Currency Converter Project Report

1. Introduction

The currency converter project allows users to convert an amount from one currency to another using real-time exchange rates. The application leverages an external API to fetch the latest conversion rates for major currencies. The primary goal is to offer an easy-to-use tool for users to perform accurate currency conversions in real time.

2. Objective

The objectives of this project are:

- To build a user-friendly web application that supports currency conversion between major currencies.
- To fetch real-time exchange rates via an external API.
- To provide an intuitive interface where users can input an amount, select source and target currencies, and view the converted result instantly.

3. Technologies Used

- **HTML**: For the structure and content of the page.
- **CSS**: For styling and layout of the page to make it visually appealing.
- **JavaScript**: To fetch live currency exchange rates from an API and implement the conversion logic.
- **API**: The currency rates are fetched using the `exchangerate-api` with the endpoint: `https://v6.exchangerate-api.com/v6/c97e21e7b04e14199d39cce2/latest/USD`.

4. Features

- **Real-Time Currency Conversion**: Fetches live exchange rates for currencies like USD, INR, JPY, GBP, RUB, etc.
- **Multiple Currencies**: Supports major world currencies for conversion.
- **Responsive Design**: The application is designed to work across various devices such as desktops, tablets, and smartphones.
- **Simple User Interface**: Easy-to-use interface for entering the amount and selecting currencies.

5. How it Works

The user enters an amount they want to convert and selects the source and target currencies. The JavaScript fetches the real-time exchange rates using the API and performs the conversion by multiplying the amount with the appropriate exchange rate. The result is then displayed to the user.

6. Code Snippets

-HTML

index.html styles.css script.js

```html
<div class="container">
    <h2>Currency Converter</h2>
    <div class="converter">
        <input type="number" id="amount" placeholder="Enter amount">
        <select id="fromCurrency"></select>
        <span>to</span>
        <select id="toCurrency"></select>
        <button onclick="convertCurrency()">Convert</button>
    </div>
    <p id="result"></p>
</div>
```

-JAVASCRIPT

index.html styles.css script.js 437m4zces

```javascript
async function convertCurrency() {
    const amount = document.getElementById("amount").value;
    const fromCurrency = document.getElementById("fromCurrency").value;
    const toCurrency = document.getElementById("toCurrency").value;

    const response = await fetch(`https://v6.exchangerate-api.com/v6/${apiKey}/latest/${fromCurrency}`);
    const data = await response.json();
    const rate = data.conversion_rates[toCurrency];
    const convertedAmount = (amount * rate).toFixed(2);
    document.getElementById("result").textContent = `${amount} ${fromCurrency} = ${convertedAmount} ${toCurrency}`;
}
```

Css-

```css
body {
    font-family: Arial, sans-serif;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    background-color: #f4f4f4;
}

.container {
    text-align: center;
    background: white;
    padding: 20px;
    border-radius: 10px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

.converter {
    margin-top: 20px;
}

input, select, button {
    padding: 10px;
    margin: 5px;
    font-size: 16px;
}

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

button:hover {
    background: #218838;
}

```

7. Challenges Faced

- **API Rate Limits**: The API has rate limits, meaning only a certain number of requests can be made in a given time.
- **Error Handling**: Ensuring the application handles API failures or invalid inputs gracefully.
- **Responsive Layout**: Ensuring the design adapts well to all screen sizes.

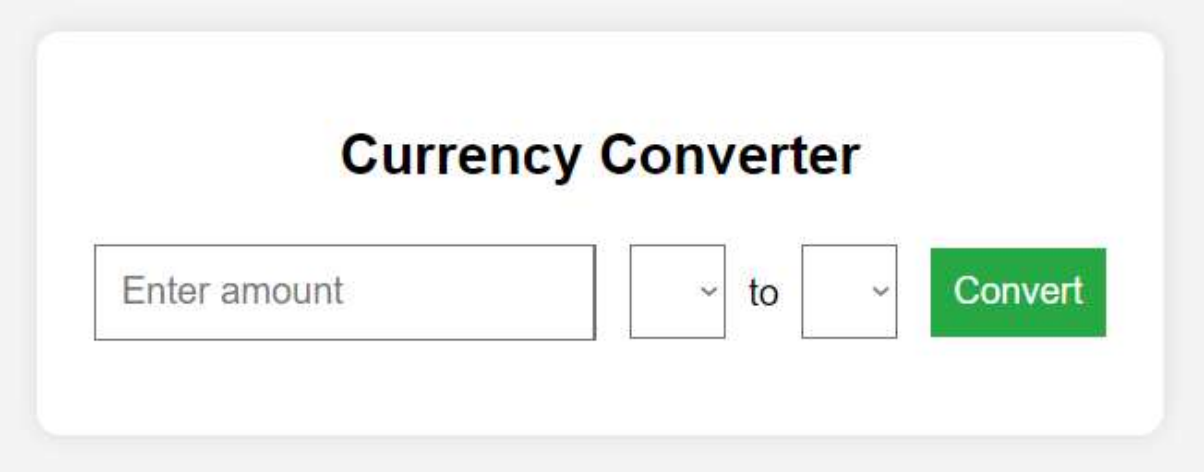
8. Future Improvements

- **Additional Currency Support**: Include more currencies and improve the list of supported countries.
- **Historical Exchange Rates**: Allow users to view historical rates for better context.
- **User Conversion History**: Add the functionality to save and view previous conversion records.

9. Conclusion

This currency converter project effectively demonstrates how to fetch and display real-time exchange rates. It is designed to be simple yet functional, providing users with a seamless experience when converting currencies. The use of HTML, CSS, and JavaScript ensures the application remains lightweight and user-friendly.

Output-



Made by
Utkarsh Jaiswal 22SCSE1040378