Novel Games

Juggling Flash Game Customization Guide

Version 1.3

Modification History

Version	Date	Description
1.3	2008-2-4	Added Troubleshooting section
1.2	2007-12-31	Added parts about ActionScript 2.0
1.1	2007-12-11	Added INITIALWAITTIME and MESSAGE_GAMEOVERTIME parameters
1.0	2007-8-17	First Draft

Table of Contents

Introduction

This document describes the source codes of Juggling 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:

```
+
+-- juggling.pdf (this file)
+-- common
|   +-- com
|       +-- novelgames
|           +-- flashgames
|               +-- common
|                   +-- MainDevelopment.as
|               +-- commonAS2
|                   +-- MainDevelopment.as
+-- juggling
    +-- juggling.fla
    +-- juggling.swf
    +-- juggling.htm
    +-- jugglingAS2.fla
    +-- jugglingAS2.swf
    +-- jugglingAS2.htm
    +-- com
        +-- novelgames
            +-- flashgames
                +-- juggling
                    +-- Config.as
                    +-- other ActionScript 3.0 .as files
                +-- jugglingAS2
                    +-- Config.as
                    +-- other ActionScript 2.0 .as files
```

To build the game, you should open juggling\juggling.fla in Flash CS3 and then publish it. The file juggling.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 juggling\jugglingAS2.fla instead.

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

All the ActionScripts are arranged in class files put in the juggling\com\novelgames\flashgames\juggling folder. (The ActionScript 2.0 .as files are put in the juggling\com\novelgames\flashgames\jugglingAS2 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
INITIALWAITTIME	The initial time (in milliseconds) the game will wait until the first ball drops
HAND_LEFTCATCHX	The x coordinate (in pixels) of the hand when it is catching the ball on the left
HAND_LEFTRELEASEX	The x coordinate (in pixels) of the hand when it is releasing the ball from the left
HAND_RIGHTCATCHX	The x coordinate (in pixels) of the hand when it is catching the ball on the right
HAND_RIGHTRELEASEX	The x coordinate (in pixels) of the hand when it is releasing the ball from the right
HAND_Y	The y coordinate (in pixels) of the hands
HAND_CATCHTOLERANCE	The maximum distance (in pixels) of the catching ball from the hand when the arrow key is pressed
HAND_MAXRELEASETIME	The maximum time (in milliseconds) a ball can stay in the hand
HAND_MINRELEASETIME	The minimum time (in milliseconds) a ball can stay in the hand

HAND_RETURNTIME	The time (in milliseconds) the hand takes to return to its catching position after tossing a ball
BALLS_ACCELERATION	The downward acceleration (in pixels per second squared) of the balls
BALLS_MINY	The minimum y coordinate (in pixels) of the ball when tossing up
BALLS_MAXY	The maximum y coordinate (in pixels) of the ball when tossing up
BALLS_COLOURS	The possible colours of the balls, in 0xRRGGBB format
BALLS_ADDCATCHCOUNT	The number of times each ball must be caught before a new ball is given
BALLS_GROUNDY	The y coordinate (in pixels) of the ground
EXPLODE_STAYTIME	The number of milliseconds the explosions will stay on the screen
EXPLODE_DISAPPEARTIME	The number of milliseconds the explosions will take to disappear
LIFE	The life
SCORE_BALL	The basic score awarded for catching a ball
SCOREPOPUP_LEFTX	The x coordinate (in pixels) of the score popup when it appears on the left
SCOREPOPUP_RIGHTX	The x coordinate (in pixels) of the score popup when it appears on the right
SCOREPOPUP_Y	The y coordinate (in pixels) of the score popup when it appears
SCOREPOPUP_MOVEHEIGHT	The height (in pixels) the score popups will move to
SCOREPOPUP_MOVETIME	The number of milliseconds the score popups will take to move
SCOREPOPUP_DISAPPEARTIME	The number of milliseconds from the appearance of a score popup will it start to disappear

`MESSAGE_GAMEOVERTIME`	The time (in milliseconds) the game over 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