Project Information

• **Title**: Laptop Dataset

• **Name:** Sreejaa R G

• **DA/DS**: May-2025

• **Batch Number:** RP-36

• **Online/Offline:** Offline

• **Roll Number**: 16525CBRE34

1. Introduction

The rapid evolution of technology and increased digital dependency have led to a growing demand for laptops across various consumer segments. Understanding the key factors that influence laptop pricing and consumer preferences is crucial for manufacturers, retailers, and buyers alike.

This project focuses on analyzing a dataset of laptops to explore patterns, trends, and relationships among key attributes such as brand, processor, RAM, storage, and price. By performing Exploratory Data Analysis (EDA), we aim to extract valuable insights that explain how different features contribute to a laptop's market value and popularity.

The dataset used in this project contains detailed specifications of laptops listed on an online retail platform. Each row represents a unique laptop model, and the columns include specifications like company, type, screen size, CPU, RAM, storage, GPU, operating system, weight, and price.

2. Aim

The primary aim of this project is to analyze the specifications and pricing of laptops to uncover meaningful insights that can inform purchasing decisions, marketing strategies, and product development.

Through comprehensive Exploratory Data Analysis (EDA), the project seeks to:

- Understand how key features such as RAM, processor type, storage, and brand influence laptop prices.
- Identify the top-selling companies and most preferred configurations (e.g., RAM size, screen size).
- Detect patterns in consumer preferences based on technical specifications.
- Prepare the dataset for further predictive modeling (e.g., price prediction or recommendation systems).

3. Business Problem / Problem Statement

In today’s competitive tech market, consumers are presented with a vast array of laptops from different brands, configurations, and price ranges. For manufacturers and retailers, understanding which specifications drive customer preferences and pricing is crucial for maximizing sales and market share.

However, businesses often face challenges such as:

- Uncertainty about which features (RAM, CPU, brand, etc.) matter most to consumers
- Difficulty pricing laptops competitively without compromising margins
- Lack of insight into market trends and shifts in consumer demand

This project addresses these challenges by analyzing a real-world dataset of laptops to extract valuable insights. By examining how features like brand, processor type, RAM, storage, and screen size influence the sales volume and pricing, the project aims to answer key business questions such as:

- Which brands and specifications are most popular among buyers?
- How do hardware features affect laptop pricing?
- What are the trends in high vs. low-selling laptops?

4. Project Workflow

1. Data Collection

- The dataset was obtained from an online laptop retail platform.
- It includes detailed specifications of laptops such as brand, type, screen size, processor, RAM, storage, GPU, operating system, weight, and price.

2. Data Understanding

- Reviewed the dataset structure, dimensions, and data types.
- Identified key variables relevant to the analysis.
- Performed an initial exploratory analysis to get a sense of data distribution and patterns.

3. Data Cleaning and Preprocessing

- Handled missing values by removing incomplete rows.
- Removed duplicate records to ensure accuracy.
- Transformed categorical and string-based fields into numeric formats (e.g., RAM, storage, weight).
- Extracted relevant components from complex fields like CPU and Memory.

4. Feature Engineering (Derived Metrics)

- Created new variables such as:
    - Numeric RAM values
    - Weight and screen resolution categories

5. Data Filtering and Subsetting

- Filtered the dataset to remove extreme outliers and irrelevant data points.
- Created subsets of data for specific queries, such as top-selling brands or laptops with high RAM.

6. Exploratory Data Analysis (EDA)

- Conducted univariate, bivariate, and multivariate analysis.
- Used visualizations (histograms, bar charts, box plots, scatter plots) to understand trends, relationships, and distribution.
- Generated insights on how specifications affect laptop pricing and popularity.

7. Insights and Reporting

- Summarized the findings in a clear and actionable format.
- Highlighted business implications such as popular configurations, pricing trends, and brand performance.

5. Data Understanding

1. Dataset Overview

- **Format:** Tabular (structured data)
- **Dimensions:**
  - **Rows:** 1,300 (each row represents a laptop)
  - **Columns:** 12 (each column represents a specification or attribute)

2. Structure and Data Types

Column Name	Description	Data Type
Company	Laptop brand name (e.g., HP, Dell, Lenovo)	Object (Categorical)
TypeName	Laptop category (e.g., Ultrabook, Gaming, Notebook)	Object (Categorical)
Inches	Screen size in inches	Float
ScreenResolution	Screen resolution details	Object
Cpu	Processor name and model	Object
Ram	RAM size (e.g., "8GB")	Object (converted to int)
Memory	Storage capacity and type (e.g., "256GB SSD")	Object
Gpu	Graphics card model	Object
OpSys	Operating system (e.g., Windows, macOS)	Object

Column Name	Description	Data Type
Weight	Laptop weight in kg (e.g., "1.5kg")	Object (converted to float)
Price	Price of the laptop	Integer

3. Initial Exploratory Analysis

a. Missing Values

- Checked for null or missing values using .isnull().sum()
- Removed rows with missing values in essential columns like Company, TypeName, Price, and Weight.

b. Duplicates

- Identified and removed duplicate entries to ensure data integrity.

```
# To remove the full blank row in dataset
dataset=dataset.dropna(how='all')
```

```
# Remove duplicates
data=dataset.drop_duplicates()
```

```
dataset.shape
```

```
(1273, 12)
```

```
# after removing the duplicates and blank rows
pd.isnull(dataset).sum()
```

```
Unnamed: 0           0
Company              0
TypeName            20
Inches              53
ScreenResolution     0
Cpu                  0
Ram                 14
Memory               0
Gpu                  0
OpSys                0
Weight              14
Price                0
dtype: int64
```

c. Basic Descriptive Statistics

dataset.describe()

```
dataset.describe()
```

	Inches	Price	RamGB	Weight_kg
count	1273.000000	1273.000000	1273.000000	1273.000000
mean	15.420031	59955.814073	8.476826	2.154281
std	4.952272	37332.251005	5.556143	1.411948
min	2.100000	9270.720000	1.000000	0.000080
25%	14.000000	31914.720000	4.000000	1.500000
50%	15.600000	52161.120000	8.000000	2.040000
75%	15.600000	79333.387200	8.000000	2.330000
max	111.800000	324954.720000	64.000000	29.000000

d. Data Type Conversion

- Converted Inches from object to numeric
- Converted Weight from string to float

```
# to change the datatype of inches(object to numeric)
dataset['Inches']= pd.to_numeric(dataset['Inches'],errors='coerce')
```

```
#fill the blankspace of inches
dataset['Inches']=dataset['Inches'].fillna(dataset['Inches'].median())
```

6. Data Cleaning

Detection:

- Used .isnull().sum()

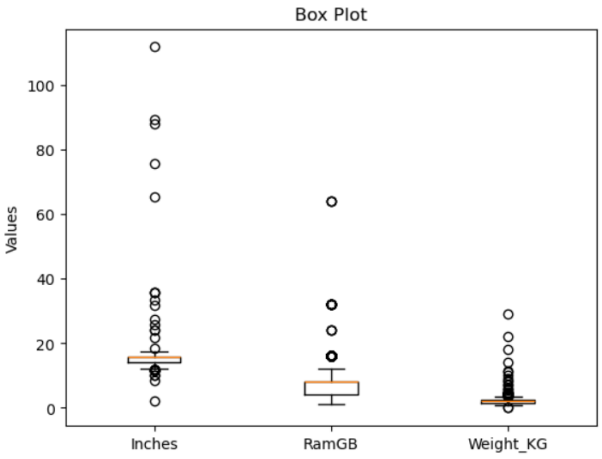
```
Unnamed: 0           0
Company              0
TypeName            20
Inches              53
ScreenResolution     0
Cpu                  0
Ram                 14
Memory               0
Gpu                  0
OpSys                0
Weight              14
Price                0
dtype: int64
```

Outlier Detection

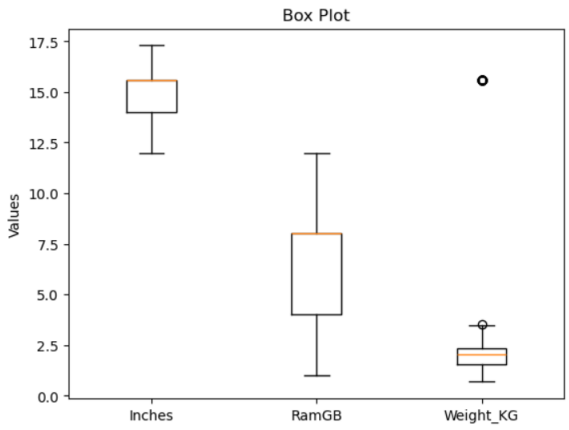
Detection:

- Used .describe() and visual plots (box plots, histograms) to spot unusually high or low values in numeric fields such as:
    - Ram
    - Weight
    - Inches

Before remove outlier:



After remove outlier:



C. Handling Inconsistent Values

1.Converted RAM to Numeric:

```
# To fill the Ram
dataset['Ram']=dataset['Ram'].fillna(dataset['Ram'].mode()[0])
```

```
# To extract the only numeric value
dataset['RamGB'] = dataset['Ram'].str.extract(r'([0-9.]+)')
```

```
# To change the dataype (object to int)
dataset['RamGB']=dataset['RamGB'].astype(int)
```

2.Cleaned Weight Field:

```
# To change the datatype of weightval(object-float)
dataset['Weightval']=dataset['Weightval'].astype(float)
```

7.Descriptive Analysis

Info()

Gives a summary of the dataset structure:

- Total rows and columns
- Column names and data types
- Non-null (filled) values

```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 1273 entries, 0 to 1302
Data columns (total 11 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   Company           1273 non-null   object 
 1   TypeName          1273 non-null   object 
 2   Inches            1273 non-null   float64
 3   ScreenResolution  1273 non-null   object 
 4   Cpu               1273 non-null   object 
 5   Memory            1273 non-null   object 
 6   Gpu               1273 non-null   object 
 7   OpSys             1273 non-null   object 
 8   Price             1273 non-null   int64  
 9   RamGB             1273 non-null   int64  
 10  Weight_kg         1273 non-null   float64
dtypes: float64(2), int64(2), object(7)
memory usage: 119.3+ KB
```

Describe()

Gives summary statistics for numerical columns:

- Mean, Median, Min, Max
- Standard deviation
- 25%, 50%, 75% values (quartiles)

```
dataset.describe()
```

	Inches	Price	RamGB	Weight_kg
count	1273.000000	1273.000000	1273.000000	1273.000000
mean	15.420031	59955.814073	8.476826	2.154281
std	4.952272	37332.251005	5.556143	1.411948
min	2.100000	9270.720000	1.000000	0.000080
25%	14.000000	31914.720000	4.000000	1.500000
50%	15.600000	52161.120000	8.000000	2.040000
75%	15.600000	79333.387200	8.000000	2.330000
max	111.800000	324954.720000	64.000000	29.000000

8.Hypothesis Testing:

Two-Way ANOVA

Two-Way ANOVA (Analysis of Variance) is a statistical method used to examine the effect of two independent categorical variables (called factors) on one continuous dependent variable.

It also checks whether there is an interaction between the two factors — that is, whether the effect of one factor depends on the level of the other factor.

```
import pandas as pd
import statsmodels.api as sm
from statsmodels.formula.api import ols
import warnings
warnings.filterwarnings('ignore')

# Drop rows with missing values in relevant columns
dataset= dataset.dropna(subset=['Company', 'TypeName', 'Price'])

# Define the model: Price explained by Company and Category, including interaction
model = ols('Price ~ C(Company) + C(TypeName) + C(Company):C(TypeName)', data=dataset).fit()

# Perform two-way ANOVA
anova_table = sm.stats.anova_lm(model, typ=2)

print("Two-Way ANOVA Results:")
print(anova_table)
```

```
Two-Way ANOVA Results:
                            sum_sq      df             F        PR(>F)
C(Company)            -2.469758e-01    18.0 -1.888354e-11  1.000000e+00
C(TypeName)                     NaN     5.0           NaN           NaN
C(Company):C(TypeName)  5.636613e+11    90.0  8.619404e+00  3.128255e-41
Residual                8.646602e+11  1190.0           NaN           NaN
```

```
alpha = 0.05

for factor in anova_table.index:

    p = anova_table.loc[factor, 'PR(>F)']
    if p < alpha:
        print(f" {factor}: Significant effect (p = {p:.4f}) - Reject Ho")
    else:

        print(f" {factor}: Not significant (p = {p:.4f}) - Fail to reject Ho")
```

```
C(Company): Not significant (p = 1.0000) - Fail to reject Ho
C(TypeName): Not significant (p = nan) - Fail to reject Ho
C(Company):C(TypeName): Significant effect (p = 0.0000) - Reject Ho
Residual: Not significant (p = nan) - Fail to reject Ho
```

9. Exploratory Data Analysis (EDA)

EDA stands for Exploratory Data Analysis. It is the process of analyzing and visualizing data to:

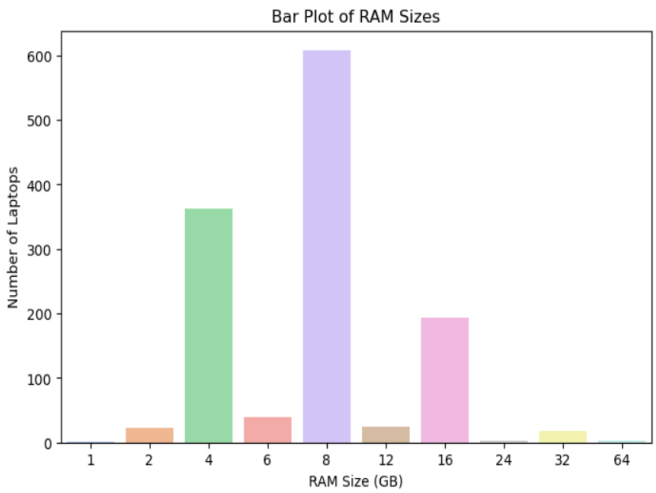
- Understand the structure and contents of a dataset
- Identify patterns, trends, or relationships
- Detect missing values, outliers, or errors
- Prepare data for further analysis or machine learning

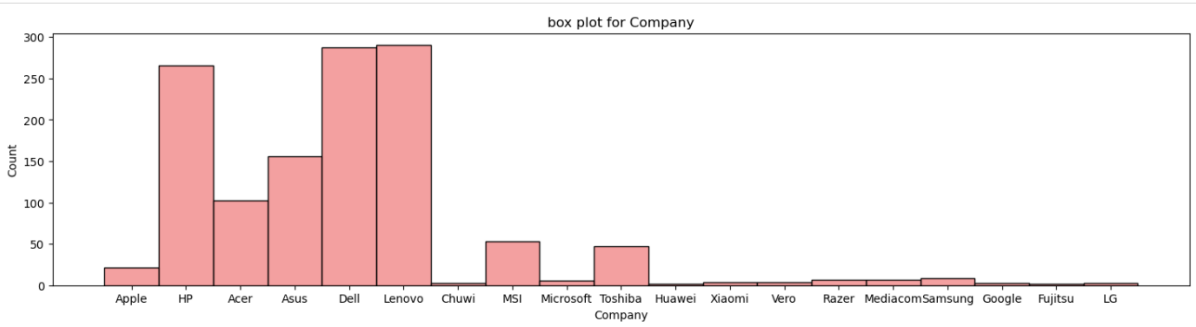
Univariate Analysis:

Univariate analysis is the simplest form of data analysis where only one variable is analyzed at a time.

Example:

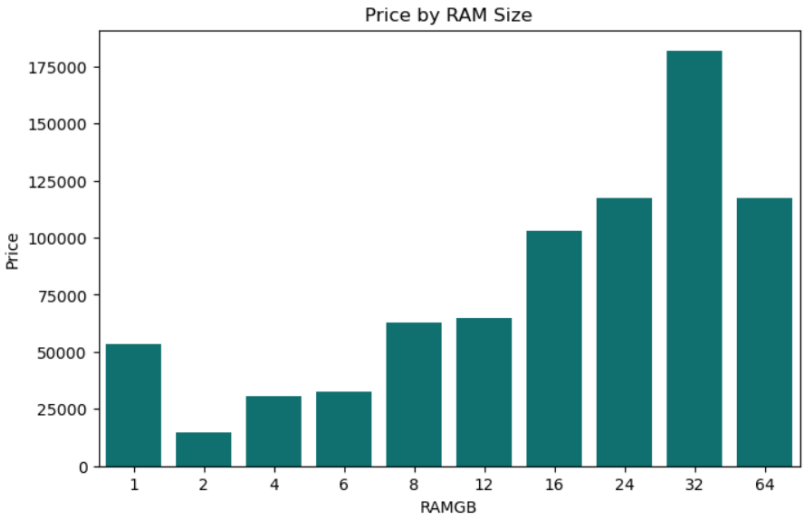
- Ram
- Price

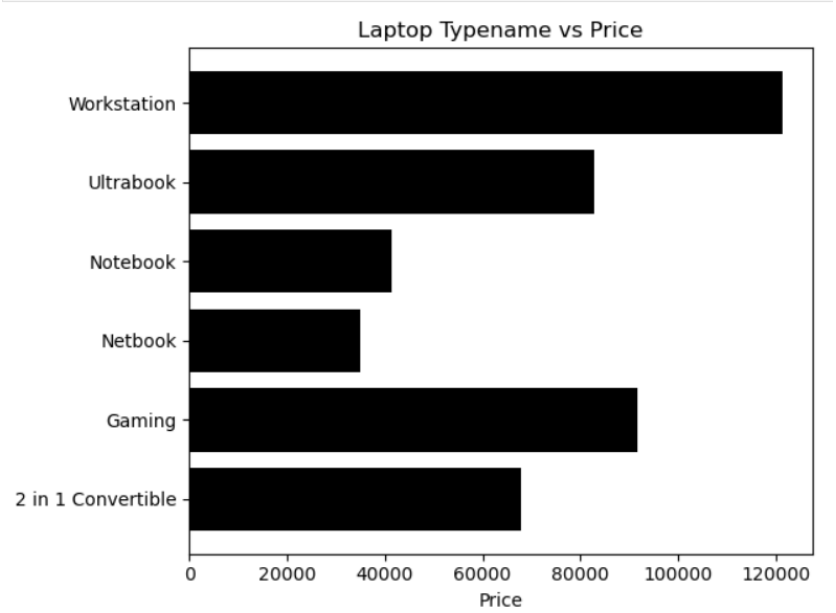




Bivariate Analysis

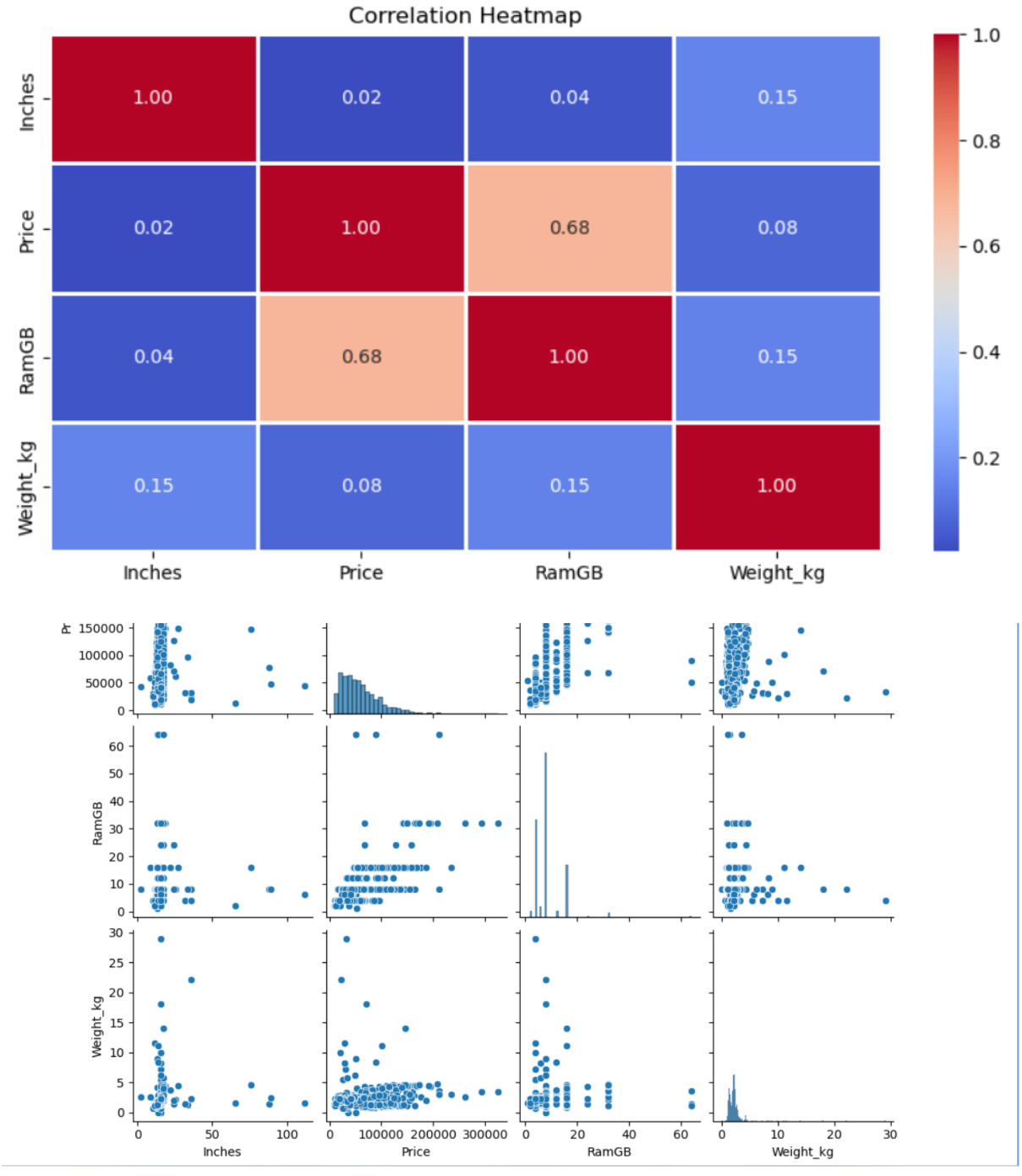
Bivariate analysis means studying two variables at the same time to see if they are related.





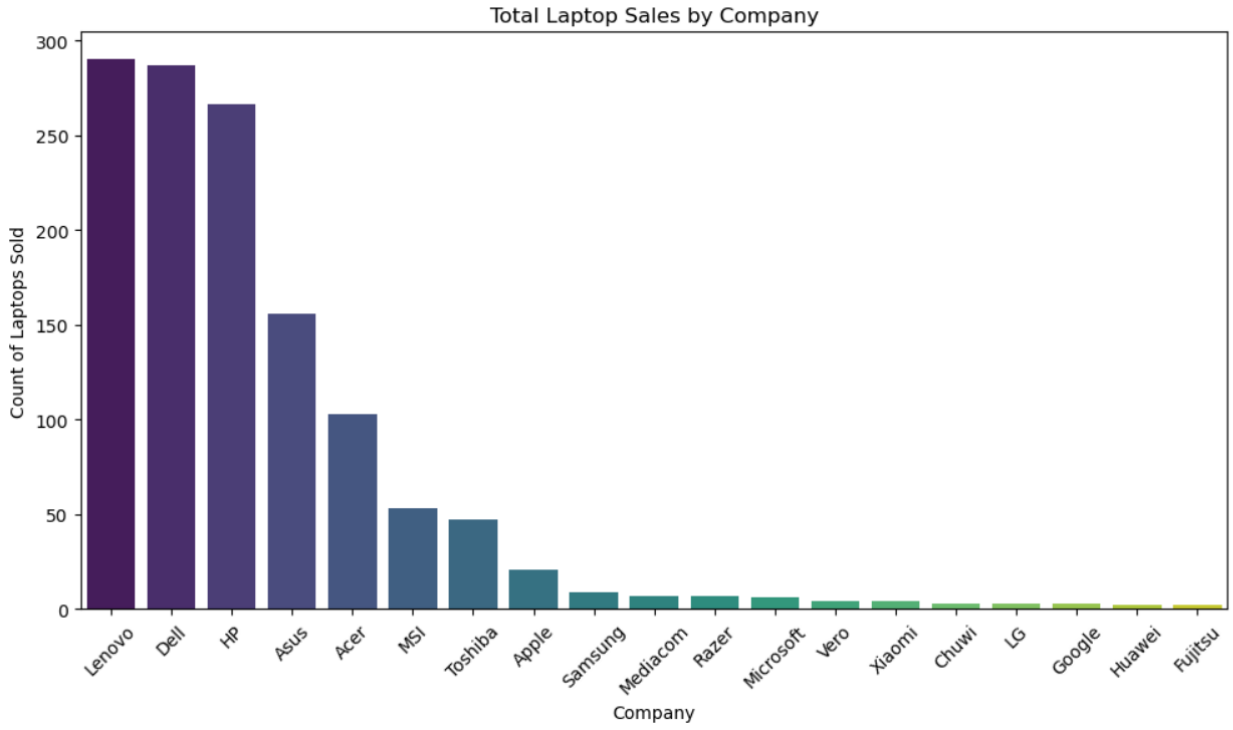
Multivariate Analysis

- Multivariate means involving more than two variables at the same time.
- In data analysis and statistics, multivariate analysis studies relationships and patterns among three or more variables simultaneously, to understand how they interact with each other.



10. Overall Insights from Analysis

- Laptops with 32 GB RAM are the most sold among all configurations.
- Laptops with 2 GB RAM have the lowest sales
- Top-Selling Company:
- The company with the highest number of laptop sales is e.g., HP / Dell / Lenovo].



11.Conclusion

The analysis of the laptop dataset involved univariate, bivariate, and multivariate approaches to uncover the key factors affecting pricing and consumer preferences. Univariate analysis highlighted the most common configurations, such as 8 GB RAM and price ranges concentrated around mid-tier laptops. Bivariate analysis revealed strong relationships between features like RAM size, SSD presence, and price, showing how these variables influence customer choice and pricing. Multivariate analysis further demonstrated that the combined effects of brand, processor type, RAM, and storage significantly impact laptop prices, with premium brands commanding higher prices regardless of specs. The market trend indicates a clear shift away from low-performance laptops, emphasizing consumer demand for better hardware. Based on these findings, manufacturers and retailers should prioritize popular configurations like 8 GB or higher RAM with SSDs while aligning pricing strategies with brand positioning. Future steps include building predictive models, performing customer segmentation, and analyzing temporal trends to better capture evolving market preferences. Overall, this comprehensive analysis provides valuable, data-driven insights to support strategic decisions in the competitive laptop industry.