
Subject: Re: A Few Simple Leveledit Questions
Posted by [_SSnipe_](#) on Fri, 13 Jul 2007 01:01:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

dont know what u mean but heres the whole section

```
/* Renegade Scripts.dll
Example Plugin Code
Copyright 2007 Whitedragon(MDB), Jonathan Wilson
```

This file is part of the Renegade scripts.dll

The Renegade scripts.dll is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2, or (at your option) any later version. See the file COPYING for more details.

In addition, an exemption is given to allow Run Time Dynamic Linking of this code with any closed source module that does not contain code covered by this licence.

Only the source code to the module(s) containing the licenced code has to be released.

```
*/
```

```
/* This is designed to serve as both an example on how to make a plugin and to give users the
basic framework of a plugin.
```

```
    The plugin is simple: it creates an object creation hook and attaches the script
"Plugin_Example_Script" to all objects.
```

```
    The script prints out a message whenever an object is created or destroyed.
```

```
    There are also examples of the new format for bhs.dll hooks.
```

```
*/
```

```
#include <fstream>
#include "scripts.h"
#include <stdarg.h>
#ifdef WIN32
#define WIN32_LEAN_AND_MEAN
#include <windows.h>
#endif
#include "engine.h"
#include "gmain.h"
#include "plugin.h"
```

```
//This is called in Set_Script_Commands the same way that SSGM_Secondary_Load is called in
SSGM.
```

```
//Keeps the loading code specific to the plugin out of dllmain.cpp.
```

```
//This is not required, you can do your loading code anyway you want.
```

```
void Plugin_Load() {
```

```
}
```

//Called in SSGM_Plugin_Unload. Keeps the unloading code specific to the plugin out of
dllmain.cpp.

//This is not required, you can do your unloading code anyway you want.

```
void Plugin_Unload() {
```

```
}
```

```
bool Is_Mod(const char *File, const char *Name) {
```

```
    ifstream; file(File, ios::in);
```

```
    string; tmp;
```

```
    while(file >> tmp) {
```

```
        if(strcmp(Name, tmp.c_str()) == 0) {
```

```
            return true;
```

```
        }
```

```
    }
```

```
    return false;
```

```
};
```

```
bool class testChatCommand : public ChatCommandClass {
```

```
void Triggered(int ID,const TokenClass &Text,int ChatType) {
```

```
    if (Is_Mod("Test_Mods.txt",Get_Player_Name_By_ID(ID))) {
```

```
        GameObject *obj = Get_GameObj_By_Player_Name(Text[0].c_str());
```

```
        Change_Team(obj,2);
```

```
        Commands->Give_Powerup(obj,"CnC_POW_AutoRifle_Player_Nod",false);
```

```
SimpleDynVecClass<GameObject*> List = Get_All_Objects_By_Preset(2,"Signal_Flares");
```

```
int Rand = Commands->Get_Random_Int(0,List.Count());
```

```
GameObject *RandObj = List[Rand];
```

```
Commands->Set_Position(obj,Commands->Get_Position(RandObj));
```

```
    }
```

```
}
```

```
};
```