**Visualization of Product Stock with Pie Chart**

Summary:

This project offers an extensive data visualization solution for managing inventory using the strong data processing and visualization capabilities of Python. The tool reads product stock data from CSV files and creates interactive pie charts, allowing stakeholders to gain easy visual insights into inventory distribution among various product categories.

--------------------------------------------------------------------------------------------------------------------------------------

1. Project Overview

### 1.1 Objective

To create a robust, optimized Python data visualization tool that converts raw inventory data into effective visual representation, facilitating instant decision-making in inventory management and business intelligence.

### 1.2 Problem Statement

Conventional inventory management systems usually provide data in tabular forms that are hard to interpret at a glance. This project resolves the requirement for:

- \*\*Visual representation\*\* of stock distribution

- \*\*Quick identification\*\* of high and low stock items

- \*\*Percentage-based analysis\*\* of inventory allocation

- \*\*Professional reporting\*\* functionality

--------------------------------------------------------------------------------------------------------------------------------------

2. Technical Architecture

### 2.1 System Design:

[CSV Data Input] → [Data Validation] → [Processing Engine] → [Visualization] → [Output Generation]

### 2.2 Core Components:

Data Layer:

- \*\*Input Format\*\*: CSV files with Product and Stock columns

- \*\*Data Types\*\*: String (Product names), Integer (Stock quantities)

- \*\*Validation\*\*: Auto-column verification and data integrity checks

Visualization Layer:

- \*\*Matplotlib Engine\*\*: Professional-level chart generation

- \*\*Custom Styling\*\*: Corporate-standard color schemes and styling

- \*\*Interactive Elements\*\*: Legends, labels, and percentage displays

3. Implementation Details

### 3.1 Code Architecture

#### \*\*Class-Based Design\*\*

```python

class StockVisualizer:

- Encapsulated functionality

- Memory-optimized with \_\_slots\_\_

- Modular method structure

- Professional error handling

```

#### \*\*Key Methods\*\*

- `load\_data()`: Efficient CSV processing with validation

- `create\_visualization()`: Optimized chart generation

- `print\_statistics()`: Comprehensive data analysis

- `save\_chart()`: High-quality image export

--------------------------------------------------------------------------------------------------------------------------------------

## 6. Results and Deliverables

### 6.1 Project Deliverables

#### \*\*Source Code\*\*

- \*\*`pie\_chart.py`\*\*: Optimized main application (180 lines)

- \*\*`product\_stock.csv`\*\*: Sample dataset with 10 products

- \*\*Comprehensive documentation\*\* and inline comments

#### \*\*Output Products\*\*

- \*\*Interactive pie charts\*\* with professional styling

- \*\*Statistical reports\*\* with key inventory metrics

- \*\*Presentation-quality exports\*\* appropriate for presentations

- \*\*Console analytics\*\* for rapid insights

### 6.2 Sample Output Analysis

#### \*\*Test Dataset Results\*\*

```

Total Stock: 525 units across 10 products

Average Stock: 52.5 units per product

Highest Stock: Mouse (120 units) - 22.9%

Lowest Stock: Camera (15 units) - 2.9%

Stock Distribution: Well-balanced across categories

```

#### \*\*Visual Insights\*\*

- \*\*Mouse and Keyboard\*\* account for 38% of overall inventory

- \*\*Camera and Tablet\*\* need close attention for restocking

- \*\*Monitor and Router\*\* depict moderate inventory levels

- \*\*Smartphone and Headphones\*\* have healthy stocks

--------------------------------------------------------------------------------------------------------------------------------------