

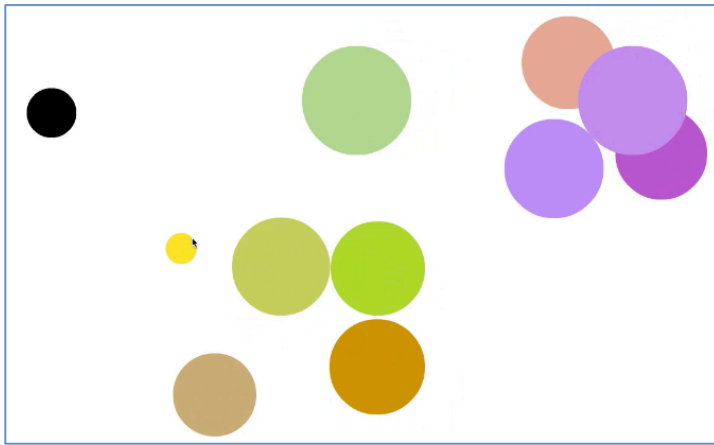
Project 1B: Avoid the Balls

due 9:00am Wednesday, September 26, 2012.

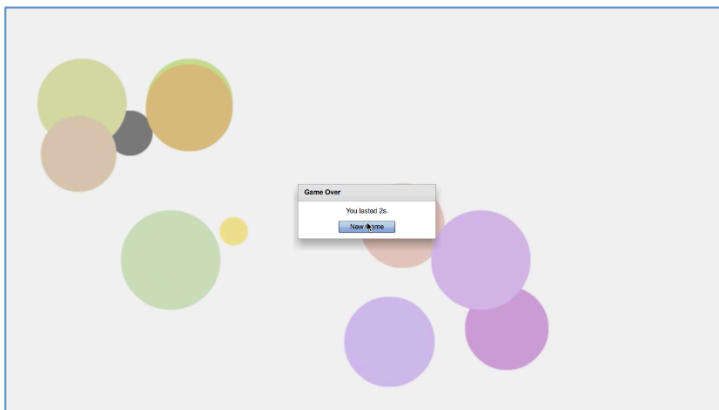
Note: Please include your name in a comment.

Goal

The goal of part B is for you to start getting familiar with ActionScript syntax. We are going to create a game with the (very original) name “avoid the balls,” shown below:



In this game, the user plays the role of the bright yellow ball, which is controlled the ball by the mouse cursor. The singular goal of the game is to avoid touching the colorful enemy balls. When the player touches an enemy ball, the game asks the user if they want to start a new game, saying how long they lasted since the game started in seconds:



The only exception is the black ball, which is actually not an enemy. Instead it helps the player stay alive in the game. When the player touches the black ball, all the colorful enemy balls will go into slow motions, and the player will become invincible for 3 seconds. During

the 3 seconds, the game will not end even if the player touches the enemy balls. After 3 seconds, the game goes back to normal and the enemy balls go back to their original speed. See the demo video.

Rather than creating a new project from scratch, we are going to import an existing project and make changes. First, download the Project1B.fxp project file from Blackboard. Then, in Flash builder, click File -> Import Flex Project (FXP) to import the file.

Files

There are four files in Project 1B:

- Project1B.mxml: The actual application
- PlayerModel.as: A class that keeps track of the position, radius, velocity, and color of a ball.
- EnemyPlayerModel.as: A Subclass of PlayerModel that keeps track of the wall positions, and automatically updates its position depending on the velocity, checking for collisions. This class is used for all of the “enemy” balls
- PlayerView.mxml: A subclass of Ellipse that has a PlayerModel field and binds its width, height, left, top, and fill properties to the appropriate fields of its PlayerModel.

Your solution will be contained in Project1B.mxml

Details

You are responsible for writing the onHit method. onHit gets called whenever the user collides with a ball. There will be one black ball, and several enemy balls. The ball’s radius is passed in as a parameter to the onHit method. If onHit is called when the user is playing, then it should do the following:

- If it’s the black ball (radius of 40) then call the invincibleMode method to let the game go to invincible mode.
- But if it’s not the black ball (radius more than 40), then it should pop up a modal dialog box telling the user how many **seconds** they lasted, and ask them if they want to play a new game. There should only be one button, with the label “New Game” for this dialog box.

If the game is in the invincible mode, then the onHit method should do nothing – nothing happens when hitting a ball when invincible. The `isInvincible` variable tells if the game is in the invincible mode.

Only **one** alert should ever be popped up at a time – while the alert is popped up, the balls will continue moving in the background, but no more alerts will be popped up if the user hits an enemy ball. To prevent multiple dialog boxes from showing up, you should check and set the `playing` variable.

When the user responds to the alert, call the `onDialogRespond` method, which will reset the position of all the balls.

Once again, you only need to write the `onHit` method. No changes outside of this method are required.

Hints

- Documentation for the `Alert` class can be found at:
http://help.adobe.com/en_US/flex/using/WS2db454920e96a9e51e63e3d11c0bf69084-7da0.html
 - The `buttonWidth` property of `Alert` can be set to customize the button width
- Documentation for the `Date` class can be found at:
<http://livedocs.adobe.com/flex/3/langref/Date.html>
- The string concatenation operator in ActionScript is '+'. For example the expression `"A"+"B"` evaluates to `"AB"`. Integer conversion is done automatically. For example, the expression `"A"+1` evaluates to `"A1"`.

Grading

| | |
|--|-----------|
| Turned in & compiles: | 10 points |
| Player ball becomes invincible when touches the black ball: | 10 points |
| Enemy balls slow down when player ball touches the black ball: | 5 points |
| Correct measurement for time lasted: | 10 points |
| Correct units for time lasted: | 5 points |
| Only one dialog box pops up: | 10 points |
| Dialog box button says "New Game": | 5 points |
| Commenting & code clarity: | 10 points |

Critical Thinking

You are not required to turn in answers to any of the questions in this section, but we recommend that you explore and think about some of the questions.

1. What are some ways to make the game easier or harder that take minimal changes to the code?
2. Right now, the "enemy" balls move by having a timer that goes off every 3 milliseconds to update their x and y positions, moving them to their new values. While this was OK for a small game, it does not scale. Why?
3. This game has some usability problems. For one, when the user hits ESC on the "New Game" dialog box, it starts up a new game, which is not consistent with other applications. Another is that the game never indicates why the game ended. How might we fix these issues? Can you think of some more usability issues?

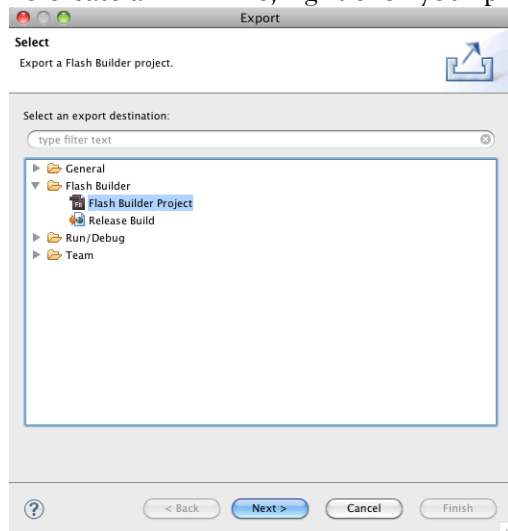
Turning it in

Project 1b is due by 9:00am September 26, 2012 as an FXP file (I describe how to create an FXP file below). Please submit it on Blackboard. Late entries will be penalized 5% for every late day. If you have any questions, please email or message me during IM hours.

Zippping up your project

An FXP file is basically a zipped archive file that contains your project source, build files, and other metadata. You will turn in your projects as FXP files.

To create an FXP file, right click your project, and click “Export”.



Select “Flash Builder Project” under “Flash Builder” and click “Next”.

Choose the project and your directory and click “Finish”

More information about the FXP file format can be found at:

http://help.adobe.com/en_US/flashbuilder/using/WSbde04e3d3e6474c4ef5411412477fbf920-8000.html