

Coder Dojo

2012-02-04
HTML5 - JavaScript
Slideshow

Why coding...

Entry level starting point will include programming.

Source - Daniel B Markham
[http://www.whattofix.com
/blog/archives/2012/01
/programming-is-1.php](http://www.whattofix.com/blog/archives/2012/01/programming-is-1.php)

It used to be there were four tiers of work... uneducated, ... high school graduates... college graduates and finally...an expert.

Newsflash: the second and third tier are going away. In it's place is a single tier: people who are literate and are able to control computers. And we're nowhere near ready for the changes coming.

Programming is the new High School Diploma.

Useful Links... 1

http://polaris.umuc.edu/~flazarus/8j_def/8J_cont15.html

Nasty URL, I know.

Its a collection of JavaScript definitions, which has a nice list of the default handlers(e.g. MouseDown)

And above all...

be cool

Sliding

yes old hands, something new

and Thank You

Mr A Panayotov,
who should be somewhere in there.

Lets start with a standard template

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    Look at these pictures
    <script>
    </script>
  </body>
</html>
```

Now with explanations...

<!doctype html>

Doctype is something that's part of HTML4. In HTML5, HTML is the default.

And yes you have webpages ending in things other than HTML.

asp
php
cgi
pl

We've covered this before

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    Look at these pictures
    <script>
    </script>
  </body>
</html>
```

Which leaves meta

Metadata is information about the data.

charset means this is the set of characters we're using.

UTF-8 is a character coding that has support for many alphabets, including non-western ones. You've might have seen other character sets used in the past, but UTF-8 is being promoted as the new standard. And it's way shorter and easier to remember than previous character encodings for HTML4.

In short, its the new standard until the next one. Live with it.

We're building a slideshow, add images

```
<!doctype html>
```

```
<html>
```

```
  <head>
```

```
    <meta charset="utf-8">
```

```
  </head>
```

```
  <body>
```

Look at these pictures

```
    <img src ="coderdojologo1.png">
```

```
    <img src ="coderdojologo2.png">
```

```
    <img src ="coderdojologo3.png">
```

```
  <script>
```

```
  </script>
```

```
</body>
```

about those images

Remember to use exact filenames.

This step isn't strictly needed, but I want to make sure all 3 images that you've found on your harddisks are displaying correctly.

add javascript

```
<!doctype html>
```

```
<html>
```

```
  <head>
```

```
    <meta charset="utf-8">
```

```
  </head>
```

```
  <body>
```

```
    Look at these pictures
```

```
    <img src ="coderdojologo1.png">
```

```
    <img src ="coderdojologo2.png">
```

```
    <img src ="coderdojologo3.png" name="slide">
```

```
    <script src="slide.js">
```

```
    </script>
```

```
  </body>
```

name="slide"?

we are naming out third image.

Why?

Because names have power...

Or in this case, by naming the element, we can get the javascript to talk to it.

And so to script. Create a slide.js file

```
var image1=new Image()  
image1.src="coderdojologo1.png"
```

```
var step=1;  
document.images.slide.src=eval("image"+step+".src");
```

What did that do?

Lets break it down...

On the HTML file we had

```
<img src ="coderdojologo3.png" name="slide">
```

Here we are declaring an image loading zone and calling it "slide"

Its similar to VAR SLIDE

Its a variable. Anyone need a refresher on variables?

(No shame, we all need reminders sometimes. sometimes I spell my own name wrong)

And on the other side

```
var image1=new Image()  
image1.src="coderdojologo1.png"
```

Working backwards... remember how we put an image in to a webpage?

```

```

Same thing here

variable image1 has a source located at coderdojologo1.png

new Image() means that I've created an object of a predefined type called "Image"

see http://www.w3schools.com/js/js_obj_intro.asp for more

And display...

```
var step=1;
```

new variable called "step" and initialise with 1

```
document.images.slide.src=eval("image"+step+".src");
```

The eval bit forms image1.src. The eval does the "calculation" to the text

```
document.images.slide.src
```

src well we've hit that a few times today

slide remember the loading zone in the html side?

Images its an image

document. usually this is the webpage

Document

If you've poked about with javascript resources on line then you have probably hit

```
document.write("I love tea");
```

http://www.w3schools.com/js/js_statements.asp

This will just stick the phrase "I love tea" somewhere.

To put it in a particular place you need a bit of fine tuning

So far

We've put a single image on the screen using javascript.

But we would like to display more than one (just not at the same time)

So we need to code more.

Yes... Moar!

in javascript...

```
var image1=new Image()  
image1.src="coderdojologo1.png";  
var image2=new Image()  
image2.src="coderdojologo2.png";
```

```
var numberOfImages = 2;  
// this number should match the number of  
// image objects above
```

```
var step=1;  
document.images.slide.src=eval("image"+step+".src");
```

in javascript...

```
var image1=new Image()  
image1.src="coderdojologo1.png";  
var image2=new Image()  
image2.src="coderdojologo2.png";
```

```
var numberOfImages = 2;  
// this number should match the number of  
// image objects above
```

```
var step=1;  
function slideit(){  
    document.images.slide.src=eval("image"+step+".src");  
}  
//call the function first time to start the  
//image appearing  
slideit()
```

and?

Well if you save and run this set of changes, it looks like nothing happened...

so lets add one more thing...

back in javascript...

```
var image1=new Image()  
image1.src="coderdojologo1.png";  
var image2=new Image()  
image2.src="coderdojologo2.png";
```

```
var numberOfImages = 2;  
// this number should match the number of  
// image objects above
```

```
var step=1;  
function slideit(){  
    step++;  
    document.images.slide.src=eval("image"+step+".src");  
}  
//call the function first time to start the  
//image appearing
```


and...

And you now have the second image appearing.

So to loop through the images, we need to loop through the steps.

Stepping through

```
var image1=new Image()
image1.src="coderdojologo1.png";
var image2=new Image()
image2.src="coderdojologo2.png";

var numberOfImages = 2;
// this number should match the number of
// image objects above

var step=1;
function slideit(){
    document.images.slide.src=eval("image"+step+".src");
    if (step < numberOfImages)
        step++;
    //call function "slideit()" every 3 seconds
    setTimeout("slideit()",3000);
}
//call the function first time to start the image appearing
slideit()
```

What?

```
if (step < numberOfImages)
```

```
step++;
```

if the value of step is less than the limit we declared earlier.

if? Remember the murder mystery from last time

```
//call function "slideit()" every 3 seconds
```

```
setTimeout("slideit()",3000);
```

The comment says is all.

setTimeout means call the function after a number of milliseconds.

http://www.w3schools.com/jsref/met_win_settimeout.asp

so we have an image changing

so we need to set it so it refers back to 1 and it becomes a repeating slideshow

Stepping through

```
var image1=new Image()  
image1.src="coderdojologo1.png";  
var image2=new Image()  
image2.src="coderdojologo2.png";
```

```
var numberOfImages = 2;  
// this number should match the number of  
// image objects above  
var step=1;  
function slideit(){
```

```
    document.images.slide.src=eval("image"+step+".src");  
if (step < numberOfImages)  
step++;  
    else  
        step = 1;
```

```
//call function "slideit()" every 3 seconds  
setTimeout("slideit()",3000);  
}
```

```
//call the function first time to start the image appearing
```

So you have a mini slideshow

Its only 2 images (and you may want to remove some of the static images from the html page).

How would you expand it?

Could you automate it?

Is there a way to just declare a "numberOfImages" without the big declaration at the start?

What would you try out?

Just one more thing

This won't work in every browser... so you may need to add a safeguard

Stepping through

```
var image1=new Image()
image1.src="coderdojologo1.png";
var image2=new Image()
image2.src="coderdojologo2.png";

var numberOfImages = 2;
// this number should match the number of image objects above

var step=1;
function slideit(){
//if browser does not support the image object,
//exit.
    if (!document.images)
        return;
    document.images.slide.src=eval("image"+step+".src");
    if (step < numberOfImages)
        step++;
    else
        step = 1;
    //call function "slideit()" every 3 seconds
    setTimeout("slideit()",3000);
}
//call the function first time to start the image appearing
slideit()
```


That's it

Now go play with it!

I'll put the code in
github.com/willknott

and

github.com/CoderDojo/CoderDojo-Kata

later