**Introduction**

This document contains instructions on how to operate the contents of this folder as well as an example walkthrough.

**Instructions**

This folder contains two programs. The first, **Parser.jar**, is an executable file. Double click it to run. This program reads in an input file containing vocabulary words, as well as possibly some extra text, and returns an output file with alternating lines of traditional and simplified characters. If a phrase is the same in traditional and simplified, both lines will be the same.

The operation of **Parser.jar** is controlled by the text file **ParserParameters.txt**. This contains the following lines:

* Source file: This line is the location of the text file to read in. It must be located in the same folder as **Parser.jar**.
* Automatically detect encoding: This line is default set to yes, which is recommended. This will allow **Parser.jar** to deal with encode method and spacing on its own, as explained below.
* Encode method: This line will only be used if Automatically detect encoding is set to no. This line accepts a number from 1-3 inclusive. Each encoding indicates an acceptable format for the source file.
  + Encoding 1: The vocab words must be in the format 禮讓(礼让). Spaces and new lines between each phrase set will be ignored.
  + Encoding 2: The vocab words must be in the format 禮(礼)讓(让). Spaces between each phrase set will be ignored; new lines will not.
  + Encoding 3: The vocab words must be in the format 禮讓 礼让, with spaces separating traditional and simplified. New phrases must be located on separate lines.
  + For each of the given phrases above, the output will be 禮讓 礼让 on separate lines.
* Ignore spaces: This line will only be used if Automatically detect encoding is set to no. This line determines whether spaces are skipped as the file is being processed. It is default set to yes; set it to no if vocabulary words are space separated rather than new-line separated.
* Write file: This line indicates the address of the output file. It must be located in the same folder as **Parser.jar**. If the same write file is used twice, the second run will overwrite any data stored in the first. Therefore it is recommended to put all vocabulary words for a grade in one source file before running **Parser.jar**. This file does not need to be manually created; if it does not exist already, **Parser.jar** will create it for you.

After running **Parser.jar**, the file specified by Write file will contain the output. This file can be manually edited to eliminate any mistakes.

The second program is a PowerPoint macro that automatically creates the presentation from the vocabulary contained in the output file. The code is contained in the text file **PowerpointCreator.txt**

Note that if you want to skip the steps below for creating and running a macro, you may use **Presentation1.pptm**, which should already have the macro loaded, along with the appropriate libraries. However you will still need parts of the following instructions to access and run it. If you are running PowerPoint 2003, the code in **Presentation1.pptm** will not apply. Instead, you will need to follow the instructions below, and instead of pasting the code from **PowerpointCreator.txt**, use the code in **PowerpointCreator(2003).txt**.

To run this code, create a PowerPoint presentation and save as a PowerPoint Macro-Enabled Presentation. Next, enable the **Developer** tab. On the **File** tab, choose **Options** to open the **PowerPoint Options** dialog box. Click **Customize Ribbon** on the left side of the dialog box. Under **Choose commands from** on the left side of the dialog box, select **Popular Commands**. Under **Customize the ribbon** on the right side of the dialog box, select **Main tabs**, and then select the **Developer** check box, and click **OK**.

Now, go to the **Developer** tab and click on **Macro Security**. Select **Enable all macros**. Now, to create the macro, click the **Macros** button on the **Developer** tab. In the dialog box, type **PowerpointCreator** under **Macro Name**. Click the **Create** button to open the editor.

Now, delete any code present in the editor and paste in the code from **PowerpointCreator.txt** (or **PowerpointCreator(2003).txt**). Go to the **Tools** tab and select **References**. Scroll down and select **Microsoft ActiveX Data Objects 2.5 Library**, and click **OK**. This is a built-in library necessary for reading in Chinese characters.

The behavior of this macro is controlled by the lines following **‘ PARAMETERS**, which should be in green. Do not touch the rest of the code. The parameters are:

* FileName: This is the location of the file to read in vocabulary from, i.e. the one created by **Parser.jar**. Do not include any directories; the file name is sufficient. Do not forget to include double quotes.
* ImageName: The image file containing the small home logo to display on each slide. It will hyperlink back to the table of contents slide. Do not include any directories; the file name is sufficient. Do not forget to include double quotes.
* DefaultTradSize: This is the font size for traditional words, if they are shorter than three characters. Longer words will automatically scale.
* DefaultSimpSize: This is the font size for simplified words, if they are shorter than three characters. Longer words will automatically scale.
* DefaultPadding: This is the space maintained between traditional and simplified text boxes.
* Fontname: The font to display characters in. Do not forget to include double quotes.
* TradBold: Defaults to true. Controls if traditional characters are bolded.
* SimpBold: Defaults to true. Controls if simplified characters are bolded.
* TradRed, TradBlue, TradGreen: Controls the colour of traditional characters, as an RGB colour. Defaults to black.
* SimpRed, SimpBlue, SimpGreen: Controls the colour of simplified characters, as an RGB colour. Defaults to blue.
* ColumnNum: The number of columns to include in the table of contents on the first slide. Defaults to 20. The number of rows will be automatically adjusted.

The macro can now be run. Click the small green arrow to run it. After the macro is run, it will create a presentation to which manual changes can be made. A slide title will need to be added, and the size and formatting of the table of contents will need to be adjusted. Everything else should be properly formatted.

Due to the time-consuming process of setting up and running macros, it is recommended that for all grades, only one Macro-Enabled Presentation is created. After creating the presentation for one grade, save the file as a regular PowerPoint Presentation. Then, delete all contents of the Macro-Enabled Presentation (except for a blank first slide), and run the macro again on the next grade’s vocabulary list, restarting the process.

**Examples**

Let’s say we are sourcing our vocabulary list from <https://quizlet.com/162147272/lesson-one-1-i-flash-cards/>. Create a source file, such as **Source.txt**, and paste the vocabulary contents from the webpage. This might create the following in **Source.txt**:

香噴噴（香喷喷）

(xiāng pēn pēn) delicious

熱騰騰（热腾腾）

(rè téng téng) hot

包子

(bāo zi) steamed stuffed bun

嘍（喽）

(lóu) particle expressing chagrin

呦

(yōu) deer bleating

靈（灵）

(líng) skilled

吃相

(chī xiàng) eating manners

規矩（规矩）

(guī jǔ) rule

塞

(sāi) stuff

亂講（乱讲）

(luàn jiǎng) say what should not be said

告狀

(gào zhuàng) complain to

互相

(hù xiāng) mutua

禮讓（礼让）

(lǐ ràng) politeness or mutual courtesy

白吃

(bái chī) eat for free

魔術（魔术）

(mó shù) magic

剩下

(shèng xià) to remain

困難（困难）

(kùn nán) difficult

Note that there will be non-Chinese definitions and words. This will be filtered out by **Parser.jar**. Now, modify **ParserParameters.txt**:

* Source file: Sample.txt, for example.
* Write file: Output.txt, for example; note that if there is any existing content in this file, it will be deleted.

This will result in the following in **Output.txt**:

香噴噴

香喷喷

熱騰騰

热腾腾

包子

包子

嘍

喽

呦

呦

靈

灵

吃相

吃相

規矩

规矩

塞

塞

亂講

乱讲

告狀

告狀

互相

互相

禮讓

礼让

白吃

白吃

魔術

魔术

剩下

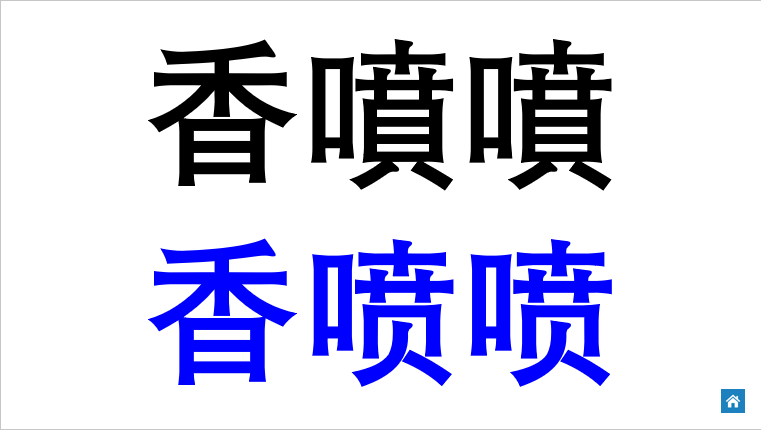
剩下

困難

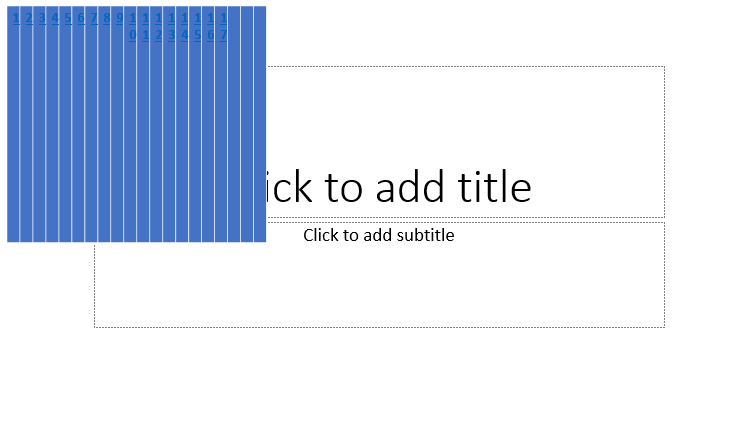
困难

Now, follow the steps to run the macro, and under parameters, set FileName = "Output.txt". It will result in a presentation that looks like this:

Vocabulary slide example:



Title slide example:



Note that the table of contents requires manual resizing and formatting. However, all the hyperlinks will be added automatically.

All code is available on <https://github.com/lpang36/ChineseVocabCreator>.