



作品名稱：篆來篆去

組別：踏實組

參賽學生：國立臺灣大學 材料科學與工程學系 許湛然

國立臺灣大學 中國文學系 蔡采吟

國立臺灣大學 工商管理學系 張鈺琳

指導老師：國立臺灣大學 語言學研究所 謝舒凱教授

網站連結：<https://splend1d.github.io/Zhuan/index.html>

目錄

- 一、創作動機與目的 p.3
- 二、作品介紹 p.4
- 三、團隊成員之跨領域合作情形與分工說明 p.4
- 四、作品使用手冊 p.5 - p.9
- 五、參考資料 p.10
- 六、其他（實作細節） p.10 - p.12

一、創作動機與目的：

小篆是由春秋戰國時代的秦國文字演變而來，字形規整勻稱。秦始皇統一六國後，李斯等人整理、統一撰《倉頡》篇，進行「書同文」工作，以當時的秦國小篆作為通用文字，消滅了「文字異形」現象，在漢字發展史上有重要意義。中國文字發展到小篆階段，逐漸定型，文字更加符號化。漢代時，隸書取代小篆成為主要字體，之後小篆主要用來刻印章、銘金石。《說文》是東漢許慎所編的文字工具書，全書分為 540 個部首、收錄 9353 字，收錄字體以小篆為主，也收錄古文、籀文，說解著重於本形本義，在文字學、訓詁學中有重要價值。

小篆是研究漢字演變的重要材料，識篆能直接閱讀古文物，獲得第一手資料或看懂新出土的文字資料；篆書線條單純、用筆簡單，用筆以中鋒為主，書法初學者能藉由習寫篆書鍛鍊對空間、線條質感的控制能力，並學習文字知識，因此適合書法初學者。對於設計領域的學生而言，小篆字形規整勻稱，不少店家招牌、商標使用小篆書寫，印章刻印也使用小篆，學習小篆能增加設計作品的人文思考。對普羅大眾而言，識篆、用篆能感受中國文字之美，增進人文素養。

我們觀察現有資源發現，對於中文系學生、考古學、器物學、文字學研究者、習書法者、古文物愛好者及其他欲學習識篆者，小篆的學習資源並不足。我們藉由訪談中文系學生，發現其學習小篆多是使用紙本書籍，藉由摹寫增進記憶，不僅造成查詢困難（欲找尋某一字的篆體，需翻閱大量紙本資料），抄寫背誦小篆部件的學習歷程也枯燥無效率。科技部人文及社科研究發展司人文處中文學門所撰之〈中文學門小學類文字學領域的困境與省思〉（中文學門小學專刊 11.3 期，2010 年 6 月）也提到「學生抱怨文字學內容乏味，恐怕導因於教師無法廣徵博引，將文字學內容教得活潑有趣。」

因此我們推出「篆來篆去」小篆學習器，希望能改善小篆學習資源不足、學習方式枯燥無效率的現象。我們藉由三種遊戲方式幫助使用者記憶小篆各部件，同時引入許慎《說文解字》之內容，帶領使用者了解文字背後的意義及形成原因，理解與記憶雙管齊下，將達到更佳學習效果。另一方面，我們整合現有的資料庫查詢篆體功能，讓使用者能在線將現代文字轉成小篆。期待「篆來篆去」能成為小篆學習新平台，幫助使用者更有效率、系統地學習小篆、理解文字之美！

二、作品介紹：

本作品引用中研院小學堂小篆資料庫為資料來源，不僅提供基本的楷體對應之小篆字體搜尋，亦結合每字部件做出延伸，將含有相似部件之字體列出，讓學習者可一次學習多個相關單字。除此互動式的查詢功能之外，我們亦設計出不同的遊戲學習模式和提示型態，讓學習者能用更有趣的方式認識小篆，使學習不再局限於背誦。

首先我們透過 Python 爬蟲取得中研院小學堂小篆資料庫之楷篆各項重要資料，對應與《說文》之解釋，並將每一個字的部件解析，以構成本作品主要的資料。接著，我們透過互動性網頁設計（HTML、CSS 與 JavaScript）將資料轉變成多種好玩的遊戲，不只為使用者帶來更好的學習效果，我們更希望透過本作品讓人文資料遊戲化，拉近大家與人文資料的距離，展現多元結合的可能性。

三、團隊成員之跨領域合作情形與分工說明

許湛然：Python 爬蟲、遊戲框架設計、網站介面優化

蔡采吟：視覺封面設計、網站介面設計、網站介面優化

張鈺琳：網站架構設計、遊戲介面設計、網站介面優化

四、作品使用手冊

本作品提供單字查詢功能，以及部件學習與單字學習兩大學習模式，在單字學習模式下又有三種學習型態，以下將詳細介紹本作品提供之服務：

● 單字查詢

可在網頁最上方輸入現代楷體字，除可查詢對應之小篆字體以外，還列出其部件組成，以及運用到類似部件的字體，若將滑鼠游標移至小篆字體上會呈現楷體，點擊字體也可以進到這個字的頁面，便於使用者對照著延伸學習，不須再一一輸入查詢。以「篆」字查詢，畫面展示如下圖：



● 部件學習

共分為三種難度，每題題目皆為五個小篆字，由簡單、中等到困難依序為部首、日常及宇宙關卡。部首關卡為找出題目中五個小篆字共同的部首，日常關卡與宇宙關卡為找出題目中共同的部件，唯宇宙關卡之答案所能衍生之單字較少，使用者可發揮觀察力來學習小篆字體的構成。下方依序為部首及日常關卡顯示答案後的畫面：



● 單字學習

由三個小遊戲組成，每個遊戲依照漢字的部件複雜度分為 20 關，每關 100 字。可以先在速成開天眼模式下學習單字，之後再進入直接試身手以及古代練功坊檢視學習成效。在測驗時，如果遇到不會的題目，在獲取提示或公布答案之後，每一張文字的圖片皆可以連結到單字學習的頁面，及時強化不會的單字。

○ 速成開天眼

直接列出楷體與小篆的對應並依常用度分級，同時呈現組成部件與類似字體，可作為學習小篆的快速攻略。網頁畫面展示如下：



○ 直接試身手

每題為一個小篆字體，輸入該字現代寫法，答對後會自動進入下一關。若需提示，可點擊「碎字機」按鈕，將列出該字的所有部件，使用者可運用歸納與整合的能力推敲出該字為何，同時在過程中加強各部件的學習。下圖為點選「碎字機」提示鈕後的畫面：



○ 古代練功坊

為所有遊戲中難度最高者，每題為以小篆呈現的該字《說文》釋義，並將答案以底線與粉色背景作為遮罩，閱讀後輸入遮罩處應為何字，在此學習模式下可以體驗古人用小篆閱讀文章。下圖為題目畫面：



若需提示，可點擊「現代眼藥水」按鈕，題目便會轉為以楷體呈現。下圖為點選「現代眼藥水」提示鈕後的畫面，此時移到字體上方，會暫時變換回小篆字體，可即時對照：



五、參考資料

- 資料庫來源：
 - 中研院小學堂小篆資料庫：<http://xiaoxue.iis.sinica.edu.tw/>（查詢介面所得之各解析度「字形圖片」及字形屬性資訊為無著作權之 CC0 1.0）
- 網站設計來源：
 - 背景圖片、封面相片：PNG Tree <https://zh.pngtree.com/>
 - 版面配置：Templated <https://templated.co/>（著作權規範為 CC By 3.0）
- 作品詳介參考資料：
 - 裘錫圭《文字學概要》
 - 科技部人文及社科研究發展司人文處中文學門〈中文學門小學類文字學領域的困境與省思〉（中文學門小學專刊 11.3 期，2010 年 6 月）
- 遊戲題目與關卡設計參考資料：
 - 林樹〈中文電腦基本用字表〉

六、其他（實作細節）

GitHub：<https://github.com/Splend1d/Zhuan>

- A. 路徑 `./assets/src/get*.py` 之檔案：為爬蟲程式，將中研院資料庫轉為 pickle 檔案存取
- 1) `getbasic.py`：爬取核心資料。蒐集的資訊包含所有部首的集合，以及每一個小篆字體的核心資訊，下面顯示此爬蟲的其中一筆結果
 - 1: {'meaning': '《說文》：“一，惟初太始，道立於一，造分天地，化成萬物。弌，古文一。”', 'fonts': ['&0.4E00;', '&1.EFB2;', '&27.4E00;', '&27.E000;']}, 2: {'meaning': '《說文》：“元，始也。从一，从兀。”', 'fonts': ['&0.5143;', '&27.5143;']}
 - 2) `getfreq.py`：爬取小篆字體中，對應成現代漢字的常用字率的資料。此處常用字字集乃依林樹教授編著之〈中文電腦基本用字表〉所記錄之字頻排序，由高字頻排至低字頻字形。此外，門檻值設為 100000 時，可獲得所有常用漢字集，是在**部件學習**模式中所使用。在部件學習模式中，為了避免在隨機生成的題目中，答案只有不常使用的古字，題幹會事先經過篩選，答案內包含夠多的常用字時，才會被納入題目當中。

- 3) `getimg.py` (使用前需先執行 `getbasic.py`)：從 `getbasic.py` 所獲得的核心資料中，蒐集每一個字元的相片，並建立小篆字型與楷書字型的對照表，對照表的一筆結果如下，分別對應到小篆字形，楷書字形，以及電腦漢字

['27.932E', '0.932E', '𪛗']

- 4) `gettree.py` (使用前需先執行 `getbasic.py`)：從 `getbasic.py` 所獲得的核心資料中，蒐集每一個字元的衍生字以及子部件，做法為透過漢字古今資料庫的部件功能，先找出每一個字元的衍生字，就可以反向推衍出子部件。

B. 路徑 `./assets/src/parse*.py` 之檔案：整理先前所爬取的資料，並將其從 `pickle` 檔轉為 `json` 檔

- 1) `parsemain.py`：在主要的資料裡加入部首的資料。
- 2) `parsemajors.py`：由於中研院資料庫頁面並沒有部首的資訊，需要由部首查詢來查詢正確的部首，有些字礙於 `utf-8` 的限制並無法直接用爬蟲獲得，於是人工查找的方式補齊每個字的部首資料。
- 3) `parsesubchars.py`：中研院資料庫裡有些字元屬於籀文的範疇，因此我們把不將異體字納入資料庫內。此程式可以分出主要的字集異體字集。
- 4) `parsetree.py`：整理字的樹狀結構。針對每一個字的子部件以及衍生字進行排序，使得在顯示子部件時，會從部件數目最多的子部件，條列到部件數目最少的子部件，如下圖之結果。衍生字則正好相反，會由子部件數目少的排到子部件數目大的。



- 5) `parsefreq.py`：對常見字集進行排序。為了使學習成效更能提升，在關卡處理上，我們希望字在部件上是能夠有連貫性的，例如：「漢」這個字的部件有「水」，則在學習漢這個字之前，應該先學習水。因此，我們採用的演算法如下：

[關卡分類演算法]

1. 找尋子部件只有自己的小篆單字，令此集合為 x 。
2. 從我們建立的子部件資料庫中，找出子部件含有 x 的單字，並將其子部件 x 暫時去除。
3. 去除後，有些單字的子部件減少，可能有符合步驟 1 之新的單字產生，重新做步驟 1。

此方法理論上可以將所有字元分類，在做完此演算法之後，我們發現有一些字詞仍然無法被分類，原因是有一些部件為彼此互為子部件，導致子部件無法被消除的情況，於是，我們增加了一條規則，來解決這種字元（2000 筆中大概有 100 筆）。

4. 若有字尚未被分類完成，而已經找不到新的基礎單字時，從剩下的字元中，選擇衍生字最多的字元，加到難度序列的下一個值。接著重複步驟 1~3。

- C. 路徑 `./assets/src/gen*.py` 之檔案：生成 `.js` 檔作為 `javascript` 的資料庫
- D. 路徑 `./assets/db/` 之檔案：我們的資料庫
- E. 路徑 `./*.html`：網站內容