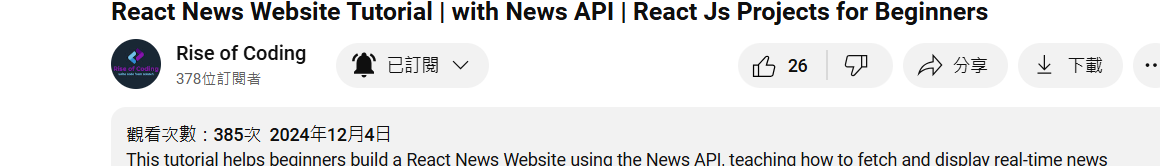
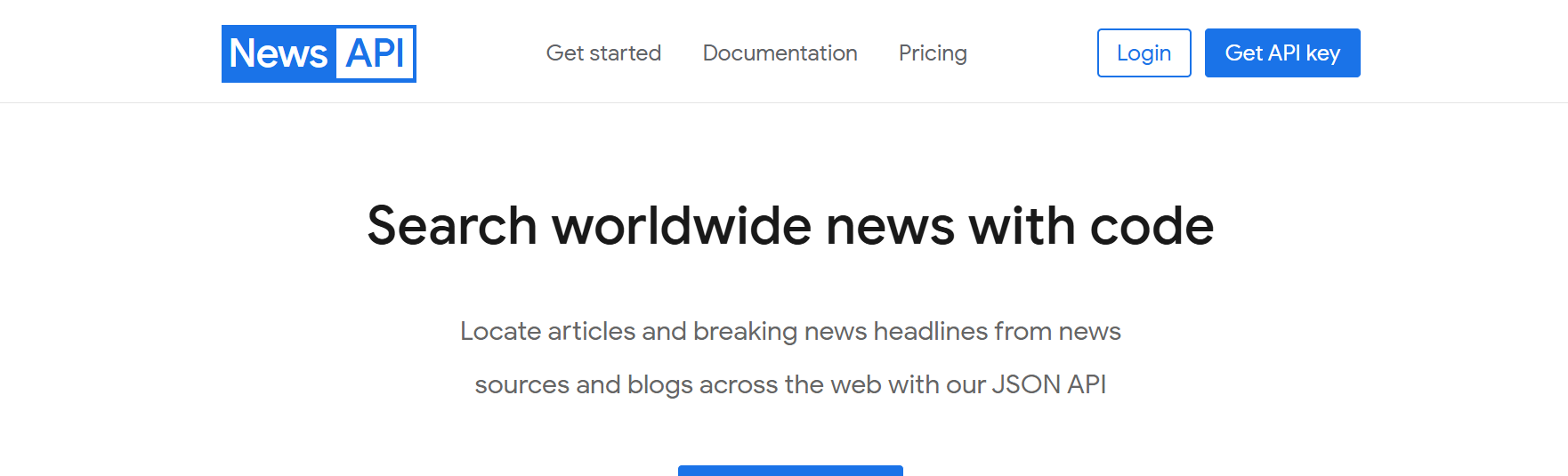
<https://www.youtube.com/watch?v=GFJi6VvGliU>



<https://newsapi.org/>

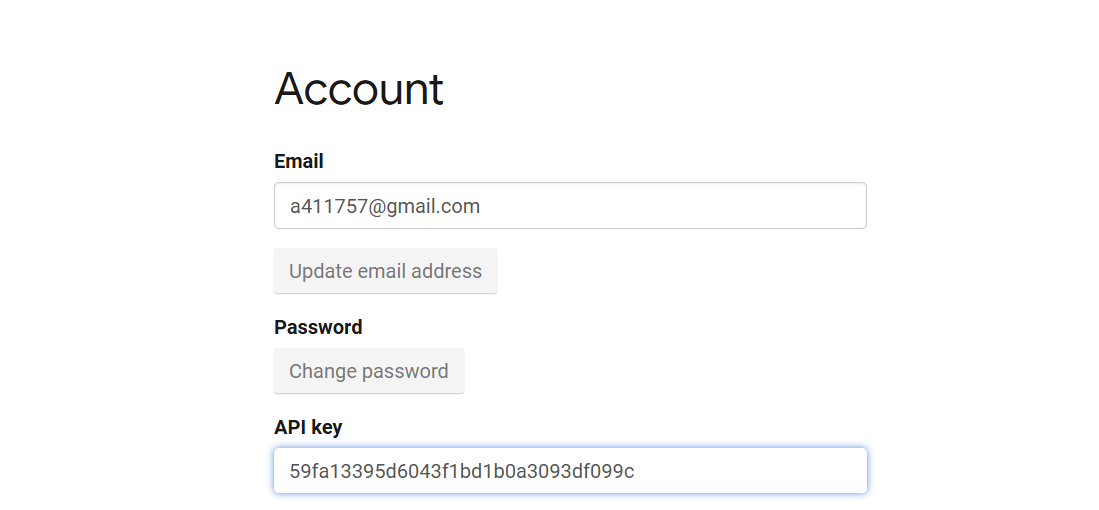


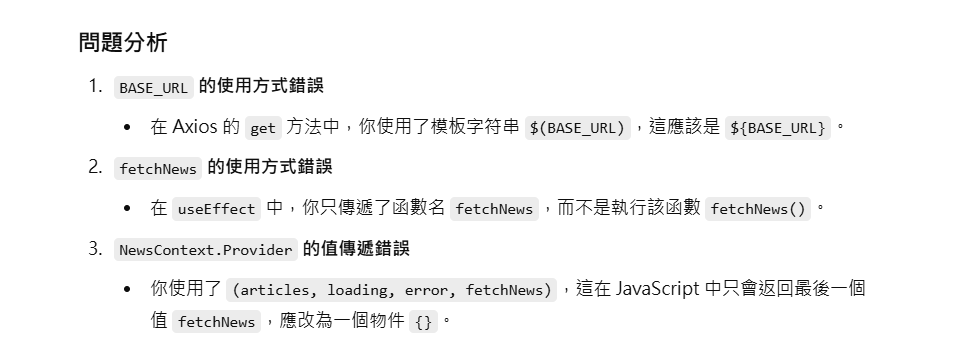
News api帳號和密碼:

[a411757@gmail.com](mailto:a411757@gmail.com)

ao411757

得到API key



使用React的useContext來建立全局的元件抓取api 

Context.jsx完整程式碼:

import axios from "axios";

import { useEffect, useState, useContext, createContext } from "react";

const NewsContext = createContext();

export const useNewsContext = () => useContext(NewsContext);

export const NewsProvider = ({ children }) => {

  const API\_KEY = "59fa13395d6043f1bd1b0a3093df099c";

  const BASE\_URL = "https://newsapi.org/v2";

  const [articles, setArticles] = useState([]);

  const [loading, setLoading] = useState(true);

  const [error, setError] = useState(null);

  const fetchNews = async () => {

    setLoading(true);

    setError(null);

    try {

      const response = await axios.get(`${BASE\_URL}/top-headlines`, {

        params: {

          apiKey: API\_KEY,

          country: "us",

        },

      });

      console.log(response.data.articles);

      setArticles(response.data.articles);

    } catch (err) {

      setError("找不到這新聞");

      console.error(err);

    } finally {

      setLoading(false);

    }

  };

  useEffect(() => {

    fetchNews(); // 注意這裡需要執行函數

  }, []);

  return (

    <NewsContext.Provider value={{ articles, loading, error, fetchNews }}>

      {children}

    </NewsContext.Provider>

  );

};

App.jsx引入useContext.jsx建立的NewsContext.Provider

import Navbar from "./components/navbar/Navbar";

import { NewsProvider } from "./context/context";

export default function App() {

  return (

    <>

   <NewsProvider >

    <Navbar />

    </NewsProvider>

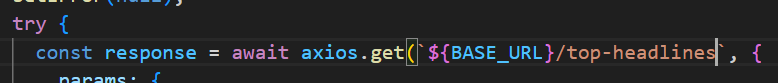
    </>

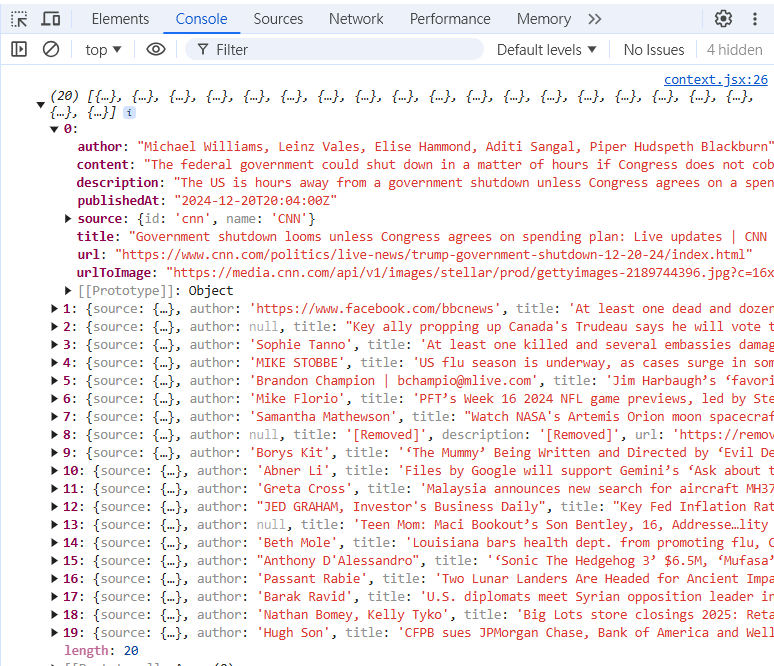
  )

}

檢查輸出:

抓取到20個News api的top-headlines的值





抓到API資料之後透過NewsCard元件渲染

先建立NewsFeed元件抓取到context內的useNewsContext的hook，引用hook內的articles, loading, error傳過來的參數儲存於變數。

判斷假如陣列沒有抓到資料 === 0 的話輸出<p>，如果有抓到資料的話輸出<NewsCard> 元件。

NewsFeed完整程式碼:

import './NewsFeed.css'

import { useNewsContext } from '../../context/context'

import NewsCard from '../NewsCard/NewsCard'

const NewsFeed = () => {

    const {articles, loading, error} = useNewsContext()

  return (

    <div className='news-feed'>

        {articles.length === 0 ? <p className='no-news-message'>沒有新聞</p> : (

            <div className='news-grid'>

               {articles.map((article, index) => (

                <NewsCard key={index} article={article}></NewsCard>

               ))}

                </div>

        )}

        NewsFeed</div>

  )

}

export default NewsFeed

NewsCard完整程式碼:

import './NewsCard.css'

const NewsCard = ({article}) => {

  return (

    <div className='news-card'>

        <img src={article.urlToImage} alt='' className='news-image'/>

        <div className='news-content'>

            <h2>{article.title}</h2>

            <p>{article.description}</p>

            <a href={article.url}></a>

        </div>

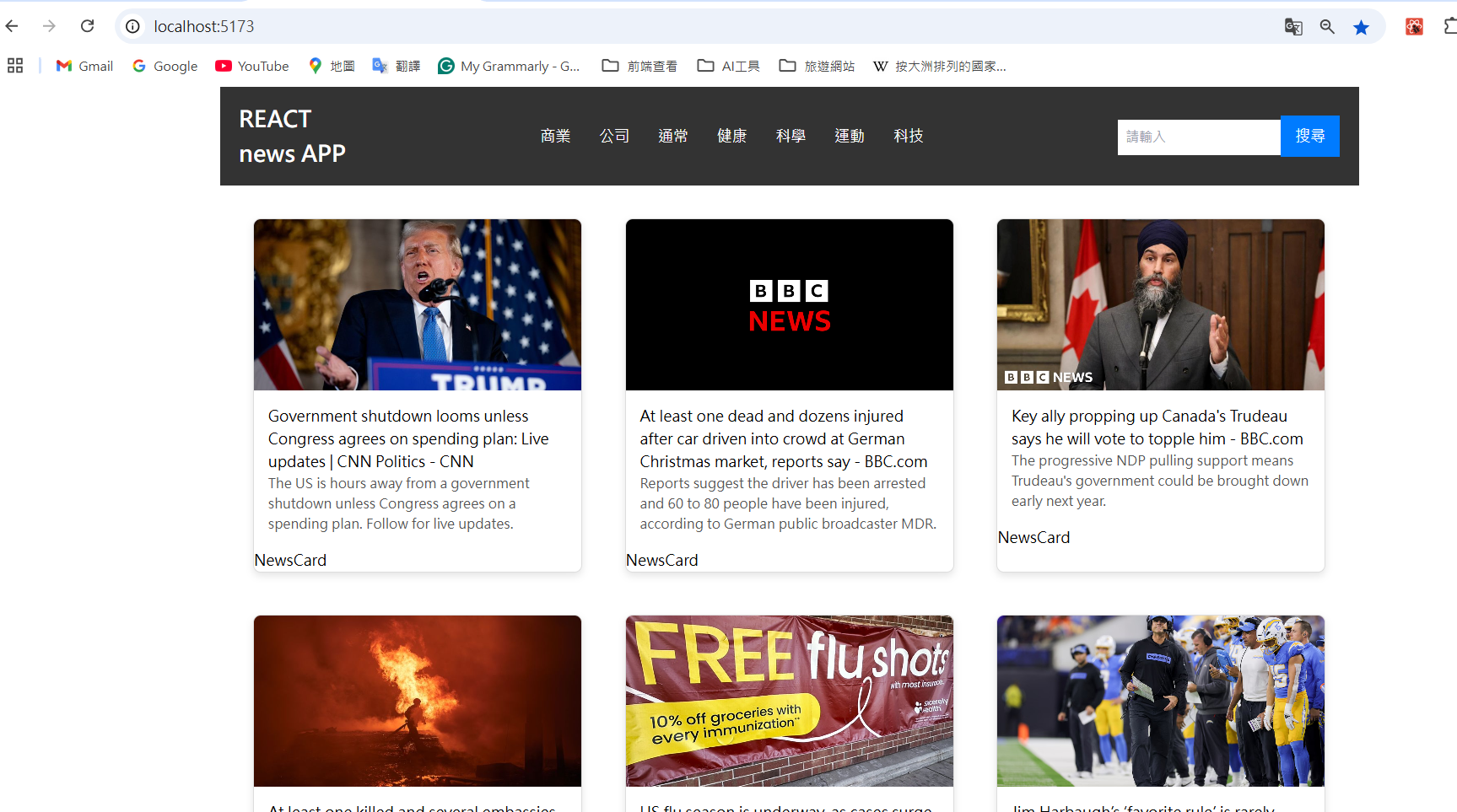
        NewsCard</div>

  )

}

export default NewsCard

輸出:



19:30講到如何實作pagination分頁功能



因為<li> 是引用page-number內的active class，所以前面要加上空格才能輸出

