

Bouncing Objects Canvas Animation

v1.0

Cpt S 489

Read all instructions carefully before writing any code. Do all work individually!

The zip that contains these instructions also includes an HTML file. You may implement all your JavaScript code inside this HTML file for this assignment. You may also create additional .js files if you wish, but if you do, make sure that they are properly referenced by the HTML page and included with your submission.

In this assignment, you will write some simple animation code to render circular objects on a canvas. There are 2 states that your code must support, animation on and animation off. When animation is on, all dynamic objects will move according to their velocity vectors. At the moment when the animation is toggled off, all dynamic objects must freeze at their current locations. They must resume from their current locations when the animation is toggled back on. You'll also have static objects that do not move regardless of whether animation is on or off. Some code in the .html file includes the basics for setting up rendering with a timer and having a bool flag to indicate whether animation is on or off.

When the user [clicks on the canvas](#), make sure the following happens:

- If the shift key is held, then a static object is created at the click location. The following must be true for static objects:
 - they are created with a random radius in the range [20,40]
 - they are created with a random color from the array of colors already declared in JavaScript within the .html file
 - they are rendered as circular objects, using the color that was associated with them during creation, and have an 'S' rendered in black in the middle
- Otherwise, a dynamic object is created at the click location. This object must be created with:
 - a random radius in the range [10,20]
 - a random color from the array of colors already declared in JavaScript within the .html file
 - a random velocity with x and y components in a range something like [-5,5] (you can use something like [-10,10] if you prefer more variety, just don't go overboard with it)

Clicking to create new objects can be done with the animation on or off. Dynamic objects move each frame when animation is on. You can just add their velocity components to the corresponding position components. In other words, your velocity units can be pixels-per-frame. Dynamic objects also must bounce off the walls of the canvas. Make the background color of the scene gray, or some other color that doesn't visually conflict with the objects and isn't the same as the background of the page (white), so that you can ensure that the borders of the canvas region are easy to distinguish. All the above requirements must be fulfilled for 2/3 points.

For the 3rd point, dynamic objects must also bounce off static objects. Dynamic objects do not have to bounce off each other, but they do have to bounce off walls and the static objects. The bouncing doesn't have to be perfectly realistic, as long as it looks ok in general. The demo MP4 video included with this assignment shows an example of this. The physics are not technically correct in the demo, but it would still get 3/3. If an object is coming in straight from the right side, hits a static object, then bounces straight up, then this won't suffice for credit. Neither will dynamic objects flying through static objects or randomly teleporting to new locations. It has to look *somewhat* accurate/realistic. Again, watch the MP4 video to get an idea.

Scoring:

3/3 if all above requirements are met

2/3 if all requirements, except collision with static objects, are met

1/3 if not all basic requirements are met, but there are still static and dynamic objects, and animation works (imperfections like not bouncing correctly off walls, not having random colors, sizes and velocities, not having an 'S' rendered within static objects, and other such things are what can lower you from 2 to 1 point)

0/3 for anything else