

## **MODUL IX : Mencetak dokumen PDF menggunakan JavaScript**

### **Kelompok 4**

**Nama :**

1. **Rafif Fernanda Wibowo – 064002200001**
2. **Chaesa Namida Arumdapta – 064002200008**
3. **Adrian Alfajri – 064002200009**
4. **Kharisma Maulida Saara – 064002200024**

**Link github :** <https://github.com/kharisma040504/modulprak9pw.git>

**DESKRIPSI MODUL :** Mencetak dokumen PDF menggunakan JavaScript

### **TEORI SINGKAT**

Visualisasi data adalah representasi grafis dari data dan informasi. Dengan menggunakan elemen visual seperti grafik, diagram, dan peta, visualisasi data memudahkan pemahaman data yang kompleks dan besar. Grafik visualisasi membantu mengidentifikasi pola, tren, dan wawasan yang mungkin tersembunyi dalam data mentah.

Langkah-langkah dasar dalam visualisasi data:

1. Pengumpulan Data: Mengumpulkan data dari sumber yang relevan.
2. Pembersihan Data: Mengolah dan membersihkan data untuk memastikan kualitas dan konsistensi.
3. Pemilihan Jenis Grafik: Memilih jenis grafik yang sesuai dengan data yang ingin ditampilkan.
4. Implementasi Grafik: Menggunakan alat atau pustaka tertentu untuk membuat grafik.
5. Interpretasi dan Analisis: Menganalisis hasil visualisasi untuk mendapatkan wawasan dan kesimpulan.

Pustaka JavaScript untuk Visualisasi Data:

1. Chart.js: Pustaka yang ringan dan mudah digunakan untuk membuat grafik sederhana.
2. D3.js (Data-Driven Documents): Pustaka yang kuat dan fleksibel untuk manipulasi data dan membuat visualisasi data yang kompleks.
3. Plotly.js: Pustaka interaktif yang mendukung berbagai jenis grafik.

Kompetensi Dasar:

1. Dasar-dasar JavaScript:
  - Memahami sintaks dan struktur dasar JavaScript.
  - Menggunakan variabel, fungsi, dan kontrol alur dalam JavaScript.

2. Memahami Konsep Dasar Visualisasi Data:
  - Mengetahui pentingnya visualisasi data dalam analisis data.
  - Memahami berbagai jenis grafik dan kapan harus digunakan.
3. Menggunakan Pustaka JavaScript untuk Membuat Grafik:
  - Menginstal dan mengimpor pustaka Chart.js, D3.js, atau Plotly.js.
  - Membuat dan mengkonfigurasi grafik menggunakan pustaka-pustaka tersebut.
  - Menggabungkan grafik ke dalam aplikasi web atau proyek JavaScript.
4. Menganalisis Data dengan Visualisasi:
  - Menginterpretasikan hasil visualisasi untuk mengidentifikasi pola dan tren.
  - Menyajikan hasil analisis dalam bentuk yang mudah dipahami oleh audiens.

## ELEMEN KOMPETENSI

### Latihan 1

1.1 Buat file HTML sederhana sebagai titik awal:

```
<!DOCTYPE html>

<html>
<head>
  <title>Cetak PDF dengan jsPDF</title>

  <script
src="https://cdnjs.cloudflare.com/ajax/libs/jspdf/2.4.0/jspdf.umd.min.js"></script>
</head>
<body>
  <button id="generate-pdf">Generate PDF</button>

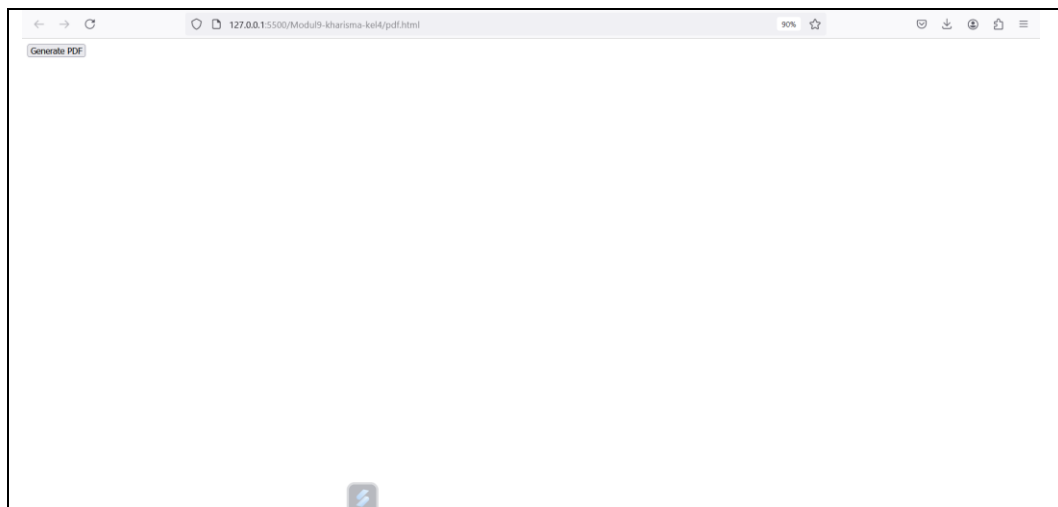
  <script src="script.js"></script>
</body>
</html>
```

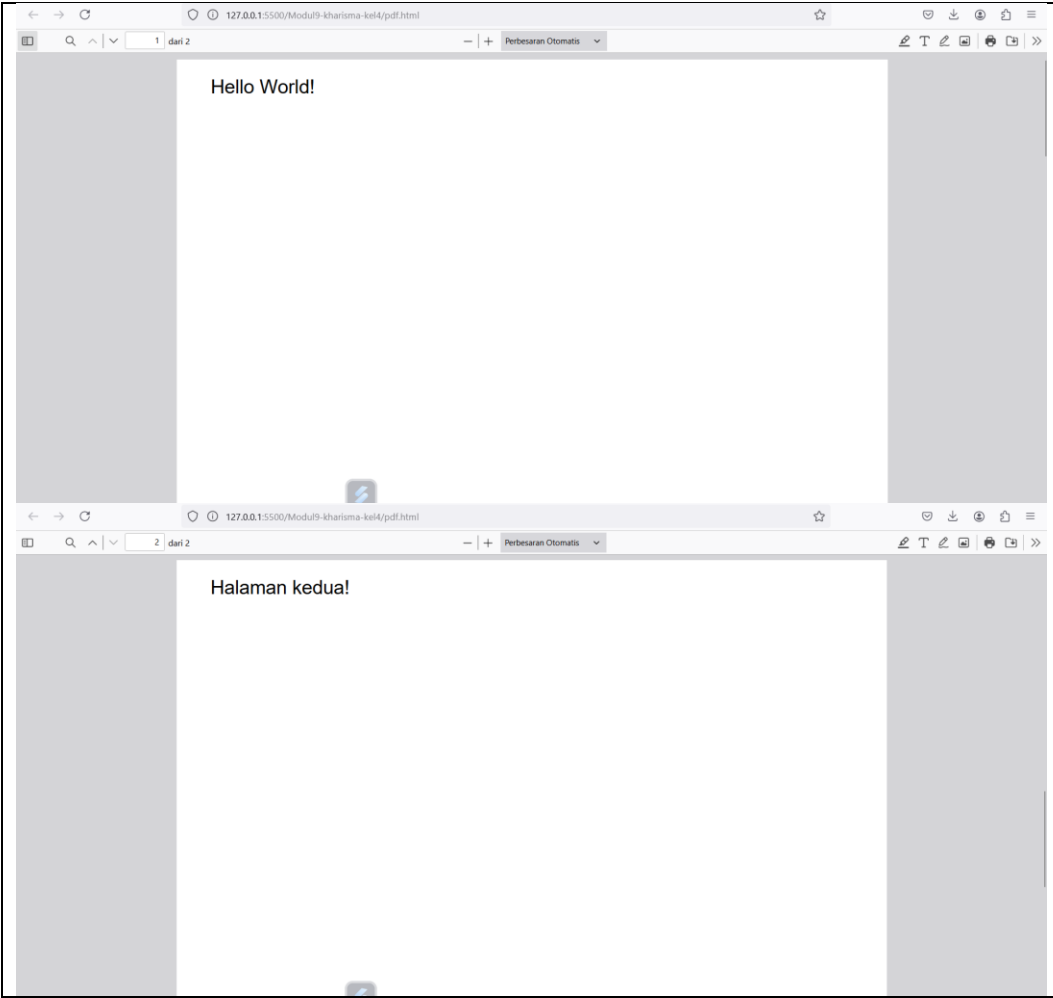
1.2 Buat file script.js dan tulis kode berikut:

```
document.getElementById('generate-pdf').addEventListener('click', function() {
  const { jsPDF } = window.jspdf;
  const doc = new jsPDF();
```

```
doc.text("Hello World!", 10, 10);  
  
doc.addPage();  
  
doc.text("Halaman kedua!", 10, 10);  
  
  
// Tampilkan PDF di browser  
doc.output('dataurlnewwindow');  
  
  
// Atau, gunakan kode ini untuk langsung mencetak PDF  
// doc.autoPrint();  
  
// window.open(doc.output('bloburl'), '_blank');  
});
```

### 1.3 Simpanlah ke dalam Folder yang sama dengan diberi nama (Modul 9 – Nama Mahasiswa)





## Latihan 2

### Instruksi

#### 1. Membuat Grafik

- Tambahkan sebuah grafik bar ke dalam dokumen PDF. Grafik ini harus menunjukkan data penjualan produk selama satu minggu.
- Gunakan array untuk menyimpan data penjualan (misal: [150, 123, 180, 240, 350, 210, 190]).
- Gunakan perpustakaan jsPDF untuk menambahkan grafik ke dalam PDF.

#### 2. Membuat Tabel

- Buat tabel yang menampilkan informasi detail penjualan setiap hari.
- Tabel harus memiliki kolom untuk "Hari", "Produk Terjual", dan "Pendapatan".
- Isi tabel dengan data yang sesuai, dan pastikan setiap baris dalam tabel terhubung dengan data di grafik.

#### 3. Penampilan dan Format

- Pastikan grafik dan tabel disusun dengan rapi dalam halaman PDF.
- Gunakan styling (seperti font size, border tabel, warna bar pada grafik) untuk membuat output lebih menarik.

#### 4. Fungsi Ekstra

- Tambahkan kemampuan untuk mengunduh PDF dengan menekan sebuah tombol yang terdapat di halaman HTML.

#### 5. Dokumentasi

- Tambahkan komentar pada kode untuk menjelaskan fungsi dari masing-masing bagian kode yang penting.
- Tulis dokumentasi singkat yang menjelaskan bagaimana data diolah dan ditampilkan dalam grafik dan tabel.

#### 6. Tes dan Debugging

- Lakukan tes untuk memastikan bahwa grafik dan tabel muncul dengan benar di PDF.
- Periksa apakah ada kesalahan atau bug dalam kode dan perbaiki sesuai kebutuhan.

### Output yang Diharapkan

Sebuah dokumen PDF yang dihasilkan harus menampilkan grafik dan tabel dengan informasi yang jelas dan tampilan yang profesional. Grafik harus secara akurat merepresentasikan data dalam format visual, dan tabel harus mudah dibaca dan informatif.

## Pdf1.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Penjualan Produk</title>
  <link rel="stylesheet" href="styles.css">
  <script src="https://cdnjs.cloudflare.com/ajax/libs/jspdf/2.4.0/jspdf.umd.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/jspdf-
autotable/3.5.17/jspdf.plugin.autotable.min.js"></script>
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
</head>
<body>
  <h1>Penjualan Produk Mingguan</h1>
  <canvas id="salesChart" width="400" height="200"></canvas>
  <button id="downloadPDF">Download PDF</button>
  <script src="script1.js"></script>
</body>
</html>
```

## Script1.js

```
// Data penjualan
const salesData = [150, 123, 180, 240, 350, 210, 190];
const days = ['Senin', 'Selasa', 'Rabu', 'Kamis', 'Jumat', 'Sabtu', 'Minggu'];

// Membuat grafik batang
const ctx = document.getElementById('salesChart').getContext('2d');
const salesChart = new Chart(ctx, {
  type: 'bar',
  data: {
    labels: days,
    datasets: [{
      label: 'Penjualan Produk',
      data: salesData,
      backgroundColor: 'rgba(255, 105, 180, 0.2)', // Warna pink dengan transparansi
      borderColor: 'rgba(255, 105, 180, 1)', // Warna pink penuh
      borderWidth: 1
    }]
  },
  options: {
    scales: {
      y: {
        beginAtZero: true
      }
    }
  }
});
```

```

// Fungsi untuk menghitung pendapatan
const calculateRevenue = (sales) => sales * 1000; // Contoh: harga produk adalah 1000

// Fungsi untuk membuat dan mengunduh PDF
document.getElementById('downloadPDF').addEventListener('click', () => {
  const { jsPDF } = window.jspdf;

  // Buat dokumen jsPDF baru
  const doc = new jsPDF();

  // Tentukan ukuran font dan judul
  const title = 'Laporan Penjualan Produk Mingguan';
  const fontSize = 18;

  // Set ukuran font
  doc.setFontSize(fontSize);

  // Hitung posisi x untuk teks agar berada di tengah halaman
  const pageWidth = doc.internal.pageSize.getWidth();
  const textWidth = doc.getTextWidth(title);
  const textX = (pageWidth - textWidth) / 2;

  // Tambahkan teks judul di posisi yang telah dihitung
  doc.text(title, textX, 22);

  // Fungsi untuk menggambar grafik bar
  function drawBarChart(doc, x, y, width, height, data, labels) {
    const maxData = Math.max(...data);
    const barWidth = width / data.length;
    const tickInterval = Math.ceil(maxData / 5); // Interval untuk sumbu y

    // Menggambar sumbu y
    for (let i = 0; i <= 5; i++) {
      const yTick = y + height - (i * (height / 5));
      doc.setFontSize(8);
      doc.setTextColor(0, 0, 0);
      doc.text(String(i * tickInterval), x - 10, yTick + 3, { align: 'right' }); // Mengatur posisi x agar
      // lebih jelas
      doc.line(x, yTick, x + width, yTick); // Garis horizontal
    }

    // Garis vertikal untuk sumbu y
    doc.line(x, y, x, y + height);

    // Menggambar grafik bar
    for (let i = 0; i < data.length; i++) {
      const barHeight = (data[i] / maxData) * height;
      doc.setFillStyle(255, 105, 180); // Warna pink untuk batang grafik
      doc.rect(x + i * barWidth, y + height - barHeight, barWidth - 5, barHeight, 'F');
      doc.setFontSize(12); // Ukuran font untuk label hari
      doc.setTextColor(0, 0, 0); // Warna teks
    }
  }

```

```

        doc.text(labels[i], x + i * barWidth + (barWidth - 5) / 2, y + height + 10, { align: 'center' }); //
Menambah jarak y untuk teks hari
    }
}

// Menggambar grafik
drawBarChart(doc, 30, 40, 150, 90, salesData, days);

// Tambahkan tabel dengan jarak yang cukup
const tableData = days.map((day, index) => [day, salesData[index],
calculateRevenue(salesData[index])]);
doc.autoTable({
    head: [['Hari', 'Produk Terjual', 'Pendapatan']],
    body: tableData,
    startY: 150, // Menambahkan jarak antara grafik dan tabel
    theme: 'grid',
    styles: {
        halign: 'center',
        fillColor: [255, 182, 193] // Warna pink untuk latar belakang sel
    },
    headStyles: {
        fillColor: [255, 105, 180] // Warna pink lebih gelap untuk header tabel
    }
});

// Unduh PDF
doc.save('penjualan_produk.pdf');
});

/**
 * Fungsi untuk menggambar grafik bar ke dalam dokumen PDF.
 * @param {Object} doc - Objek jsPDF.
 * @param {number} x - Posisi x untuk grafik.
 * @param {number} y - Posisi y untuk grafik.
 * @param {number} width - Lebar grafik.
 * @param {number} height - Tinggi grafik.
 * @param {Array} data - Array data penjualan.
 * @param {Array} labels - Array label untuk data penjualan.
 */
function drawBarChart(doc, x, y, width, height, data, labels) {
    const maxData = Math.max(...data);
    const barWidth = width / data.length;
    const tickInterval = Math.ceil(maxData / 5); // Interval untuk sumbu y

    // Menggambar sumbu y
    for (let i = 0; i <= 5; i++) {
        const yTick = y + height - (i * (height / 5));
        doc.setFontSize(8);
        doc.setTextColor(0, 0, 0);
        doc.text(String(i * tickInterval), x - 10, yTick + 3, { align: 'right' }); // Mengatur posisi x agar
lebih jelas
    }
}

```



```

    doc.line(x, yTick, x + width, yTick); // Garis horizontal
  }

  // Garis vertikal untuk sumbu y
  doc.line(x, y, x, y + height);

  // Menggambar grafik bar
  for (let i = 0; i < data.length; i++) {
    const barHeight = (data[i] / maxData) * height;
    doc.setFillColor(255, 105, 180); // Warna pink untuk batang grafik
    doc.rect(x + i * barWidth, y + height - barHeight, barWidth - 5, barHeight, 'F');
    doc.setFontSize(12); // Ukuran font untuk label hari
    doc.setTextColor(0, 0, 0); // Warna teks
    doc.text(labels[i], x + i * barWidth + (barWidth - 5) / 2, y + height + 10, { align: 'center' }); //
Menambah jarak y untuk teks hari
  }
}

```

## Styles.css

```

body {
  font-family: 'Roboto', sans-serif;
  display: flex;
  flex-direction: column;
  align-items: center;
  margin: 0;
  padding: 20px;
  background-color: #f4f4f9;
  color: #333;
}

h1 {
  margin-bottom: 20px;
  color: #555;
  font-size: 2em;
}

#salesChart {
  margin-bottom: 30px;
  background-color: #fff;
  border-radius: 8px;
  box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
  padding: 20px;
}

button {
  padding: 12px 24px;
  font-size: 16px;
  font-weight: bold;
  color: #fff;
  background-color: #007bff;
}

```

```

border: none;
border-radius: 4px;
cursor: pointer;
transition: background-color 0.3s ease;
box-shadow: 0 4px 6px rgba(0, 123, 255, 0.3);
}

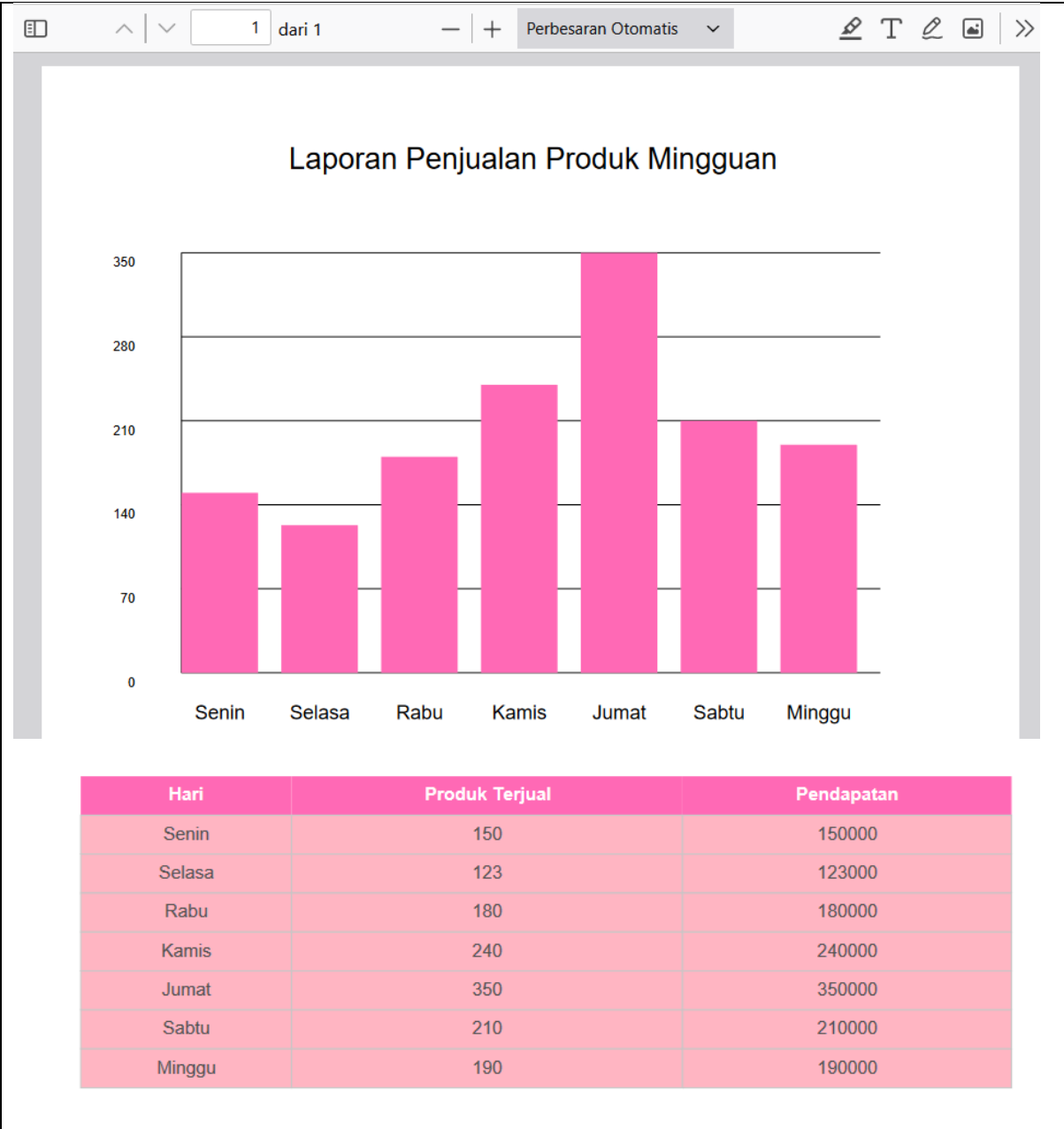
button:hover {
  background-color: #0056b3;
}

button:focus {
  outline: none;
  box-shadow: 0 0 0 3px rgba(0, 123, 255, 0.5);
}

```

## Tampilan





Hari	Produk Terjual	Pendapatan
Senin	150	150000
Selasa	123	123000
Rabu	180	180000
Kamis	240	240000
Jumat	350	350000
Sabtu	210	210000
Minggu	190	190000

## KESIMPULAN

**Berikan kesimpulan berdasarkan apa yang sudah di kerjakan pada modul ini minimal 200 kata**

Praktikum 9 ini bertujuan untuk memperkenalkan dan mempraktikkan cara mencetak dokumen PDF menggunakan JavaScript dengan bantuan pustaka jsPDF. Melalui modul ini, kita belajar bagaimana memanfaatkan JavaScript untuk membuat dokumen PDF yang mencakup grafik dan tabel, serta mengunduhnya dengan mudah. Praktikum ini memberikan pemahaman yang mendalam tentang dasar-dasar visualisasi data dan penerapannya dalam konteks pencetakan dokumen. Langkah-langkah dasar visualisasi data seperti pengumpulan dan pembersihan data, pemilihan jenis grafik, serta interpretasi dan analisis hasil diaplikasikan dalam praktikum ini. Dengan menggunakan data penjualan produk mingguan, peserta membuat grafik bar yang mewakili penjualan harian dan menampilkan tabel yang merinci informasi tersebut. Pustaka JavaScript seperti Chart.js digunakan untuk membuat grafik interaktif, sementara jsPDF digunakan untuk menambahkan grafik dan tabel ke dalam dokumen PDF. Praktikum ini juga melibatkan beberapa tugas yaitu membuat grafik bar menggunakan Chart.js untuk menampilkan data penjualan dalam bentuk grafik bar pada halaman HTML, membuat tabel menggunakan jsPDF untuk merinci hari, produk terjual, dan pendapatan, memastikan grafik dan tabel ditampilkan dengan rapi dan menarik dalam dokumen PDF, menambahkan fungsi untuk mengunduh PDF dengan menekan tombol pada halaman HTML, dan menyertakan komentar dalam kode serta menyusun dokumentasi singkat tentang bagaimana data diolah dan ditampilkan. Proses pengujian dan debugging dilakukan untuk memastikan bahwa grafik dan tabel muncul dengan benar di PDF serta tidak ada kesalahan dalam kode. Tes ini memastikan bahwa output PDF menampilkan informasi dengan jelas dan profesional, dan grafik secara akurat merepresentasikan data penjualan. Hasil akhir dari praktikum ini adalah sebuah dokumen PDF yang menampilkan grafik penjualan produk mingguan dan tabel yang merinci informasi penjualan harian. Dengan menggunakan jsPDF dan pustaka tambahan seperti Chart.js, kita dapat membuat dokumen PDF yang informatif dan menarik dengan relatif mudah. Praktikum ini tidak hanya mengajarkan teknis pembuatan PDF dengan JavaScript tetapi juga menggarisbawahi pentingnya visualisasi data dalam analisis data, memudahkan interpretasi dan penyampaian informasi kepada pembacanya.