

2024 年【科學探究競賽-這樣教我就懂】

普高組 成果報告表單

題目名稱：博古通今：古文學在時代的流動中與今科技的邂逅

一、摘要

本報告探討了一項由學生發明的文言文翻譯平台，源於學生在課綱中學習 15 篇古文的需求。透過 Google Site 平台和 JavaScript 程式語言的結合，我們成功地建立效能高、準確性高、並深入了解作者背景的文言文翻譯工具，旨在協助學生理解並更深入地學習古文內容。在這個翻譯機的開發過程中，我們首先選擇了 Google Site 作為報告呈現的平台，以其直觀的介面和易於分享的特點。透過 JavaScript 程式，我們成功整合翻譯平台的功能，使其能夠精準且即時地將文言文轉換成現代文，以助學生理解古文的內容與意旨。學生在課綱中需閱讀 15 篇古文，面對古文難解的挑戰，我們應運而生。這個翻譯機不僅簡化了學習過程，更提供了一個有趣且互動性的工具，使學生能夠輕鬆地理解文言文的文法結構和詞彙意義。並在大科技時代中利用科技保留住歷史的美，巧妙融合科技與文學看似不交集的兩物。

二、探究題目與動機

對於許多現代學生來說，理解和使用文言文是一項具有挑戰性的任務。我們想要建立這個平台，以幫助學生更容易地學習和理解文言文，讓這個寶貴的文化資產不會因難以理解而消失，也助於學生在課業上的表現。然而，在科技意識抬頭的時期，我們注意到文學的流失與不重視，某天赫然發現若將科技結合文學，那必然在時代的碰撞之下產生新奇的火花。

三、探究目的與假設

目的：製作學習平台期望幫助學生了解文章字義與作者本身背景。

假設一：翻譯平台能夠準確翻譯並對應作者背景及經歷。

假設二：JavaScript 程式語言能完整嵌入網站並順利完成翻譯。

四、探究方法與驗證步驟

(一) 研究方法

首先我們用 Google Site 建立起系統最基礎的樣子，並加入美化元素以改善網站看起過於單調的問題，接著便是程式碼的嵌入，分析不同程式碼的優劣後，認為 JavaScript 最為符合該網站所需，在參考各式各樣的學習平台後，我們以學習且創新的模式進行製作，製作出能精確運作的翻譯網站平台。

(二) 程式語言挑選

經過仔細比較三種程式語言(Python、C++、JavaScript)後，我們最終選擇了 JavaScript 作為網站開發的程式語言。JavaScript 在瀏覽器端具有廣泛應用，支援動態效果和互動功能，適合現代網頁開發需求。其豐富的生態系統和社群支持，使我們能夠更靈活地應對網站的不同需求，同時確保順暢的用戶體驗和開發效率。如下表一所示，各程式碼的特色分析表。

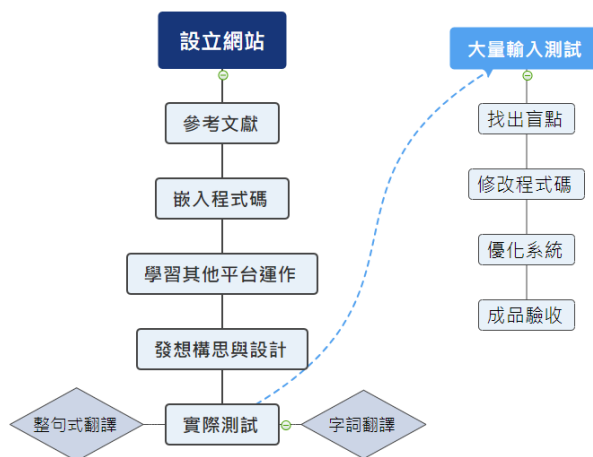
表一：各程式碼的特色分析表

程式碼	Python	C++	JavaScript
發展特色	Python 有簡單的語法，支援跨平台，以簡易的語法編輯出強大的系統。	C++ 是一種靜態類型語言，變數的類型在編譯時即被確定，這將提高代碼的效能和安全性。	JavaScript 主要用於瀏覽器中，有多元的生態系統，且支 Node.js 讓 JavaScript 也可以用於伺服器端開發。
應用領域	Python 主要被使用於數據科學、網絡開發、人工智慧、等高層次應用。	C++ 通常用於遊戲開發和需要高性能的應用，像是系統編成等等。	JavaScript 主要用於前端網頁開發，但也可以通過 Node.js 用於後端開發。

(三) 研究架構



圖一：研究架構圖



圖二：研究流程圖

(四) 研究分析與結果

本研究使用 Google Site 並嵌入 JavaScript 程式碼，此系統的設計理念是希望利用資訊科技，與看似無交集的國文結合，達成博古通今的構思，並針對 15 篇古文進行剖析，透過國文老師傳授作者背景、時空因素，讓科技幫助學生透析理解古文的意義，準確翻譯出古字詞在相對限縮的每篇文章中的意思，實作過程將在以下以圖片配上文字做簡單說明。

1、學習平台首頁

學習平台首頁有架設一個翻譯機，如下圖三所示。使用者可在下方空欄使用翻譯機功能，輸入想要翻譯的古字詞後按下翻譯按鈕即可以達成文言文翻譯之功能並將結果呈現於下方空格。這翻譯機的設計使得古文翻譯變得迅速且方便，並且將翻譯結果清晰地呈現在下方的空白區域中，供使用者輕鬆查閱。



圖三：學習平台首頁展示

2、精選古文

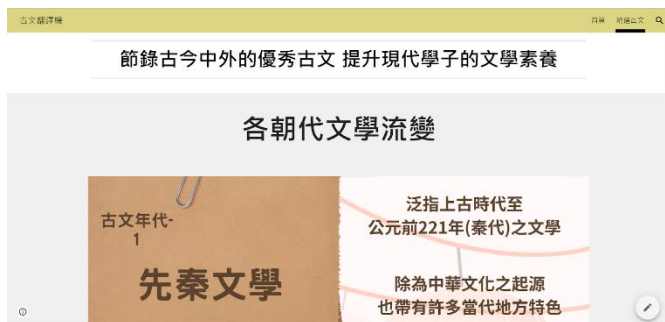
進入精選古文頁面之後點擊各朝代按鈕即可閱覽不同時期之精選古文，點開各篇選文之下拉頁即可閱覽每篇選文之原文、譯文以及解析選擇「返回歷史長河」即可回到精選古文頁面，利用 google site 內建的功能使整體網頁更精美，以此渲染出文藝氣息之風。



圖四：各朝代選文展示

3、網站內額外增加除翻譯外的閱覽功能

進入精選古文頁面後可點選代表各朝代的介紹圖片，在接受文學的薰陶之前先快速認識不同朝代的文學特色及代表作品。



圖五、google site 網站展示

(五) 網頁內部程式碼分析

1、輸入端程式碼編輯

我們採用了 javascript 程式碼內建取得網頁中的 localStorage 函式，使程式得以在客戶端取得字串型別的輸入原文。在調整文字及設定參數之後即可發送至我們設定好的翻譯 api 服務。

```
function Translate(e) {
  // localStorage只有本地才能存取！沒綁定是可以用
  let stringIn = document.querySelector("#input").value;
  let stringInJSON = JSON.stringify(stringIn);
  let output = "";
  if(stringIn){
    if(stringIn.length <= 500){
      localStorage.setItem("paragraph", stringIn);
      localStorage.setItem("paragraphJSON", stringInJSON);
      let stringOut = localStorage.getItem("paragraph");
      let stringOutJSON = localStorage.getItem("paragraphJSON");
      let stringOutJSONArr = JSON.parse(stringOutJSON);

      //encrypting
      //let stringOutNoSpace = adjust(stringOutJSONArr);
      let stringOutUTF8 = UTF8(stringOutJSONArr);
      let stringOutURLEncode = encodeURIComponent(stringOutUTF8);

      //百度api
      const url = "https://fanyi-api.baidu.com/api/trans/vip/translate";
      let qn = stringOutURLEncode;
      let q = stringOutUTF8;
      const from = "en";
      const to = "zh";
      const appid = " ";
      const key = " ";
      let salt = (new Date).getTime();
      let adSready = appid + q + salt + key;
      let sign = MD5(adSready);
    }
  }
}
```

圖六：獲取使用者輸入資料及整理之程式

2、生成翻譯結果與輸出端之程式碼

在使用 jQuery 中的 ajax 函數發送請求後，我們設定了幾種情況來取得翻譯後的值，取得成功即修改部分文字格式後上傳至網頁上完成翻譯；取得失敗即發送網頁彈出式視窗提醒用戶翻譯失敗也對超過五百字的文章和空白文字做出相對應的警示。

```
$.ajax({
  url: url,
  type: 'get',
  dataType: 'jsonp',
  data: {
    q: qn,
    appid: appid,
    salt: salt,
    from: from,
    to: to,
    sign: sign
  },
  success: function(data) {
    console.log("nice bro, get succeeded");
    console.log(data);
    let temp = data.trans_result[0].dst;
    temp = temp.replace(/\\s/g, "");
    temp = temp.split("/");
    output = adjust(string(temp));
    document.getElementById("output").innerHTML = output;
    // document.getElementById("output").innerHTML = output;
  },
  complete: function(jqXHR, textStatus){
  },
  error: function(jqXHR, textStatus, errorThrown){
    console.log("uhhh bro, get failed:" +errorThrown);
    alert("翻譯失敗，請重新輸入");
    document.getElementById("output").innerHTML = "翻譯失敗，請重新輸入";
  }
});
} else {
  alert("過字數上限！(五百字為限)");
}
} else {
  alert("輸入都非得為空！");
}
```

圖七：根據使用者輸入進行翻譯並輸出結果之程式碼編寫

(六) 針對系統弱點進行 SWOT 分析

表一：翻譯系統之 SWOT 分析

STRENGTHS 優勢	WEAKNESS 劣勢
1.翻譯精準且呼應作者本身 2.函式目的及功能明確 3.學生使用成效高	1.有過多字數的限制 2.只能精確對應 15 篇課綱內的古文 3.有些缺乏歷史的證據以證明

OPPORTUNITIES 機會	THREATS 威脅
1. 網站發展的可能性高 2. 適合教育使用 3. 能推廣 SDGs 第五點優質教育	1. 依然有不確定性存在 2. 使用者使用率不高 3. 成品完成前無法使用

五、結論與生活應用

(一) 結論

- 1、這項古文翻譯機為我們提供了更精確、迅速的方式來理解古代文獻，節省了大量的時間與精力。是學生的一大福音，在短時間內能達到理解並內化為本身知識。而這無疑是一個寶貴的工具，豐富了我們在國文學習領域的學習體驗。
- 2、在處理資料的方面，雖然這系統的益處眾多，卻也遇到許多中間障礙與困難，像是在使用 javascript 程式碼中，因為打了一長串程式碼，卻在執行中顯示錯誤，讓我們非常難在一片程式海中精確地抓出錯誤點且修正。

(二) 生活應用

- 1、當學生在學習文言文遇到困難時，可利用此網站進行翻譯，幫助學生了解文言文，透過此翻譯平台，也使學生熟知文章背景，有助於透徹剖析文意，深化學習文言文，提升學習效率，增進文學素養。
- 2、本系統若更精進，必然能與教育體系結合，形成科技與教育的結晶，在大科技革新中保有文學的美，利用科技晚留住歷史的流動。

參考資料

1. Ken Chung (2018年12月11日)。寫給程式麻瓜的 JavaScript 基礎介紹，跟工程師溝通零障礙。 <https://www.awoo.ai/zh-hant/blog/javascript-introduction/>
2. 百度翻譯開放平台 (2024)。通用翻譯 API 接入文檔。 <https://fanyi-api.baidu.com/product/113>