

1 Description

This assignment is worth 10 points, or 10% of your term grade. It is due on September 21.

Instructions for electronic submission of assignments will be given out in class on September 6.

A rubric for grading this assignment will also be given out in class on September 6.

The purpose of this assignment is to highlight some of the fundamental aspects of game programming, such as:

- definition of graphical game objects
- animation of graphical game objects
- interaction (e.g., collision) between graphical game objects
- user interaction with graphical game objects
- keeping score

For this assignment, you will implement these in HTML5, which is a relatively simple environment for programming games. Later in the semester, you will implement these in more sophisticated environments.

So, for this first assignment you will program a **classic arcade game** in HTML5. Depending on your experience level with HTML5, you can decide how sophisticated to make this. For example, if you have never programmed in HTML5 or Javascript before this semester, then I will expect you to produce something like a simple version of **Pong**, something like this:

<http://www.classicgamesarcade.com/games/pong.swf>

On the other hand, if you took the HCI class with me in the Spring and you already did some HTML5, or you learned it some other way, then I will expect you to produce something more sophisticated, like **Pac-Man**:

<http://www.classicgamesarcade.com/games/pacman.swf>

If you want to look at some examples of classic arcade games, then I suggest you go to this site:

<http://www.classicgamesarcade.com/games/pong.swf>

where you will find a wide range of replicated games that you can run in a browser.

The next section contains links to tutorials and reference materials for learning HTML5 to program games.

2 Learning Game Programming in HTML5

There are 4 environments/languages that you should get to know in order to be able to program in HTML5:

- HTML — Hyper-Text Markup Language: the fundamental language interpreted by browsers
- CSS — Cascading Style Sheets: defines how things look in a web page
- JavaScript — client-side scripting that defines how things behave in a web page
- HTML5 — newest whiz-bang version of HTML that includes expanded types of elements

2.1 HTML

If you don't know any HTML, then start with an HTML Beginner tutorial, like this one:

- <http://www.htmldog.com/guides/htmlbeginner/>

or this one:

- <http://www.w3schools.com/html/>

These pages both discuss beginning, intermediate and advanced features of HTML.

2.2 CSS

If you know some HTML, but don't know any CSS, then start with a CSS Beginner tutorial, like this one:

- <http://www.htmldog.com/guides/cssbeginner/>

or this one:

- <http://www.w3schools.com/css/>

These pages both discuss beginning, intermediate and advanced features of CSS.

2.3 JavaScript

If you don't know any JavaScript, then start with a JavaScript Basics tutorial, like this one:

- <http://www.w3schools.com/js/>

2.4 HTML5

If you don't know anything about HTML5, then start with an HTML5 tutorial, like either of these:

- <http://www.w3schools.com/html5/>
- <http://diveintohtml5.info/>

You should also go through my *HTML5 Canvases* lab exercises.

2.5 Books

If you want more detail than the tutorials provide, then you can also find some very good in-depth books on the O'Reilly web site:

- <http://oreilly.com/>

Books are available both in print and electronic formats.

Some specific O'Reilly books:

- *JavaScript: The Definitive Guide, 6th Edition.* by David Flanagan. O'Reilly Media, 2011.
- *CSS: The Definitive Guide, 3rd Edition.* by Eric A. Meyer. O'Reilly Media, 2006.
- *HTML5: Up and Running.* by Mark Pilgrim. O'Reilly Media, 2010.

2.6 Online tutorial sites

As you might have guessed from the links above, the two sites that I recommend for online tutorials are:

- <http://www.w3schools.com/>
This site contains tutorials on HTML, HTML5, CSS, CSS3, JavaScript (and many other things).
- <http://www.htmldog.com/>
This site contains tutorials on HTML and CSS, and as well as references.

3 References

3.1 Some HTML form elements

from *JavaScript: The Definitive Guide*, page 397

HTML property	Type property	Event handler	Description and events
<input type="button"> <button type="button">	"button"	onclick	a push button
<input type="checkbox">	"checkbox"	onchange	a toggle button without radio button behavior
<option>	none	none	a single item within a Select object; event handlers are on the Select object, not on individual Option objects
<input type="password">	"password"	onchange	an input field for password entry; typed characters are not visible
<input type="radio">	"radio"	onchange	a toggle button with radio button behavior
<input type="reset"> <button type="reset">	"reset"	onclick	a push button that resets a form
<select>	"select-one"	onchange	a list or drop-down menu from which one item may be selected (also see <option>)
<select multiple>	"select-multiple"	onchange	a list from which multiple items may be selected (also see <option>)
<input type="submit"> <button type="submit">	"submit"	onclick	a push button that submits a form
<input type="text">	"text"	onchange	a single-line text entry field
<textarea>	"textarea"	onchange	a multi-line text entry field

3.2 Some CSS properties

Colors and Backgrounds

color, background-color, background, background-image, background-repeat, background-position, background-attachment

Text and Fonts

font, font-family, font-size, font-weight, font-style, font-variant, line-height, letter-spacing, word-spacing, text-align, text-decoration, text-indent, text-transform, vertical-align, white-space

Positioning and Display

position, top, right, bottom, left, clip, overflow, z-index, float, clear, display, visibility

The Box Model - dimensions, padding, margin and borders

padding, padding-top, padding-right, padding-bottom, padding-left,
border, border-top, border-right, border-bottom, border-left,
border-style, border-top-style, border-right-style, border-bottom-style, border-left-style, border-color, border-top-color, border-right-color, border-bottom-color, border-left-color, border-width, border-top-width, border-right-width, border-bottom-width, border-left-width, outline, outline-style, outline-color, outline-width,
margin, margin-top, margin-right, margin-bottom, margin-left
width, height, min-width, max-width, min-height, max-height

Lists

list-style, list-style-type, list-style-image, list-style-position

Tables

table-layout, border-collapse, border-spacing, empty-cells, caption-side

3.3 Some HTML5 Canvas elements and Javascript functions

Below is a quick reference guide to some of the basic JavaScript functions for drawing in an HTML5 canvas. A complete function reference is available here:

http://www.w3schools.com/html5/html5_ref_canvas.asp

Attributes for setting colors and styles

Attribute	Value(s)	Description	Default value
fillStyle	color/style	The fill-color of the drawing.	black
strokeStyle	color/style	The stroke-color of the drawing,	black
lineWidth	number	The width of the drawing stroke,	1
lineCap	butt, round, square	The style of the ending of the line	butt
lineJoin	bevel, round, miter	The style of the corners of the line	miter

Functions for drawing lines, rectangles, circles and arcs

Method	Description
moveTo(x,y)	Positions the drawing pen at the specified point, without creating a line
lineTo(x,y)	Draws a line from the current position of the drawing pen to the given point
rect(x,y,w,h)	Draws an unfilled rectangle
strokeRect(x,y,w,h)	Draws the lines of a rectangle using the color/style of the strokeStyle attribute
fillRect(x,y,w,h)	Draws a filled rectangle using the color/style of the fillStyle attribute
clearRect(x,y,w,h)	Clears a rectangle area by erasing all pixels inside the specified area
arc(x,y,r,a0,aN,clockwise)	Creates a circle, or part of a circle, with center at (x,y) and radius r , starting at angle $a0$ and ending at angle aN ; if "clockwise" is set to true, then the arc is drawn clockwise; otherwise it is drawn counterclockwise
arcTo(x1,y1,x2,y2,r)	Creates an arc of radius r that extends from point $(x1,y1)$ to point $(x2,y2)$

Attributes for drawing text

Attribute	Value	Description	Default
font	font properties	Specifies font properties used for writing on the canvas, using the same format as CSS	10px sans-serif
textAlign	start, end, left, right, center	The alignment of the text	"start"
textBaseline	alphabetic, bottom, hanging, ideographic, middle, top	The vertical alignment of the text	"alphabetic"

Functions for drawing text

Method	Description
strokeText(text,x,y,maxWidth)	Draws text on the canvas using the color set by the strokeStyle attribute.

Attributes for drawing and manipulating images

Attribute	Value	Description
width	number	Specifies the width of the image data object
height	number	Specifies the height of the image data object
data	array	An array containing the rgba values of all pixels in an image data object

Functions for drawing and manipulating images

Method	Description
drawImage(img, x, y)	Display an image in the canvas at position (x,y)
drawImage(img, x, y, w, h)	Display an image in the canvas at position (x,y) , scaled to width w and height h

There are many more JavaScript drawing functions that do things like manipulate the pixels of an image, change the transparency of an image, create composite images, transform images and shapes drawn using "paths", draw curves using quadratic or Bezier functions, create gradients, and draw lines with shadows. Refer to the page listed above for a more complete reference.