

Script Pre-Processor

The C pre-processor is a powerful tool that allows programmers many options when writing their code - among these options are commands such as `#include ""`, which allows the user to include the contents from one code file into another.

Setting Up

In the below ZIP-File a copy of [MCP](#) is included as the pre-processor. I have also written a batch script which will call the pre-processor with the appropriate arguments as well as a display error messages & log them into a text file.

[Zip File Download](#)

The next thing to do is set up notepad++ to use the batch file and MCP. Extract the contents of the zip file to a safe place (For example in a new folder in the amnesia directory). Open up notepad++ and go into the "run" menu and click "run" (or press F5). Into this dialogue box put the following:

```
[FOLDER]\PreProcess.bat $(CURRENT_DIRECTORY) $(FILE_NAME) $(NAME_PART).hps
```

Replace [FOLDER] with the folder the batch file is in. For example:

```
C:\Program Files (x86)\Amnesia\HPS_PREPROCESSOR\PreProcess.bat  
$(CURRENT_DIRECTORY) $(FILE_NAME) $(NAME_PART).hps
```

It is recommended that you then click save and bind the command to something like F6 - otherwise you will have to enter this every time you start notepad++ .

Warning: Do not press "run" (or F6 if you bound it) on a script file which ends in ".hps"! This will wipe the script file!!! Give your script files which use the pre-processor a different extension (E.g .phps or .lhps). u

Usage

The files that exist before pre-processing will have to have a different extension to the one after pre-processing. The simplest way to do this is to call your script files something like "level.phps" so that the pre-processor writes "level.hps". Consider the following test code:

```
//level.phps//  
#include "inclusion_test.phps"  
  
void OnEnter()  
{  
    AddDebugMessage("File 1",false);  
}
```

```
//inclusion_test.php//
void OnStart()
{
    AddDebugMessage("File 2",false);
}
```

Running the script from test.php (F6 if you followed the instructions above) should create "test.hps":

```
void OnStart()
{
    AddDebugMessage("File 2",false);
}

void OnEnter()
{
    AddDebugMessage("File 1",false);
}
```

Which is what the game will see from test.map and run. When you distribute your custom stories only this exported file is required. Obviously, since we are now using a C pre-processor there is a whole [host of other things you can do](#).

Reccomended Extra Installation Setup

You may notice that stuff like #include isn't color coded - and that your new script files ".phps" aren't automatically recognised as HPS files! I Have updated the notepad++ files to fix this (Note that these are updated versions of the [overhauled notepad++ files](#), which provide a fixed function list, folding regions and a new color scheme)

[Download](#)

Installation (Steps 1→4 are optional but recommended):

1. Close notepad++
2. Go to where you installed notepad (Probably C:\Program Files\Notepad++ or C:\Program Files (X86)\Notepad++\)
3. Go to the folder "Plugins" Then to the folder "APIs"
4. Copy across the downloaded version of "hps.xml" into this folder. If you are prompted to overwrite, say yes.
1. Start notepad++
2. Go into view→ User-Defined Dialogue
3. On the drop-down box, if there is the option to select "HPS", select it and click "Remove"
4. Click import, and import "UserDefinedDialogue.xml"

You will now have a fixed functions list for amnesia (Adds missing functions & keywords, removes non-existing ones), as well as a new color scheme, folding regions (+ - /** */ *Begin End*) etc. ".hps", ".phps", ".lhps" are now detected, and "#..." are coloured correctly.

Original thread: <http://www.frictionalgames.com/forum/thread-12023.html>

From:

<https://wiki.frictionalgames.com/> - **Frictional Game Wiki**

Permanent link:

<https://wiki.frictionalgames.com/hpl2/resources/preprocess?rev=1336434645>

Last update: **2012/05/08 00:50**

