AFFORD MEDICALS

TASK 1

Code:

Index.html

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Number Management Service</title>
</head>
<body>
    <h1>Merged Unique Integers</h1>
    <div id="result"></div>

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

Mockresponse.json

```json
{
    "numbers": [1, 2, 3, 5, 8, 13]
}
```

Script.js

```javascript
const corsAnywhereUrl = 'https://cors-anywhere.herokuapp.com/';

// mockResponse.json file
const mockResponseUrl = 'mockResponse.json';
const urls = [mockResponseUrl];

// milli seconds timeout for fetch
const timeout = 500;


//This is the function to merge integers from arrays
const mergeUniqueIntegers = (arrays) => {
    const mergedArray = [].concat(...arrays);
```

```javascript
    const uniqueSet = new Set(mergedArray);
    return [...uniqueSet];
};

// This is the Function to fetch data with timeout
const fetchWithTimeout = (url) => {
    return new Promise((resolve, reject) => {
        const timer = setTimeout(() => {
            reject(new Error('Request timed out'));
        }, timeout);

        fetch(url)
            .then(response => {
                if (!response.ok) {
                    throw new Error(`Network response was not ok:
${response.status}`);
                }
                return response.json();
            })
            .then(data => {
                clearTimeout(timer);
                resolve(data.numbers);
            })
            .catch(error => {
                clearTimeout(timer);
                console.error('Fetch error:', error);
                resolve([]); // Ignore failed requests
            });
    });
};

// HTML element where the result will be displayed
const resultDiv = document.getElementById('result');

// To load mock json respone
fetch('mockResponse.json')
    .then(response => response.json())
    .then(data => {
        const numbersArrays = Object.values(data);
        const mergedNumbers = mergeUniqueIntegers(numbersArrays);
        resultDiv.textContent = JSON.stringify({ numbers: mergedNumbers });
    })
    .catch(error => {
        console.error('Error loading mock JSON:', error);
    });


// Fetch data from Promise.all
Promise.all(urls.map(url => fetch(url)))
```
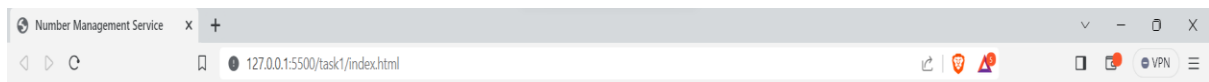
```
    .then(results => {
        return Promise.all(results.map(response => response.json()));
    })
    .then(data => {
        const mergedNumbers = mergeUniqueIntegers(data);
        resultDiv.textContent = JSON.stringify({ numbers: mergedNumbers });
    })
    .catch(error => {
        console.error(error);
    });
```

Output:

Merged Unique Integers

{"numbers":[{"numbers":[1,2,3,5,8,13]}]}

TASK 2

Using post man we are registering and posting the data