Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 6**	
Title of Lab Assignment: Implement data Visualization With Ggplot2.		
DOP: 22-10-2023	**DOS: 27-10-2023**	
CO Mapped: **CO4**	**PO Mapped:** **PO1, PO2, PO3, PO4, PO5,** **PO7, PO8, PO9, PO12,** **PSO1, PSO2**	**Signature:**

Practical No. 6

Aim: Implement data Visualization With Ggplot2.

Description:

Data visualization is a critical aspect of data analysis, allowing you to present complex data in a clear and understandable way. ggplot2 is a popular data visualization package in R that offers a versatile and flexible approach to creating a wide range of plots. This note provides an overview of the process for implementing data visualization with ggplot2.

1. **Load and Install ggplot2:**

   If you haven't already, install and load the ggplot2 package using the following commands:

   install.packages("ggplot2")

   library(ggplot2)

2. **Data Preparation:**

   Start with a well-structured and clean dataset. Ensure your data is organized, and any necessary data wrangling or transformations have been applied. In ggplot2, you typically work with data frames.

3. **Basic ggplot Structure:**

   The fundamental structure of a ggplot2 visualization includes the following components:

   - Data: Specify the data frame that contains your dataset.
   - Aesthetics (aes): Map data variables to visual properties such as x and y coordinates, color, size, and shape.
   - Geometric Objects (Geoms): Choose a geom to determine how the data will be visually represented (e.g., points, lines, bars).
   - Layers: Add layers to the plot, including additional geoms, statistical transformations, facets, themes, and annotations.

4. **Creating a Simple Scatter Plot:**

   To create a basic scatter plot, use the following template:

   ggplot(data = your_data, aes(x = variable1, y = variable2)) + geom_point()

- `data`: The data frame.
- `aes`: Aesthetics, mapping variables to visual properties.
- `geom_point()`: Geom for a scatter plot.

5. **Customization:**
   Customize your plot to make it more informative and visually appealing. Use various functions to:
   - Set titles and axis labels with `labs()`.
   - Customize axis scales with `scale_x_()` and `scale_y_()`.
   - Adjust plot themes with `theme()`.
   - Annotate data points with labels using `geom_text()` and `geom_label()`.
   - Add statistical summaries with functions like `geom_smooth()`.

6. **Faceting:**
   Create small multiples or facet your plot using the `facet_wrap()` or `facet_grid()` functions. This is useful when you want to compare subsets of your data in separate panels.

7. **Export Your Plot:**
   Save your plot to a file using the `ggsave()` function.
   For example:
   ggsave("my_plot.png", plot = your_plot, width = 6, height = 4, units = "in")

8. **Advanced Visualizations:**
   ggplot2 supports a wide range of visualizations, including bar charts, line graphs, box plots, and more. Each type of plot requires specific geoms and customization options. You can also use dplyr for advanced data transformations.

9. **Learning Resources:**
   To master ggplot2, consider reading "ggplot2: Elegant Graphics for Data Analysis" by Hadley Wickham. Online tutorials, documentation, and the R community provide valuable resources for learning and troubleshooting.

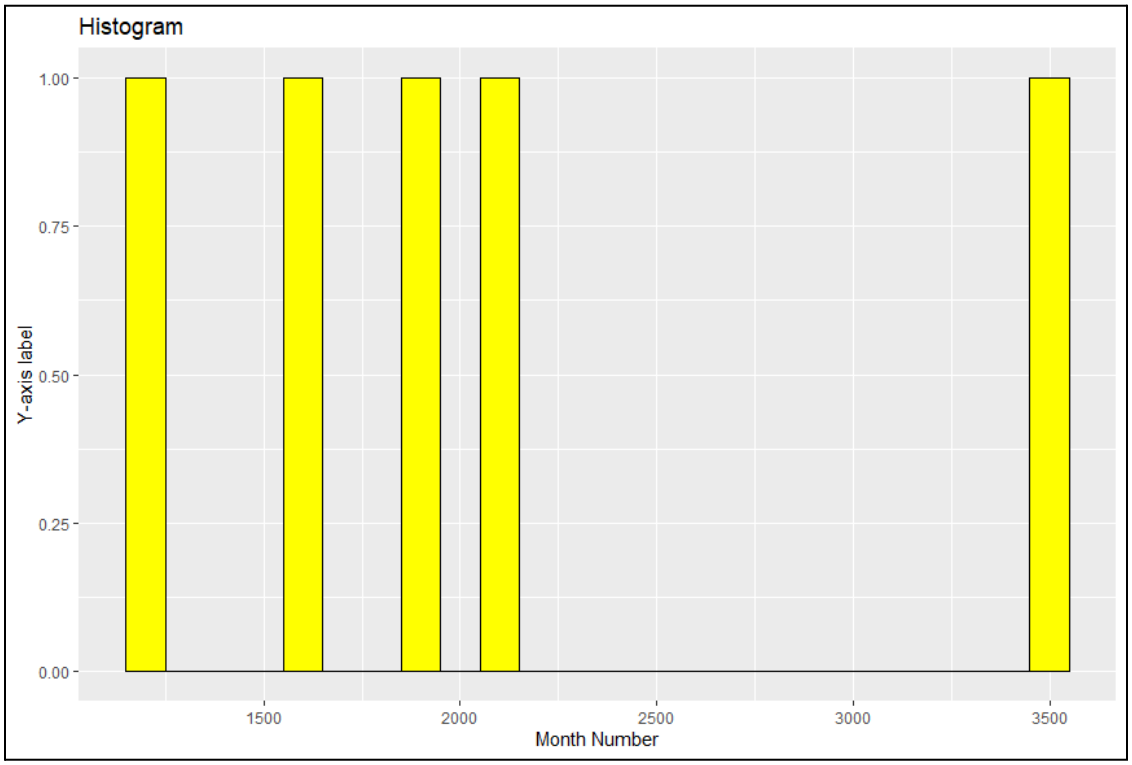
Code:

```
install.packages("ggplot2")  # Install and load necessary packages
library(ggplot2)
data <- read.csv("F:/Pushkar/MCA/Sem-1/DAR/SalesData1.csv")
data
# Create a histogram
ggplot(data, aes(x = shampoo)) +
  geom_histogram(binwidth = 100, fill = "yellow", color = "black") +
  labs(title = "Histogram", x = "Month Number", y = "Y-axis label")

# Create a boxplot
ggplot(data, aes(x = 'Group' , y = facewash)) +
  geom_boxplot(fill = "green", color = "black") +
  labs(title = "Boxplot", x = "Group", y = "Facewash")

# Create a bar chart
ggplot(data, aes(x = shampoo)) +
  geom_bar(fill = "purple", color = "black") +
  labs(title = "Bar Chart", x = "X-Shampoo", y = "Y-axis label")

# Create a line graph
ggplot(data, aes(x = shampoo, y = facewash)) +
  geom_line(color = "red") +
  labs(title = "Line Graph", x = "Shampoo", y = "Facewash")

# Create a scatterplot
ggplot(data, aes(x = shampoo, y = facewash)) +
  geom_point(color = "orange") +
  labs(title = "Scatterplot", x = "Shampoo", y = "Facewash")

# Create a pie chart
pie(data$total_units, labels = data$labels, main = "Pie Chart")
```

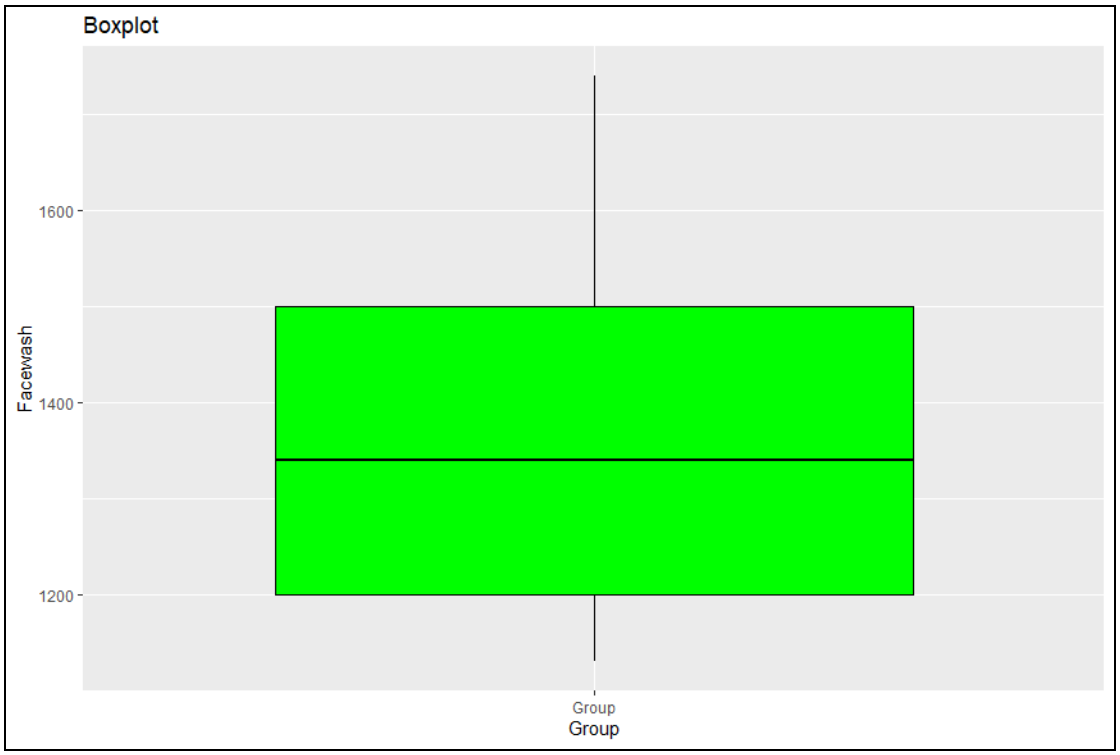
Output;

```
> library(ggplot2)
>
> data <- read.csv("F:/Pushkar/MCA/Sem-1/DAR/SalesData1.csv")
> data
  month_number facecream facewash toothpaste bathingsoap shampoo moisturizer
total_units total_profit
1            1      2500     1500       5200        9200    1200        1500
21100       211000
2            2      2630     1200       5100        6100    2100        1200
18330       183300
3            3      2140     1340       4550        9550    3550        1340
22470       224700
4            4      3400     1130       5870        8870    1870        1130
22270       222700
5            5      3600     1740       4560        7760    1560        1740
20960       209600
> # Create a histogram
> ggplot(data, aes(x = shampoo)) +
+  geom_histogram(binwidth = 100, fill = "yellow", color = "black") +
+  labs(title = "Histogram", x = "Month Number", y = "Y-axis label")
> # Create a boxplot
> ggplot(data, aes(x = 'Group' , y = facewash)) +
+  geom_boxplot(fill = "green", color = "black") +
+  labs(title = "Boxplot", x = "Group", y = "Facewash")
> # Create a bar chart
> ggplot(data, aes(x = shampoo)) +
+  geom_bar(fill = "purple", color = "black") +
+  labs(title = "Bar Chart", x = "X-Shampoo", y = "Y-axis label")
> # Create a line graph
> ggplot(data, aes(x = shampoo, y = facewash)) +
+  geom_line(color = "red") +
+  labs(title = "Line Graph", x = "Shampoo", y = "Facewash")
> # Create a scatterplot
> ggplot(data, aes(x = shampoo, y = facewash)) +
+  geom_point(color = "orange") +
+  labs(title = "Scatterplot", x = "Shampoo", y = "Facewash")
> # Create a pie chart
> pie(data$total_units, labels = data$labels, main = "Pie Chart")
```

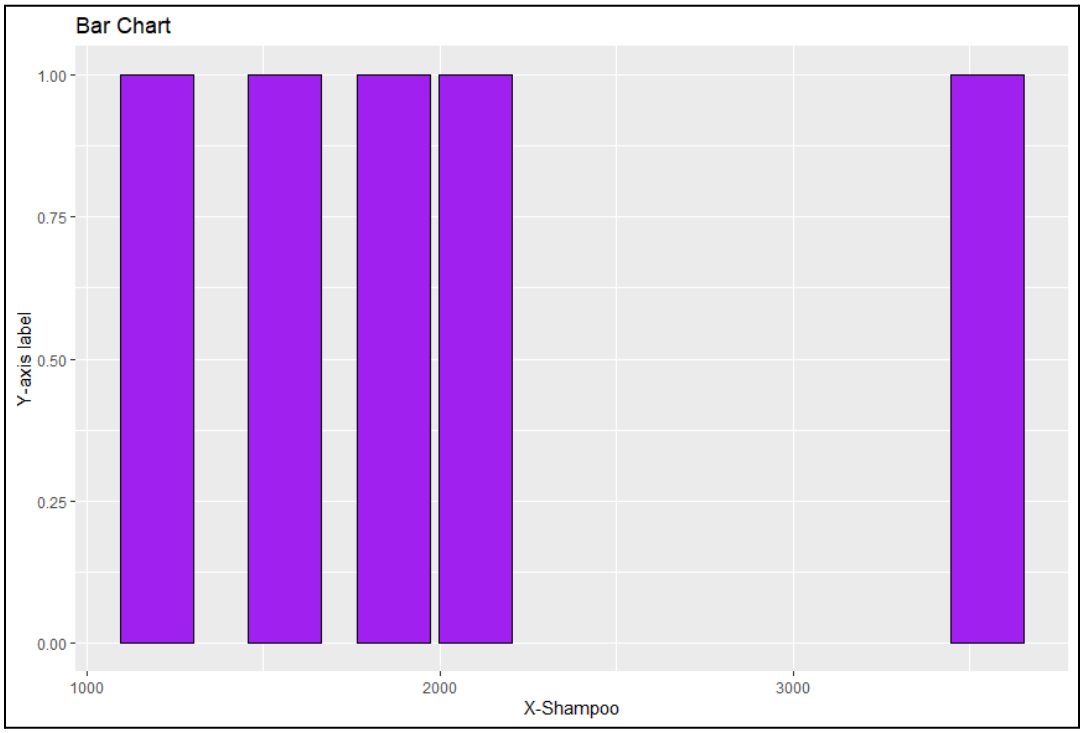
Histogram using Ggplot2:.



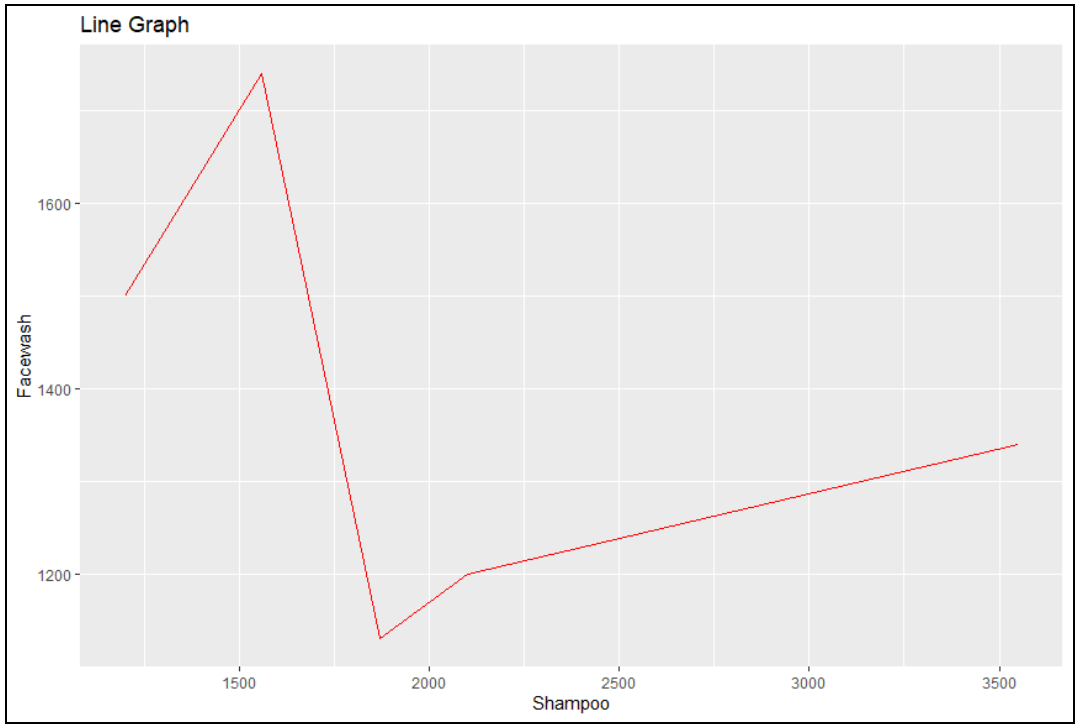
Boxplot using Ggplot2:



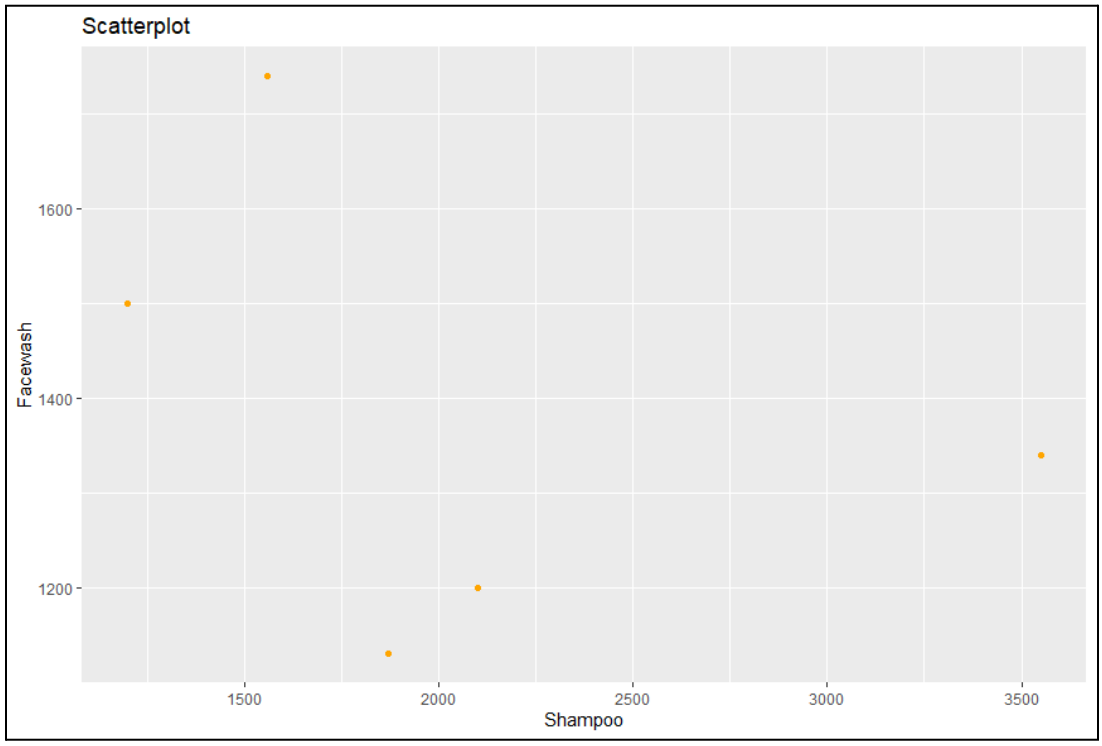
BarChart using Ggplot2:



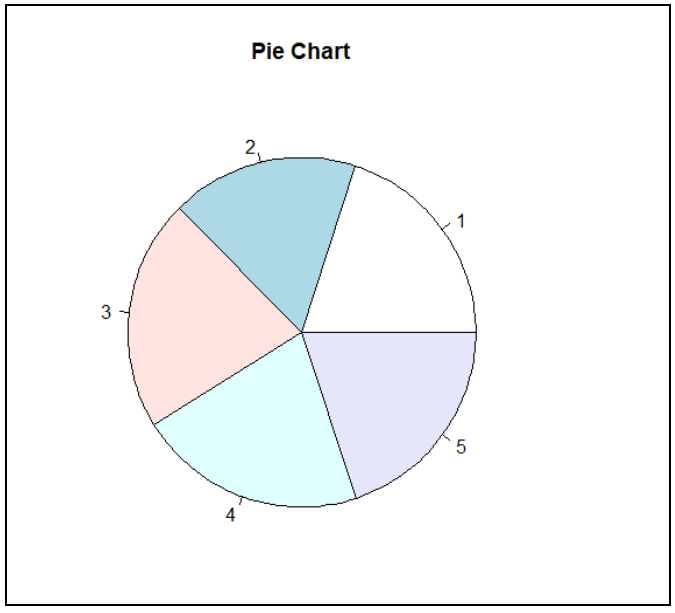
Line Graph using Ggplot2:



ScatterPlot using Ggplot2:



PieChart using Ggplot2:



Conclusion: In this practical we learned implementation of data visualization using ggplot2.