

KEY IN-CLASS STUDENT ACTIVITY 2.2: CHOOSE AND CONFIGURE HTML5 TAGS TO DISPLAY GRAPHICS

Lesson Objective 2.2:

Choose and configure HTML5 tags to display graphics.

Resources, software, and additional files needed for this lesson:

1. Microsoft® Expression® Web 4 (preferred) with latest service packs, or Notepad.
2. Internet access (Internet Explorer® 9 or later is preferred).

Guiding questions:

1. **What criteria determine whether the developer uses the <canvas> tag or the <SVG> tag?**
Developer knowledge and skillset, the size of the screen, and the number of graphic objects are key elements in making the decision. Generally, as the size of the screen increases, the canvas begins to degrade as more pixels need to be drawn. As the number of objects increases on the screen, SVG begins to degrade as they are being added to the Document Object Model (DOM).
2. **Why is it recommended to use transformations when drawing on the canvas?** Transformation is a modification layer that sits between the commands issued and the output on the canvas display. A key recommendation for reusable code is to draw at the origin and apply transformations such as scale, translate, and rotate to modify the drawing code into the final appearance.
3. **What code can developers include to test for browser support of the new HTML5 <canvas> tag?**
Use a try/catch statement and add text to the <canvas> tag element. If the canvas element is supported, the text will be ignored; otherwise (if it is not supported), it will display the message: "HTML5 Canvas is not supported in your browser."

```
try{  
  
    document.createElement("canvas").getContext("2d");  
  
    document.getElementById("support").innerHTML = "HTML5 Canvas is supported  
in your browser.";}  
  
catch (e) {  
  
    document.getElementById("support").innerHTML = "HTML5 Canvas is not  
supported in your browser."; }
```

Student activity:

Directions to the student:

Read the following scenario. Work in pairs to program a game using SVG and canvas. Answer the questions below. When you are finished programming and testing, verify your answers with the instructor. Share your answers with the class.

Scenario:

Mu Han works as a web developer for WingTip Toys. He previously worked extensively with HTML4 and is now preparing the company websites for the upgrade to HTML5. His supervisor wants to add some mini-games on the website during the transition.

Mu knows that the HTML5 canvas element can help to make dynamic animations that work well for games on the web. Mu has decided to create the games at home and have his twin sons test them before he introduces them on the WingTip Toys website.

Mu Han discovered this Microsoft site for programming simple games using canvas or SVG:

[http://msdn.microsoft.com/en-us/library/gg589521\(v=VS.85\).aspx](http://msdn.microsoft.com/en-us/library/gg589521(v=VS.85).aspx)

Content:

1. Copy and paste the URL into a browser.
2. Work with a partner. One partner will program the game using SVG. The other partner will use the canvas. Compare your experiences with both techniques.
 - a. Which game was easier to develop and why?
Answers may vary. Concentrate on why one was easier to use than the other.
 - b. Do both games provide the same results and performance?
Answers may vary. Try to find instances in which the results or performance were different, and challenge the students to identify why this happened.
 - c. Discuss ways to change the game or add challenges.
Answers may vary depending on the games that students develop. They might suggest adding a second level.