

Clozemaker - Information

[Index](#) > [Teachers Animation Toolkit](#) > [Clozemaker](#) > [Info](#)

Download all the files in an archive [here](#) (.zip)

Download a pdf of this page [here](#)

Clozemaker generates a simple cloze activity with scoring and a reset button. The text and some of the variables that control how it looks are loaded from a text file ([clozetext.txt](#)). The aim is to simplify the production of this type of short revision activity.

Lets start with the content of [clozetext.txt](#). It isn't very easy to read because of the rules about how Flash loads variables from a text file, unfortunately Flash is very picky. However, it **is** easy to adapt.

`bColor=0xccccff&fontname=Tahoma&fontsize=20&boxHeight=500&boxWidth=550&clozeText=The #cell cycle includes cell growth when the numbers of #organelles increase and the #chromosomes are copied when the two strands of each #DNA molecule separate and new #strands form alongside them. @ @ The cell cycle then completes with the division of each set of chromosomes into two #daughter cells. This is called #mitosis.`

The text file is a made up of a consecutive set of variable names followed by their values.

Here they are, together with what they do.

`bColor=0xccccff` - sets the background colour of the box containing the text. Flash uses a 6 figure code for each colour (e.g. `ccccff`). This code is given in hexadecimal which is why there are letters as well as numbers. Without going into how it works in detail, the code is really three pairs of digits giving the amount of red, green and blue needed to make the colour. Put "color picker" into your favourite search engine and it will list any number of online tools that help you pick a colour and get its hexadecimal code. I have included a [link to an example](#) here. Most of these tools precede the code with a `#` but Flash uses **0x** instead. As an example you could make the background white by replacing `0xccccff` with `0xffffffff`.

`fontname=Tahoma` - sets the text font to Tahoma. If you enter something that the system doesn't recognise as a legal font name, the text will show as Times New Roman or whatever the default font is on the computer being used.

`fontsize=20` - sets the font size to 20. This allows you to try out different values to see what fits best. This is probably the quickest way to get your text to fill the available space.

`boxHeight=500` - sets the height of the box containing the text in pixels. The movie stage height is set to 600 pixels so make sure that this value is less than that to allow you to see the score and reset button. You can change the stage dimensions in the Flash editor (see [later](#)) alternatively, leave these variables as they are and change the font size to get things to fit.

`boxWidth=550` - sets the width of the box containing the text. The stage is set to 800 pixels. You must therefore set this value to less than this since the missing words are displayed to the right of the text box.

`clozeText=The #cell cycle...` - sets the text to be converted into the cloze passage. Note that it does not have "" characters around it. Instead the file uses the `&` character between each variable so that Flash knows where one ends and the next one starts. You will notice that there are two other special characters `#` and `@`.

The hash (`#`) immediately before a word marks it out as a word to be hidden and added to the missing words list.

The `@` character is used to force a new line.

A quick way to produce a cloze activity is to copy some text e.g. from your favourite word processor, and then paste it in after `clozeText=` in a text editor such as notepad. You can then run the cursor along the passage, adding `#` characters to hide words. Save the text file and then run the movie ([clozemaker.swf](#)).

You will probably need to alter the `fontsize`, `boxHeight` and `boxWidth` values to get it to fit properly. The animation sorts everything else out automatically.

As in life, there are a number of limitations.

1. The [clozetext.txt](#) file and the [clozemaker.swf](#) files must be in the same folder (unless you adapt the Flash document file - see [below](#)).
2. The [clozetext.txt](#) file cannot be called anything else (unless you adapt the Flash document file - see [below](#)).
3. I am not clever enough to work out how to alter the movie's stage size when the movie is running and so you are limited to whatever size is set in the Flash document file. I have chosen 800 x 600 but you can alter it from

within the Flash editor.

4. Some text characters might be interpreted by Flash in a way other than the way you intended so if something odd happens, it might be because it is being confused by something in the text.
5. The movie isn't clever enough to allow you to hide the same word twice.
6. The movie isn't clever enough to have extra words that don't fit a gap.

Here are some easy ways to adapt the movie by loading the [clozetext.fla](#) file into the Flash editor.

Alter the stage colour by clicking on the stage and choosing a colour in the properties window.

You could also alter the stage size in the same properties window.



Alter the clues and gap colours by altering the value of the **gapText** variable in frame 1 of the scripts layer.

Alter the text colour by altering the value of **textColour**. Both of these variables use the same six figure hexadecimal system as the background colour.

startX and **startY** give the offset for the text box on the screen and **indent** gives the offset for the text within the text box.

You can alter the movie so that you can store different text files in the same folder by altering **fileName**. You must also publish each movie with a different name.

```
1 // create some global variables - don't mess with these
2 var wordCount;
3 var bigWord;
4 var answerCount;
5 var answered = new Array();
6 var words = new Array();
7 var wordObject = new Array();
8 var randomPlace = new Array();
9 var gap;
10 var score = 0;
11 var xdif;
12 var ydif;
13 var mymc;
14 // stuff you can alter fairly safely
15 var startX = 20;
16 var startY = 20;
17 var indent = 10;
18 var scoreDisplay = "";
19 var gapTextColour = 0x660000;
20 var textColour = 0x0000ff;
21 var fileName = "clozetext.txt";
```

Alter anything else at your peril!

For the sake of completeness, I have also produced two alternative versions of clozemaker. You can download zipped archive files of them both below.

[clozemaker_novars](#) stores all the variables in the flash movie leaving just the passage in the text file.

[clozemaker_allvars](#) stores all the variables in the text file. This means you can change the look of the activity without editing the Flash document.