JCloze Drag the Words by   2014-2015
     Stan Bogdanov
     Daphnie
     Glenys Hanson
     Agnès Simonet
Revisited by Agnès Simonet for Hot Potatoes v7   November 2020
Documentation: June 2021

JCloze Drag the Words

JCdw v 2.0

(for Hot Potatoes v 7.0)

In memory of Glenys Hanson

Preface

This add-on has been asked to Stan Bogdanov and myself by Glenys Hanson and Daphnie in 2014.

We worked hard during some months:

- *I was writing the javascript code.*

- *Stan was writing the css for a nice display, reporting issues in the code, creating the RTL version and testing.*

- *Glenys and Daphnie were imagining new features and testing, again and again.*

We built an operational version with only a little erratic issue with the drag on touch devices, due to the external library jQuery.

And we still had to write the tutorial which is always a time consuming work.

Then life, with its train of misfortunes, put a stop to our project which has never been released.

Having improved my knowledges about HTML5 drag & drop, I decided, in winter 2018, to modify the code, getting rid of jQuery. But it was the time when Hot Potatoes v6 was becoming v7, so I waited until now to adapt it to the new version of Hot Potatoes. It led to this version 2.0.

This version 2.0 could not have existed without the work of our team in 2014-2015.

Agnès Simonet

Table of contents

The examples in this tutorial are for version 2.0.1.

You can easily adapt them for other versions (2.0.2 etc.)

ATTENTION

Quotes in the provided codes in this tutorial must be **regular quotes**.

Regular (straight) quotes are frequently converted to smart (curly) quotes in word processors and PDFs .

Regular quote: ""...

Smart quote: "..."

To avoid this, directly type code or copy-paste it in a plain text editor (such as Notepad, Gedit…) and then copy-paste it from the plain text editor into Hot Potatoes.

I What is a JCloze Drag the Words exercise?

Words (or phrases) are added to the text using the mouse or the finger (on touch devices) to drag words (or phrases) and drop them in the gaps.

Like in the original JCloze, clues and alternative correct answers can be added.

A bonus of JCloze Drag the Words is the possibility to have distractors in the list of draggable words.

List of words — Word being dragged — Distractor — Word having been dropped — Word

Drag answers in the gaps.

Additive Color System

blue blue cyan green magenta orange red

magenta

Media that combine emitted lights to create the sensation of a range of colors are using the additive color system.

Typically, the primary colors used are : _____ (?), _____ and _____ .

Green and _____ (?) are complementary colors.

Yellow and green (?) are complementary colors.

Check   Solution

Dropped word — Clue — Gap

After checking:

Word having been incorrectly dropped and still droppable

Drag answers in the gaps.

Additive Color System

blue blue cyan green magenta orange

Media that combine emitted lights to create the sensation of a range of colors are using the additive color system.

Typically, the primary colors used are : _____ (?), _____ and red .

Green and _____ (?) are complementary colors.

Yellow and green (?) are complementary colors.

Check   Solution

Incorrect answer — Correct answer

Note: the words (or phrases) must be dragged. You should not type them directly in the gap.

II HTML5 specifications

The code has been rewritten using HTML5 specifications for drag&drop and audio elements.

1- Drag&drop

- All the PC browsers understand HTML5 for drag&drop.

- On touch devices, since iOS 11, drag&drop works on iPads and iPhones.

- Since 2019 it is also working on Android with Chrome v49.

- For other touch devices (using ios before iOS 11, or android with Chrome version <49 or other browser) you need "**DragDropTouch.js**" in your exercise(s) folder. You will find it in the "demos" folder.

2- Mini audio button

With Hot Potatoes v7, via "Insert> Insert a media object", you can insert a classic HTML5 audio player:
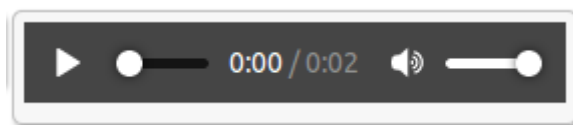
You can also use the HTML5 mini audio button created for my Hot Potatoes add-ons:

See audio_tutorial_4.pdf

<u>Note</u>: audio files (and images) can't be used in the list of draggable items.
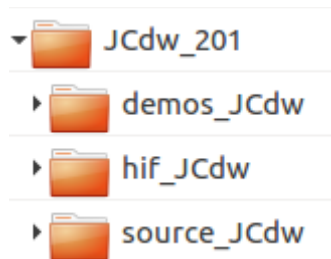
III How to install JCloze Drag the Words

Unzip JCdw_201.zip and **copy the JCdw_201 folder** where you want, for example in the source folder of Hot Potatoes:
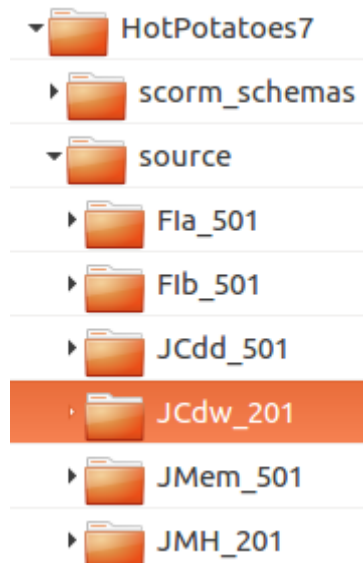
Note: you can copy the folder anywhere on your hard disk and rename it as you like.

Check that the demos, hif and source folders are present:

IV How to load the interface file

You don't necessarily have to do it but if you load the interface file associated with JCloze Drag the Words it will be easier for you to create your exercise and change the configuration output.

While JCloze is opened, click on **Options->Interface->Load interface file**.
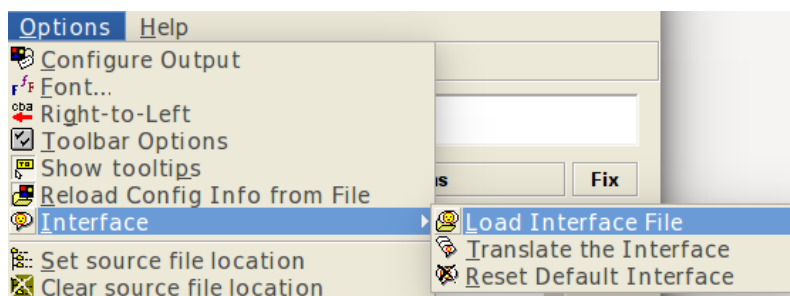

Browse to the folder JCdw_201/hif_JCdw and choose the **JC_dw_201_en.hif** .

In the JCdw_201/hif_JCdw folder you will only find the hif.in English You will find hif for other languages, if they exist, on the hpaddons site*.*

You can add the translation in your language, if it does not exist, on the Paolo Cutini's site *or contact me on the* hotpotusers google group*.*

A dialogue box will ask you if you would like to change the interface for all applications. Choose "**No**".
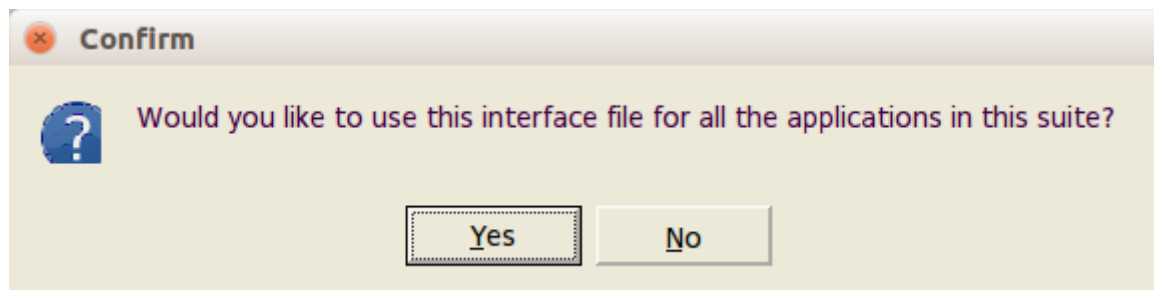
You will notice the changes in Options>Configuration output, for example:
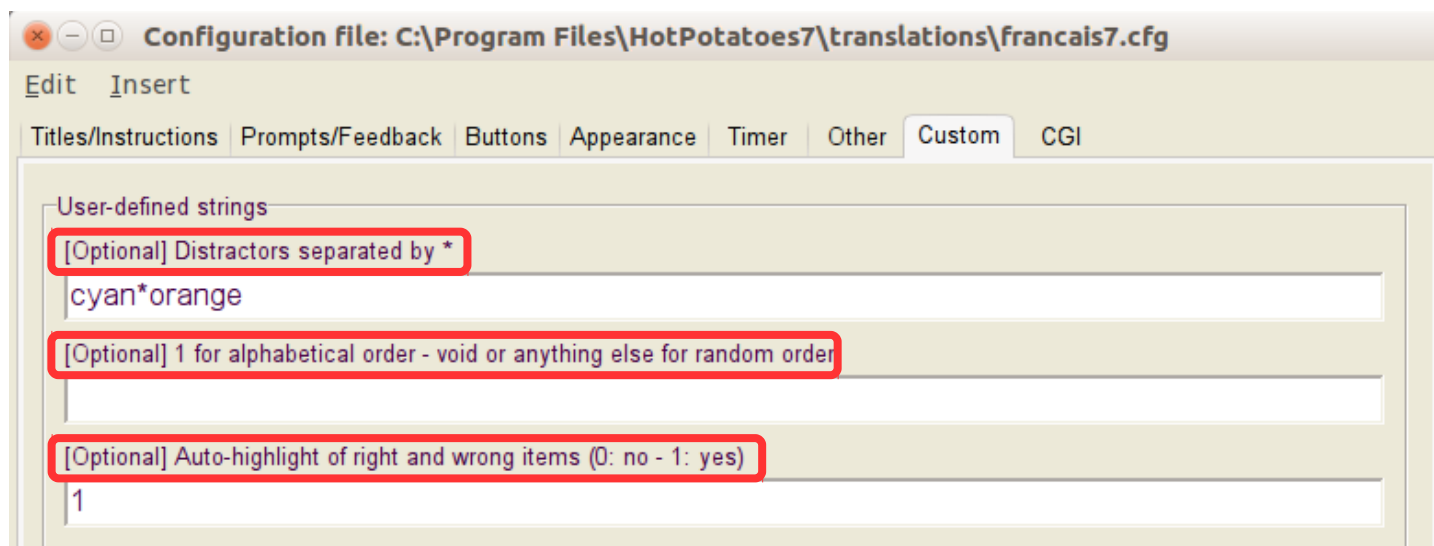
V How to create a JCloze Drag the Words exercise

1- **Start JCloze** and use it as you are used to with the original version.

For more information on how JCloze works, check out the Help file -- just press the F1 key (it will open hotpot.chm which is in The HotPotatoes folder).

- To add clues, see below (page 9)

- To configure the output see below (page 10)

- You have to **check "Include word list with text"**:
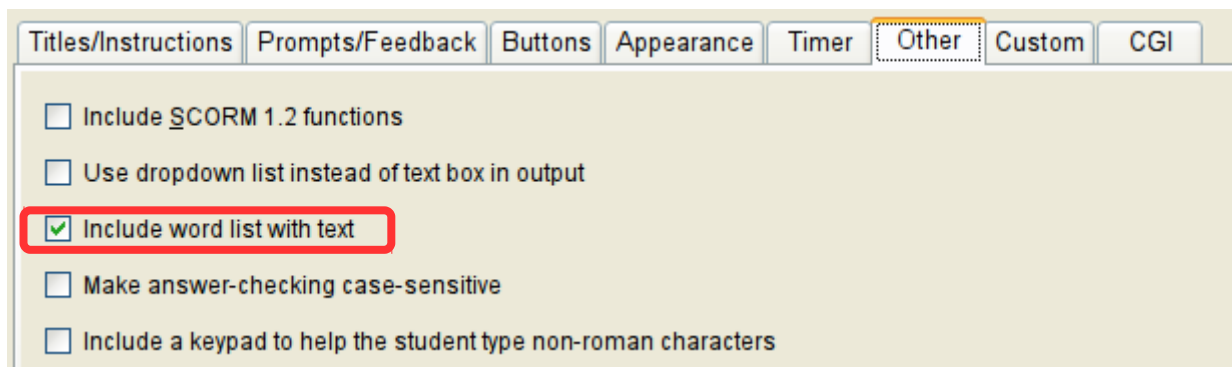
2- Now it's time to tell JCloze that instead of using the original source files, the special source files have to be used.

Press the **Shift + Control + ALT + S** keys **simultaneously** or the appropriate file menu item (Options>Set source file location 🗁) .

Choose the special source folder where the needed file(s) are located:
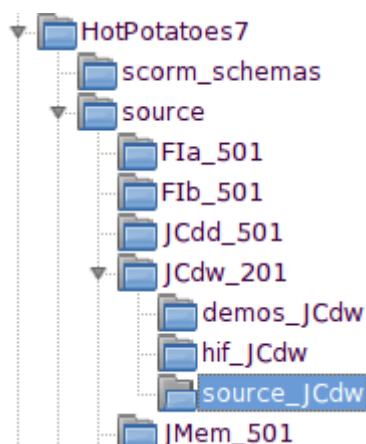
You will see which source folder is in use:
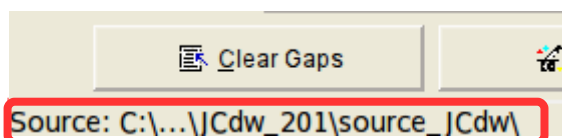
Be aware that the next time the JCloze program is loaded, it will use the source files within this special source files folder.

If you want to create a normal JCloze exercise, press the Shift + Control + ALT + T keys simultaneously.

f you want to create another type of JCloze exercise (using another add-on), press the Shift + Control + ALT + S keys simultaneously and choose the adequate source folder.

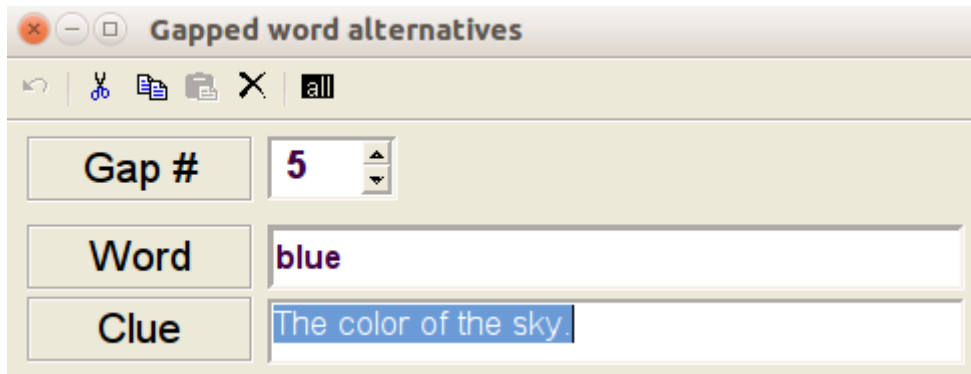3- Create your web-exercise using **F6** or the appropriate file menu item 🕸

VI How to add clues

The display of the clues uses an external script called dhtmltooltip. So you must add **"dhtmltooltips_JCdw.js" in your exercise folder**. You will find it in the demos_JCdw folder.

The examples below all refer to JCdw_demo1.jcl in the demos_JCdw folder.

1- Text clue
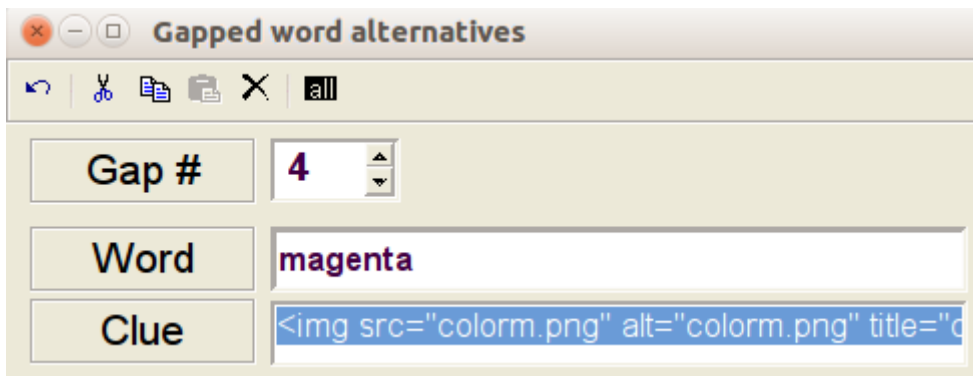
Type in the text:

2- Image clue

- Remember to **add the image** you wish to use **into your exercise folder**.
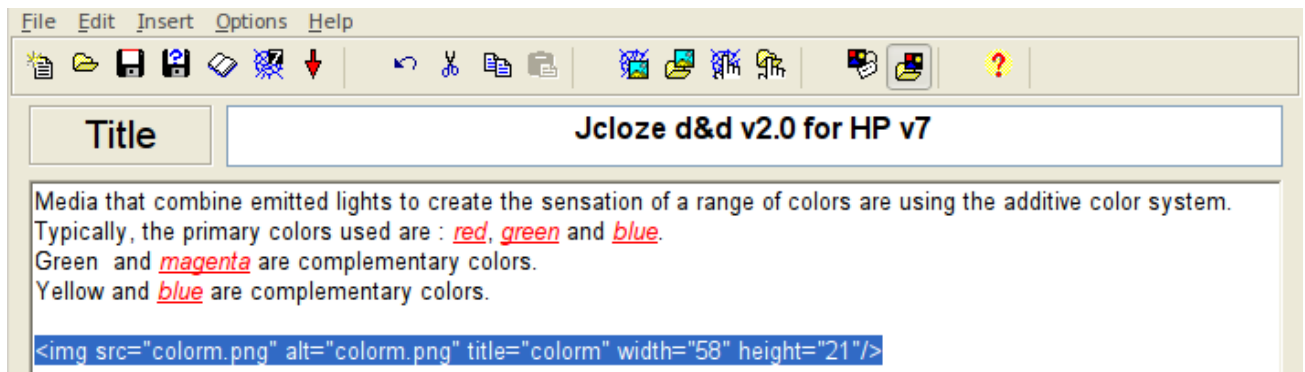- Insert the following code for the image into the "Clue" field in JCloze:

```
<img src="colorm.png" alt="colorm.png" title="colorm" width="58" height="21"/>
```
(Use the name of your file instead of colorm.png and the appropriate width and height.)

Tip: In the main text field, use the Insert > Picture from local file menu of JCloze to automatically get the html code and then copy the code into the clue text field.

Warning! After copy don't forget to delete the code in the text field of the exercise.

3- Audio clue

- Remember to **add the audio file** you wish to use **into your exercise folder**.

- Paste the following code into the "Clue" field in JCloze:

```
<audio controls="controls" autoplay><source src="blood_color.mp3"
type="audio/mp3"/></audio>
```
(Use the name of your file instead of blood_color.mp3)

Note: the code must be on a single line.

Note: use audio clues with short audio files as the user will not be able to rewind or stop while listening to the sound.

- For other formats change *type="audio/mp3"* to:
  - type="audio/ogg"   for ogg format
  - type="audio/wav"   for wav format

VII Configuration

In the interface, some options have been added or modified. The other ones are left unchanged.

If you have loaded the interface file (page 7) it will be easier to change the configuration output.

1- How to add distractors to the list of words

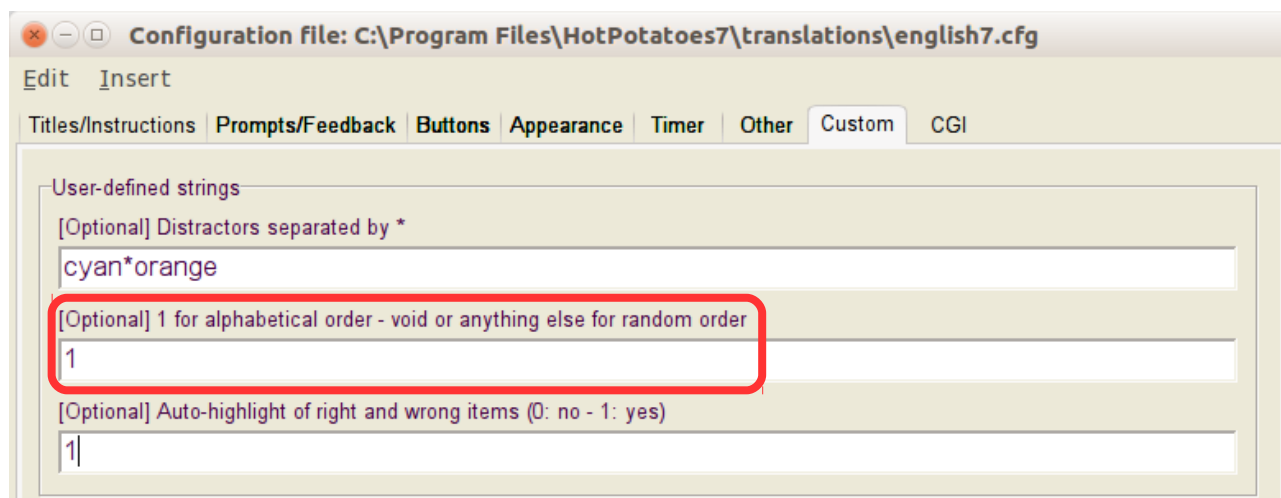**Options>Configuration output>Custom**:
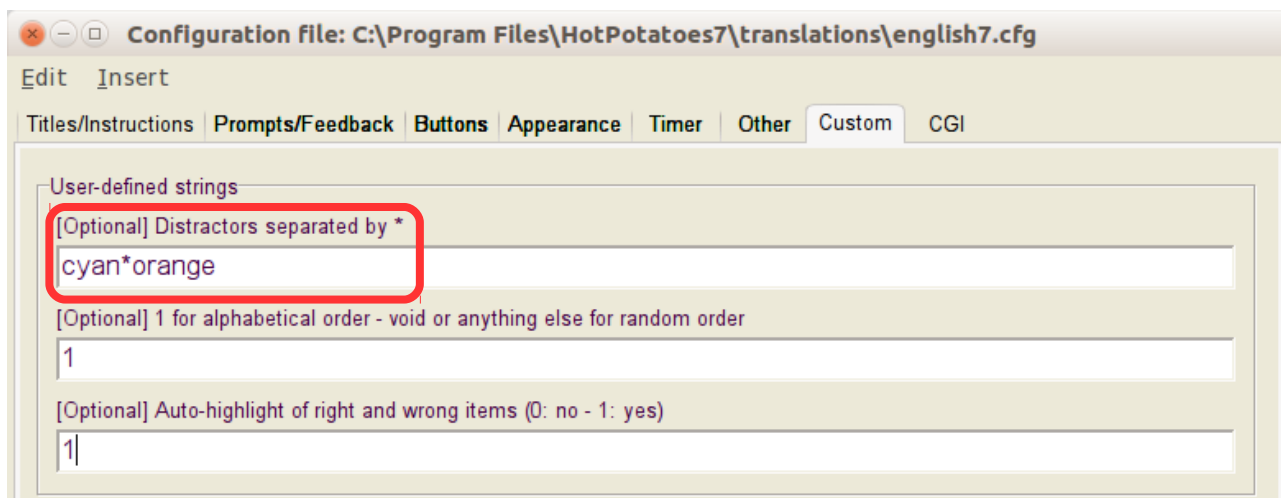
- You can add one or more distractors. If you add more than one you must separate them by an asterisk (*).

- If you keep the box empty, no distractors will be added to the list of words.

2- How to have the words in alphabetical order in the list of words

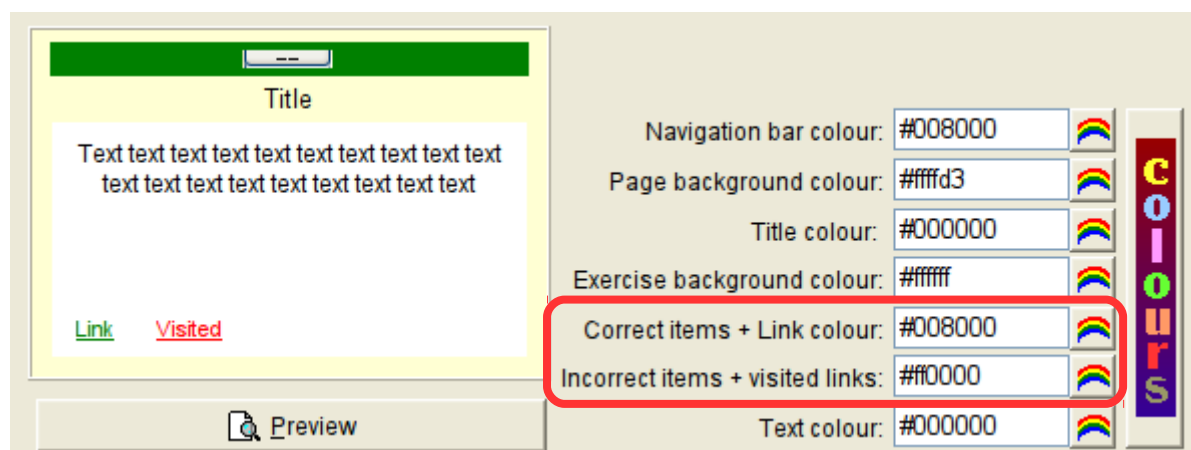Alphabetical order is useful if there is a lot of words in the list.

Options>Configuration output>Custom:

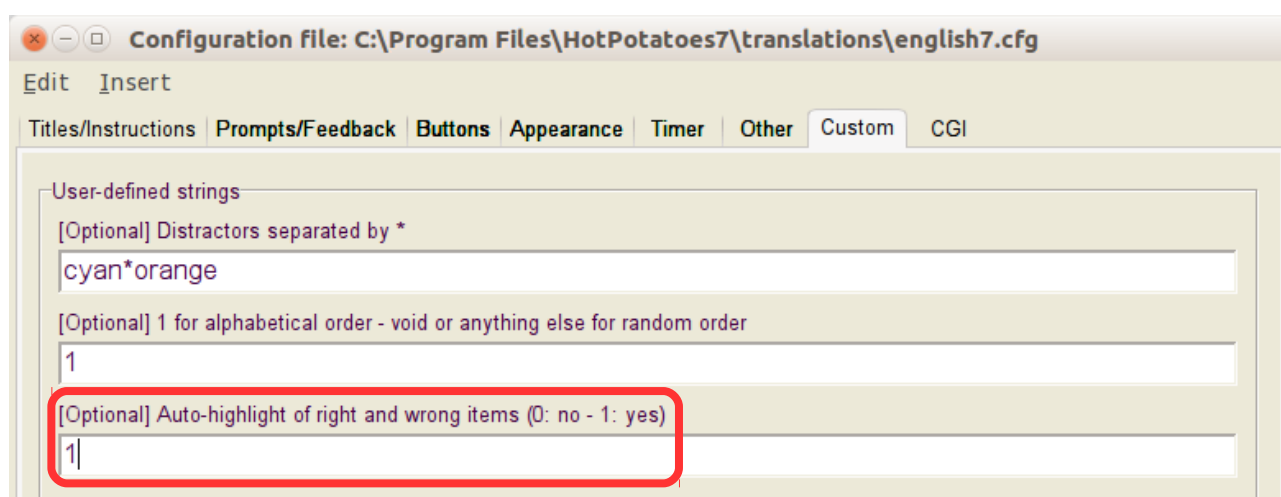## 3- How to choose the colors for wrongly and rightly positioned items

The colors are the same as the links and visited links colors.

Options>Configuration output>Appearance:

4- How to enable Practice mode with auto-highlight of right and wrong items

Options>Configuration output>Custom

- Type "**1**" to enable the **Practice mode**: in Practice mode the items are automatically highlighted as soon as they are dropped, using the defined colors (see 3- page 11)

blue magenta red cyan green orange

Media that combine emitted lights to create the sensation of a range of colors are using the additive color system.

Typically, the primary colors used are : _____ (?),   blue   and _____ .

Green and _____ (?) are complementary colors.

Yellow and   cyan   (?) are complementary colors.

- If **empty** or if you type "**0**" the default mode is **Test mode**. Answers will be highlighted after validation.

blue cyan green magenta orange red

Med

Your score is: 20%.
Some of your answers are incorrect.

nsation of a range of colors are using

the

OK

Typ (?), _____ and   blue   .

Green and _____ (?) are complementary colors.

Yellow and   cyan   (?) are complementary colors.

5- How to define the width of the reading text container

If you use a reading text you can easily choose the width of the reading text container.

Options>Configuration output>Custom:

```
Code for insertion into <head> tag

<style>
div.ReadingContainer{flex-grow: 20;}
div.ExerciseContainer{flex-grow: 80;}
</style>
```

The logic is:

- width of {ReadingContainer + ExerciseContainer}: 20 + 80 =100
- width of ReadingContainer: 20/100 (20%)
- width of ExerciseContainer: 80/100 (80%)

If you want the width of the reading text container to be 40% of the width of the page, use 40 for ReadingContainer and 60 (=100-40) for ExerciseContainer. So type:

```
<style>
div.ReadingContainer{flex-grow: 40;}
div.ExerciseContainer{flex-grow: 60;}
</style>
```



```
Code for insertion into <head> tag
<style>
div.ReadingContainer{flex-grow: 40;}
div.ExerciseContainer{flex-grow: 60;}
</style>
```
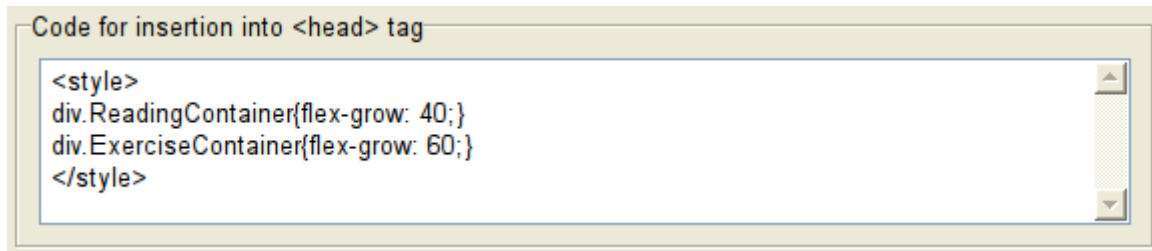
If empty, the widths will be 50% for ReadingContainer and 50% for ExerciseContainer.

VIII Other tips

1) Colors

They are defined in **hp7.cs_**, near line 84.

```
/* for JCloze drag the words */
   --strWordsColor: [strTextColor]; /* color of the text of words in the list
*/
   --strWordsBorderColor: silver; /* color of the border of words in the list
*/
   --strWordsBG: Gainsboro; /* background of words in the list */
   --strDroppedWord: blue;  /* for testmode */

   --strGapDroppedWord: blue; /* for test mode */
   --strGapBorderColor: silver; /* bordure inférieure des gaps */
   --strGapDrop: blue;    /*for practice mode and  test mode */
   --strGapDropCorrect: [strLinkColor];; /* for practice mode and check in
testmode */
   --strGapDropIncorrect: [strVLinkColor]; /* for practice mode */
   --strGapEnter: silver; /* for practice mode and  test mode */
   --strGapCheckIncorrect: [strVLinkColor];  /* for check in testmode */

   --strClueColor: green; /* color of the text of the clue marker (?) */
   --strClueBG: gold; /* color of the background of the clue marker (?) */
   --strClueBorderColor:#444; /* color of the border of the clue marker (?) */
```

You can change the values.

You can find the names or the HTML hexadecimal codes for colors on wikipedia.

2) Clues

You can:

a) **Modify the colors of the clue marker**

The colors are defined in --strClueColor, --strClueBG and --strClueBorderColor (see above).

You can change the values. For example, to get (**?**) for the clue marker:

```
--strClueColor: black; /* color of the text of the clue marker (?) */
--strClueBG: white; /* color of the background of the clue marker (?) */
--strClueBorderColor:red; /* color of the border of the clue marker (?) */
```

b) Modify the style of the tip

Search for dhtmltooltip **in hp7.cs_**, near line 1020.

```css
#dhtmltooltip{
position: absolute;
   max-width: 30%;
   border-left: 5px solid green;
   border-right: 1px solid #333;
   border-top: 1px solid  #333;
   border-bottom: 1px solid #333;
   padding:8px;
   background-color: gold;
   visibility: hidden;
   z-index: 100;
   border-radius: 5px;
   /* opacity:0.8;*/
   box-shadow: 4px 4px 5px #888;
}
```

For example, if you choose:

```css
border-left: 5px solid red;
background-color: white;
```

You will get:
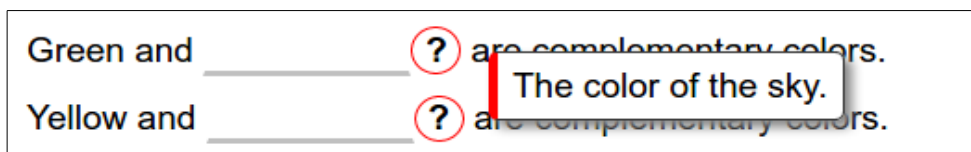
c) Use any text you want instead of the "?"

For example:
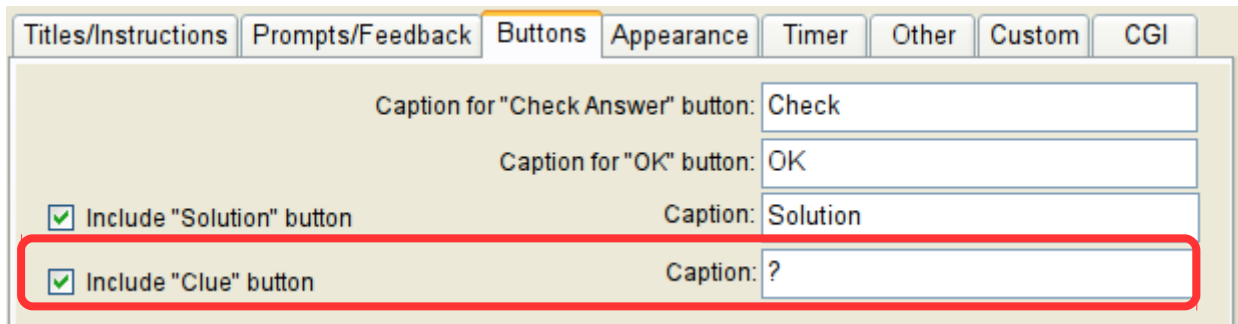
Configuration	Result
	Typically, the primary colors used are : _____ (i),
	Typically, the primary colors used are : _____ (Help),

You can notice that the shape of the clue marker has changed. To modify that, see below.

d) **Modify the shape of the clue marker**

The shape of the clue marker depends on:
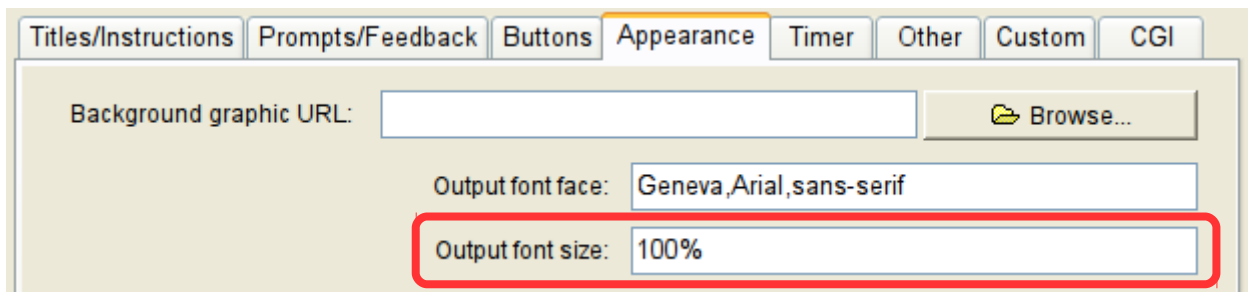
- the text (?, i etc.) ,

- the font size, defined in the configuration file:

- the font weight, the padding and the border radius, defined in **hp7.cs_** near line 1075:

```
.clue {
    cursor:help;
    font-weight:bold;
    color:var(--strClueColor);
    background-color:var(--strClueBG);
    padding: 0.1em 0.35em; /*  for ?  */
    /*padding: 0.1em 0.5em;  for i or Help */
    line-height:1;
    border:1px solid var(--strClueBorderColor);
    border-radius: 50%; /*  for ?  */
    /*border-radius: 5px;  for Help  */
}
```

You have to try different values.

Examples:

| settings	Result
font-weight:bold; **padding: 0.1em 0.5em;** border-radius: 50%;	i
font-weight:bold; **padding: 0.1em 0.5em;** **border-radius: 5px;**	Help

Note: with some touchscreen devices the perfect circle you get on your PC may not be so perfect.

Don't forget to save the modified file.

IX Virtual pagination

To use Virtual pagination like in JCdw_demo3.htm you need:

- to put **virtualpaginate.js** and **virtualpagination.css** in your exercise folder. You will find them in the demos_JCdw folder.

- to use **source_JCdw_VP**.
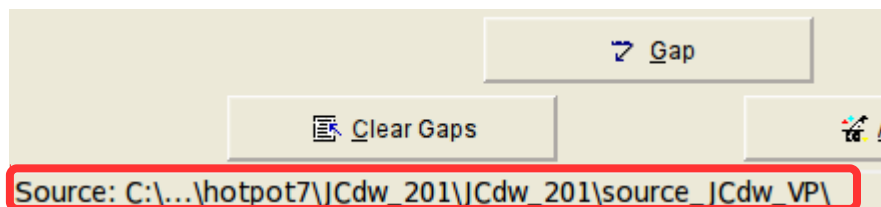
- to put each chunk of text between:

    * <div class="virtualpage"> and </div> for text in the exercise container.

    * <div class="virtualpage_**rt**"> and </div> for text in the reading container.

<u>Note</u>: **don't insert a carriage return** between the parts, otherwise the parts will be shifted down a line.

For example, don't write:

<div class="virtualpage">text of part1</div>

<div class="virtualpage">text of part2</div>

but:

<div class="virtualpage">text of part1</div><div class="virtualpage">text of part2</div>

X External files (summary)

Files that you need in your exercise folder:

• dhtmltooltips_JCdw.js (for the display of the clues)

• [Optional] DragDropTouch.js (for touch devices)

• [Optional] player4.js (for mini audio button)

• [Optional] virtualpaginate.js and virtualpagination.css (for Virtual pagination)

You will find them in the demos_JCdw folder.