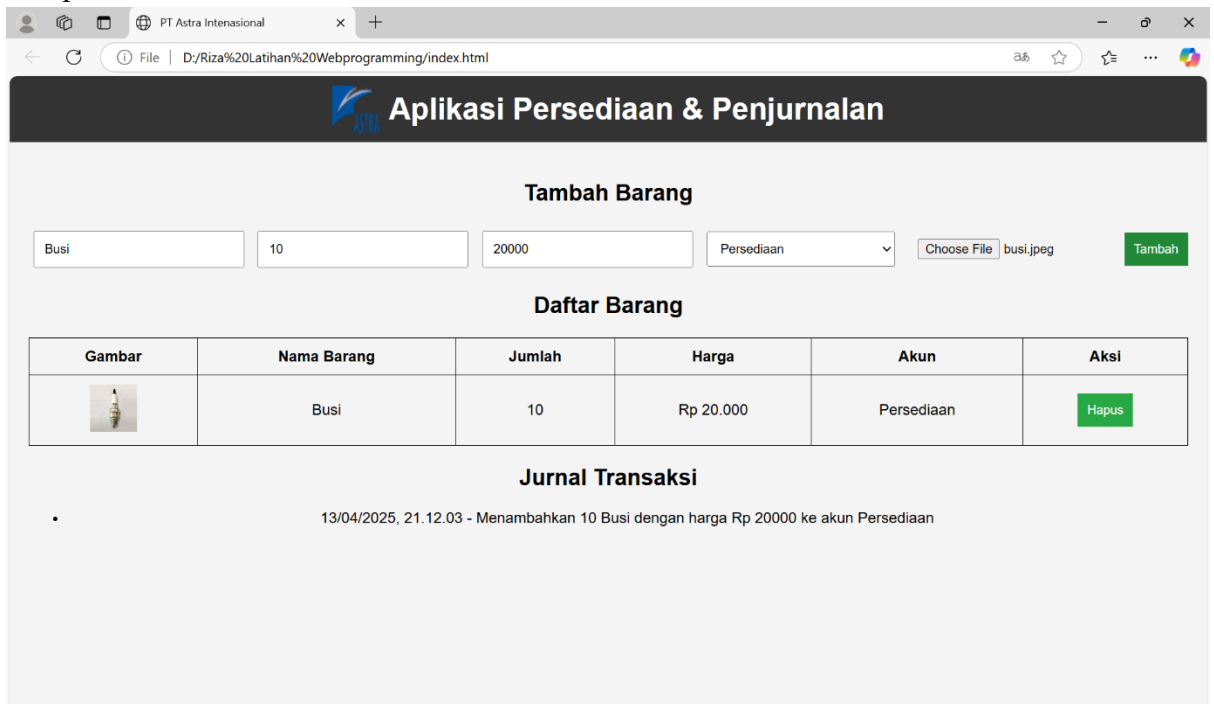


TUGAS HTML, CSS DAN JAVASCRIPT (APLIKASI PERSEDIAAN DAN PENJUALAN)

NAMA : Riza Zhafran Akbar
NIM : 12030123130142
MATA KULIAH : Pengkodean dan Pemrograman
KELAS : D

1. Tampilan



2. Kode HTML

```
<!DOCTYPE html>
<html lang="id">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>PT Astra Intenasional</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <header>
    
    <h1>Aplikasi Persediaan & Penjualan</h1>
  </header>
```

```

<main>
  <!-- Form untuk menambah barang -->
  <section class="inventory-section">
    <h2>Tambah Barang</h2>
    <input type="text" id="itemName" placeholder="Nama Barang">
    <input type="number" id="itemQuantity" placeholder="Jumlah">
    <input type="number" id="itemPrice" placeholder="Harga (Rp)">

    <select id="accountType">
      <option value="Persediaan">Persediaan</option>
      <option value="Kas">Kas</option>
      <option value="Modal">Modal</option>
    </select>

    <input type="file" id="itemImage" accept="image/*">
    <button onclick="addItem()">Tambah</button>
  </section>

  <!-- Tabel untuk daftar barang -->
  <section class="inventory-list">
    <h2>Daftar Barang</h2>
    <table>
      <thead>
        <tr>
          <th>Gambar</th>
          <th>Nama Barang</th>
          <th>Jumlah</th>
          <th>Harga</th>
          <th>Akun</th>
          <th>Aksi</th>
        </tr>
      </thead>
      <tbody id="inventoryTable">
        <!-- Data barang akan ditampilkan di sini -->
      </tbody>
    </table>
  </section>

  <!-- Log jurnal transaksi -->
  <section class="journal-section">
    <h2>Jurnal Transaksi</h2>
    <ul id="journalList">
      <!-- Data jurnal akan ditampilkan di sini -->
    </ul>
  </section>

```

```
        </ul>
    </section>
</main>

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

3. Kode CSS

```
body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 0;
    background-color: #f4f4f4;
    text-align: center;
}
```

```
header {
    background: #333;
    color: white;
    padding: 10px;
    display: flex;
    align-items: center;
    justify-content: center;
}
```

```
.logo {
    width: 50px;
    margin-right: 10px;
}
```

```
h1 {
    margin: 0;
}
```

```
main {
    padding: 20px;
}
```

```
input, select {
    padding: 10px;
    margin: 5px;
    width: 200px;
}
```

```

button {
  padding: 10px;
  background: #28a745;
  color: white;
  border: none;
  cursor: pointer;
}

button:hover {
  background: #218838;
}

table {
  width: 100%;
  border-collapse: collapse;
  margin-top: 20px;
}

table, th, td {
  border: 1px solid black;
}

th, td {
  padding: 10px;
  text-align: center;
}

img {
  width: 50px;
  height: 50px;
}

```

4. Kode JS

```

// Cek apakah ada data sebelumnya di localStorage
let inventory = JSON.parse(localStorage.getItem("inventory")) || [];

// Fungsi untuk menambahkan barang
function addItem() {
  let itemName = document.getElementById("itemName").value;
  let itemQuantity = parseInt(document.getElementById("itemQuantity").value);
  let itemPrice = parseFloat(document.getElementById("itemPrice").value);
  let accountType = document.getElementById("accountType").value;
  let imageInput = document.getElementById("itemImage").files[0];
}

```

```

    if (itemName === "" || isNaN(itemQuantity) || itemQuantity <= 0 ||
    isNaN(itemPrice) || itemPrice <= 0) {
        alert("Masukkan nama, jumlah, dan harga barang yang valid!");
        return;
    }

    // Simpan gambar barang
    let imageUrl = "assets/default.png"; // Gambar default
    if (imageInput) {
        let reader = new FileReader();
        reader.onload = function (e) {
            saveItem(itemName, itemQuantity, itemPrice, accountType, e.target.result);
        };
        reader.readAsDataURL(imageInput);
    } else {
        saveItem(itemName, itemQuantity, itemPrice, accountType, imageUrl);
    }
}

// Fungsi untuk menyimpan barang
function saveItem(name, quantity, price, account, imageUrl) {
    let existingItem = inventory.find(item => item.name === name);
    if (existingItem) {
        existingItem.quantity += quantity;
    } else {
        inventory.push({ name, quantity, price, account, imageUrl });
    }

    saveToLocalStorage();
    updateInventoryTable();
    addJournalEntry(`Menambahkan ${quantity} ${name} dengan harga Rp ${price}
    ke akun ${account}`);
}

// Fungsi untuk menyimpan ke localStorage
function saveToLocalStorage() {
    localStorage.setItem("inventory", JSON.stringify(inventory));
}

// Fungsi untuk memperbarui tabel barang
function updateInventoryTable() {
    let tableBody = document.getElementById("inventoryTable");

```

```

tableBody.innerHTML = "";

inventory.forEach((item, index) => {
  let row = `<tr>
    <td></td>
    <td>${item.name}</td>
    <td>${item.quantity}</td>
    <td>Rp ${item.price.toLocaleString()}</td>
    <td>${item.account}</td>
    <td>
      <button onclick="removeItem(${index})">Hapus</button>
    </td>
  </tr>`;
  tableBody.innerHTML += row;
});
}

// Fungsi untuk menghapus barang
function removeItem(index) {
  let removedItem = inventory[index];
  addJournalEntry(`Menghapus ${removedItem.quantity} ${removedItem.name}
dari akun ${removedItem.account}`);
  inventory.splice(index, 1);
  saveToLocalStorage();
  updateInventoryTable();
}

// Fungsi untuk menambah entri jurnal
function addJournalEntry(entry) {
  let journalList = document.getElementById("journalList");
  let listItem = document.createElement("li");
  listItem.textContent = new Date().toLocaleString() + " - " + entry;
  journalList.prepend(listItem);
}

// Muat data saat halaman dibuka
window.onload = updateInventoryTable;

```