import React, { useEffect, useState } from 'react';

import { LineChart, Line, XAxis, YAxis, CartesianGrid, Tooltip, Legend } from 'recharts';

import { CategoryScale, Chart } from 'chart.js';

Chart.register(CategoryScale);

interface StockData {

'Meta Data': {

'1. Information': string;

'2. Symbol': string;

'3. Last Refreshed': string;

'4. Output Size': string;

'5. Time Zone': string;

};

'Time Series (Daily)': {

[date: string]: {

'1. open': string;

'2. high': string;

'3. low': string;

'4. close': string;

'5. volume': string;

};

};

}

async function getData() {

const response = await fetch(`https://www.alphavantage.co/query?function=TIME\_SERIES\_DAILY&symbol=BTCUSDT&outputsize=compact&apikey=6GGYG3DU8XAEGHIC&datatype=json`);

const data: StockData = await response.json();

console.log(data)

return data;

}

const FwdData: React.FC = () => {

const [stockData, setStockData] = useState<StockData | null>(null);

useEffect(() => {

async function fetchStockData() {

try {

const data = await getData();

setStockData(data);

} catch (error) {

console.error("Error fetching stock data:", error);

}

}

fetchStockData();

}, []);

if (!stockData) {

return <div>Loading...</div>;

}

const timeSeries = stockData['Time Series (Daily)'];

// Add a null check to avoid accessing undefined or null object

if (!timeSeries) {

return <div>No stock data available.</div>;

}

const chartData = Object.keys(timeSeries).map(date => ({

date,

closingPrice: parseFloat(timeSeries[date]['4. close'])

}));

return (

<div>

<h2>Stock Closing Prices</h2>

<LineChart width={600} height={300} data={chartData}>

<CartesianGrid stroke="#ccc" />

<XAxis dataKey="date" />

<YAxis />

<Tooltip />

<Legend />

<Line type="monotone" dataKey="closingPrice" stroke="#8884d8" />

</LineChart>

</div>

);

};

export default FwdData;

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

"use client"

import React from 'react';

import FwdData from './stocks';

export default function Display() {

return (

<div>

<h1>Display Stock Data</h1>

<FwdData/>

</div>

);

}