



Novel Games

**Lightning Flash Game
Customization Guide**
Version 1.6

Modification History

Version	Date	Description
1.6	2008-2-4	Added Troubleshooting section
1.5	2007-12-31	Added parts about ActionScript 2.0
1.4	2007-12-28	Changed LIGHTNINGPROBABILITY to LIGHTNING_MINTIME and LIGHTNING_MAXTIME, added MESSAGE_ENDTIME
1.3	2007-2-22	Updated Optional Functions section
1.2	2006-1-27	Added lightning.htm
1.1	2006-12-20	Added the TIME_BADPICTURE parameter
1.0	2006-9-11	First Draft

Table of Contents

Introduction	1
Directory Structure	2
The Configuraton File (Config.as)	4
Optional Functions (MainDevelopment.as)	5
Troubleshooting	6



Introduction

This document describes the source codes of Lightning Flash Game and the ways to customize the game.

In order to use the source code your computer should have Flash Professional CS3 or above installed (You can also use Flash 8, but you will only be able to compile the game using ActionScript 2.0). You should also have basic knowledge of the Flash authoring environment, such as the steps to change a picture and the steps to change the value of a variable, etc.



Directory Structure

The source codes are arranged in the following directory structure:

```
+
+-- lightning.pdf (this file)
+-- common
|   +-- com
|       +-- novelgames
|           +-- flashgames
|               +-- common
|                   +-- MainDevelopment.as
|               +-- commonAS2
|                   +-- MainDevelopment.as
+-- lightning
    +-- lightning.fla
    +-- lightning.swf
    +-- lightning.htm
    +-- lightningAS2.fla
    +-- lightningAS2.swf
    +-- lightningAS2.htm
    +-- com
        +-- novelgames
            +-- flashgames
                +-- lightning
                    +-- Config.as
                    +-- other ActionScript 3.0 .as files
            +-- lightningAS2
                +-- Config.as
                +-- other ActionScript 2.0 .as files
```

To build the game, you should open lightning\lightning.fla in Flash CS3 and then publish it. The file lightning.swf in the same directory is the compiled swf file for your reference. If you only have Flash 8 or if you want to use ActionScript 2.0 then you should open lightning\lightningAS2.fla instead.

The file lightning.htm contains sample HTML codes for you to use in your web site.



All the ActionScripts are arranged in class files put in the lightning\com\novelgames\flashgames\lightning folder. (The ActionScript 2.0 .as files are put in the lightning\com\novelgames\flashgames\lightningAS2 folder).

Depending on what you'd like to customize, you will need to modify different files:

- 1) If you want to change the graphics and the sounds, then you should edit the fla file.
- 2) If you want to tune some parameters of the game, then you should edit the Config.as file.
- 3) If you want to add the support of high scores or change the more games button link, then you should edit the MainDevelopment.as file.
- 4) If you want to change the behaviour of the game and such changes cannot be achieved by tuning the game parameters as in 2) above, then you should edit the ActionScript .as files.

We will discuss 2) and 3) in the next two sections.



The Configuraton File (Config.as)

In this section we'll explain the parameters in Config.as and how they affect the behaviour of the game.

Unless otherwise stated, the units of the parameters are expressed as follows:

- length units are express in pixels;
- time units are in milliseconds;
- angle units are in degrees;
- speed units are in pixles per second;
- rotation speed units are in degrees per second;
- acceleration units are in pixels per second squared;
- colour units are in 0xRRGGBB format.

Parameter	Description
LIGHTNING_MINTIME	The minimum time (in milliseconds) after which the lightning will occur
LIGHTNING_MAXTIME	The maximum time (in milliseconds) before which the lightning will occur
TIME_BADPICTURE	The maximum time in milliseconds spent to take a nice picture
SCORE_BASE	The base score
SCORE_PERMILLISECOND	The score deducted for each millisecond of reaction time
MESSAGE_ENDTIME	The time (in milliseconds) the end message will be shown for



Optional Functions (MainDevelopment.as)

If you want to support high scores then you'll need to modify the following 2 functions in this file:

```
public function showHighScores():void {  
    trace("showHighScores()");  
}  
  
public function showEnterHighScore(score:int):void {  
    trace("showEnterHighScore(" + score + ")");  
}
```

The showHighScores function will be called if the high scores button is clicked.

The showEnterHighScores function will be called when the game has finished and the score of the user has been calculated.



Troubleshooting

If you get compiler errors when you try to compile the games, try the following:

- 1) Select "Control" and then "Delete ASO Files" from the menu bar
- 2) Compile the game again

If you still get compiler errors, try the following:

- 1) Select "Control" and then "Delete ASO Files" from the menu bar
- 2) Restart Flash
- 3) Compile the game again

If you still get compiler errors, try the following:

- 1) Select "Control" and then "Delete ASO Files" from the menu bar
- 2) Edit all the .as files (by adding a space at the end for example), and then save the files
- 3) Restart Flash
- 4) Compile the game again

If you still get compiler errors, then please contact us at support@novelgames.com