Experiment No. 10

Inventory Dashboard – Visualizing Data with Chart.js

Objective

To build a browser-based application that provides a visual representation of inventory statistics through pie and bar charts, utilizing the Chart.js library to facilitate better decision-making in inventory control.

Required Software

- HTML5
- CSS3
- JavaScript (ES6)
- Chart.js (via CDN link)

Overview

Modern inventory systems benefit greatly from visual analytics to monitor and manage stock levels effectively. Chart.js is a flexible, open-source JavaScript library that allows for the easy creation of dynamic and responsive charts.

This project displays inventory items across five key categories using two types of charts:

1. **Pie Chart** – For visualizing proportional shares of each category
2. **Bar Chart** – For comparing item quantities across categories

---

Implementation Details

HTML File: `index.html`

This file sets up the structure of the web page and links the Chart.js library and the JavaScript file responsible for generating the visualizations.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Inventory Dashboard</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      text-align: center;
      padding: 50px;
    }
    canvas {
      margin: 30px auto;
    }
  </style>
</head>
<body>

  <h1>Inventory Dashboard</h1>
  <canvas id="pieChart" width="400" height="400"></canvas>
  <canvas id="barChart" width="400" height="400"></canvas>

  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
  <script src="script.js"></script>
</body>
</html>
```

JavaScript File: `script.js`

This script defines the dataset and sets up the configuration for both chart types.

```
const inventoryData = {
  labels: ['Electronics', 'Clothing', 'Home Appliances', 'Books', 'Toys'],
  datasets: [{
    label: 'Stock Count',
    data: [200, 150, 100, 80, 50],
    backgroundColor: [
      '#FF6384',  // Electronics
      '#36A2EB',  // Clothing
      '#FFCE56',  // Home Appliances
      '#4BC0C0',  // Books
      '#9966FF'   // Toys
```

```
    ]
  }]
};

// Create Pie Chart
const pieContext = document.getElementById('pieChart').getContext('2d');
const pieChart = new Chart(pieContext, {
  type: 'pie',
  data: inventoryData,
  options: {
    responsive: true,
    plugins: {
      title: {
        display: true,
        text: 'Stock Distribution Overview'
      }
    }
  }
});

// Create Bar Chart
const barContext = document.getElementById('barChart').getContext('2d');
const barChart = new Chart(barContext, {
  type: 'bar',
  data: inventoryData,
  options: {
    responsive: true,
    plugins: {
      title: {
        display: true,
        text: 'Stock Quantities by Category'
      }
    },
    scales: {
      y: {
        beginAtZero: true
      }
    }
  }
});
```

Output

The resulting web page features two interactive data visualizations:

- **Pie Chart:** Shows the proportion of items across inventory categories
- **Bar Chart:** Highlights the quantity of items in each category