

Report File FULL STACK

Student Name: Aastha

UID: 23BAI70432

Branch: BE-AIT-CSE

Section/Group: 23AIT-KRG-G2

Semester: 5th

Subject Code: 23CSP-339

Subject Name: Full Stack

Aim - To create a responsive banking interface with deposit/withdrawal functionality using HTML5, CSS3, and JavaScript.

Objectives:

1. Design a banking UI with balance display
2. Implement deposit/withdrawal buttons
3. Make the interface responsive using media queries
4. Add interactive elements with JavaScript
5. Validate form inputs

Hardware Requirements:

- Processor: Intel i3/Ryzen 3 or higher
- RAM: 4GB minimum
- Display: 1366x768 resolution

Software Requirements

- VS Code/Sublime Text
- Chrome/Firefox Developer Edition
- Live Server extension
- Git for version control

About the Experiment

This lab introduces:

- Semantic HTML5 structure
- CSS Flexbox/Grid for layout
- Responsive design principles
- DOM manipulation basics

- Form validation patterns

Key Concepts:

- Media queries for responsiveness
- CSS variables for maintainability
- Event handling in JavaScript

Code Implementation

HTML:

```
<div class="bank-interface">
  <h2>Account Balance: <span id="balance">$1000</span></h2>
  <input type="number" id="amount" placeholder="Enter amount" min="0">
  <div class="btn-group">
    <button onclick="deposit()">Deposit</button>
    <button onclick="withdraw()">Withdraw</button>
  </div>
  <p id="message"></p>
</div>
```

CSS:

```
:root {
  --primary-color: #2c3e50;
  --secondary-color: #3498db;
}

.bank-interface {
  display: grid;
  gap: 1rem;
  max-width: 500px;
  margin: 2rem auto;
  padding: 2rem;
  border-radius: 8px;
  box-shadow: 0 2px 10px rgba(0,0,0,0.1);
}
```

```
@media (max-width: 600px) {
  .bank-interface {
    padding: 1rem;
    margin: 1rem;
  }
}
```

JavaScript:

```
let balance = 1000;
```

```

function updateUI() {
  document.getElementById('balance').textContent = `$$${balance}`;
  document.getElementById('amount').value = "";
}

function deposit() {
  const amount = Number(document.getElementById('amount').value);
  if (amount > 0) {
    balance += amount;
    updateUI();
  }
}

function withdraw() {
  const amount = Number(document.getElementById('amount').value);
  if (amount > 0 && amount <= balance) {
    balance -= amount;
    updateUI();
  } else {
    document.getElementById('message').textContent = 'Invalid withdrawal amount!';
  }
}

```

Expected Output

1. Clean banking interface with:
 - Current balance display (\$1000 initial)
 - Input field with validation
 - Deposit/Withdraw buttons
 - Responsive layout (collapses on mobile)
2. Functional balance updates
3. Error messages for invalid inputs

Account Balance: \$2200

124

Deposit

Withdraw