effective comparison of Total Sales across 'Small', 'Medium', and 'High' outlet sizes.
- **Outlet Location (Funnel Chart):** This visual compares sales performance across the three outlet location tiers, effectively showing the contribution of each tier to the total.
- **Metrics by Outlet Type (Matrix Visual):** This powerful matrix displays all key metrics (Total Sales, Average Sales, etc.) broken down by each 'Outlet Type' (e.g., 'Supermarket Type1', 'Grocery Store'). Conditional background colors were applied to specific columns to create a heatmap effect, making it easier to spot patterns and compare performance across different outlet formats.

The final step was to add user controls to make the dashboard fully interactive.

6.0 Implementing Interactivity

The final stage of development focused on making the dashboard a fully dynamic and user-friendly analytical tool. This was achieved by adding slicers for filtering and customizing the default interaction behavior between visuals.

6.1 Configuring Slicers

Three slicers were added to the left-hand panel to give users control over the data displayed in the report.

- **Outlet Location:** Filters the entire report by outlet tier (Tier 1, 2, or 3).
- **Outlet Size:** Filters the data by outlet size (Small, Medium, or High).
- **Item Type:** Allows users to select one or more specific product categories for a focused analysis.

To optimize canvas space and maintain a clean design, each slicer was formatted as a **Dropdown** menu and styled with the same yellow background as the panel it resides in.

6.2 Customizing Visual Interactions

A critical step to enhance the user experience was to change the default interaction behavior between the dashboard's visuals.

- **Process:** The interaction settings were modified by selecting a visual, navigating to the **Format** tab, and clicking **Edit interactions**.
- **Configuration Change:** The default interaction behavior in Power BI is 'Highlight', which dims non-selected data points in other visuals. This was changed to **'Filter'** for all relevant interactions.
- **User Benefit:** This change provides a much more intuitive analytical experience. Now, when a user clicks a data point—for example, the 'Medium' slice on the Outlet Size donut chart—all other visuals on the page are actively filtered to show data only for medium-sized outlets. This removes visual clutter and makes the relationships between different data points much clearer.

With these interactive elements in place, the blinkit sales dashboard was complete, delivering a fully functional and insightful business intelligence tool.