



Announcements

Friday is Valentines Day



Animation

JavaScript can be used for animating images on a web page



The Plan

An animation is the rapid display of a series of still images ... like cartoons

There are three steps to animate

- Place first still image(s) on web page
- Prefech the series of images and store them
- Setup a timer to cycle through the images
- **new0.gif, new1.gif, new2.gif, new3.gif**

Smooth motion requires 30 times/sec display



Creating GIFs

GIF files for animation are progressively different ... make them w/Photoshop

- The series should all have the same size
- Begin with an initial GIF and build all others from it
- Getting the motion to be smooth may take a bit of fiddling

Animated GIFs -- GIFs that automatically cycle use a special format and software



Place Still Image(s)

Placing the image uses a standard `` tag

```

<html><head><title>Test Page</title></head>
<body>
  
  <script language="JavaScript"> Code here
</script>
</body>
</html>

```



The document.images

When HTML draws a page, the images go in an array: **document.images**

- Recall, arrays are names w/ indexes, like `A[1]`
- Each element of `document.images` array holds one image
- Pictures are put into `document.images` in the order the encountered on the page ... so for Test Page, `document.images[0]` ⇔ `new0.gif`
- Changing the `.src` property of the array changes the picture

But the images must be prefetched first



Prefetch Images I

"Prefetch" means to get the images and save them in (our own) array so they are handy to assign to doc.i

- We must declare an array (and probably an index variable, too):

```
var i, prefetch = new Array(4);
```

- Then we set it up to hold images:

```
for (i=0; i<4; i++) {
  prefetch[i] = new Image;
}
```



Prefetch Images II

Once the array is declared and setup, get the images and assign them to the .src field of the array:

```
for (i=0; i<4; i++) {
  prefetch[i].src = "new" + i + ".gif";
}
```

- Notice that the names of the images, **new0.gif, new1.gif, new2.gif, new3.gif** are "constructed" using the index variable



Test It

```

<body>


<script language="JavaScript">
var i, prefetch = new Array(4);
for (i=0; i<4; i++) {
  prefetch[i] = new Image;
}
for (i=0; i<4; i++) {
  prefetch[i].src="new"+i+".gif";
}
document.images[0].src=prefetch[1].src;
</script>
</body>

```

Place two "0" pix then change the first to "1"

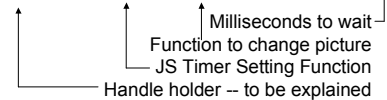


Change Image

Once Web page is drawn, nothing happens unless you cause an event

- To animate a series of stills you must cause the computer to "wake-up" and change to the next image 30 times a second
- Set a timer to cause the wake-up

```
timerID=setTimeout("animate()", 30);
```



Animate Function

animate() must advance the frame counter, update the image and schedule the next timer ...

```

var frame=0, timerID;
function animate(){
  frame=(frame+1)%4; //advance
  document.images[0].src
    = prefetch[frame].src; //update
  timerID=setTimeout("animate()", 30);
}

```



Watch It Go

```



<script language="JavaScript">
var i, prefetch = new Array(4);
var frame=0, timerID;
for (i=0; i<4; i++){
  prefetch[i] = new Image;
}
for (i=0; i<4; i++){
  prefetch[i].src="new" + i + ".gif";
}
timerID=setTimeout("animate()", 5000);
function animate(){
  frame = (frame+1)%4;
  document.images[0].src=prefetch[frame].src;
  timerID = setTimeout("animate()", 30);
}
</script>

```

Start action



Changes ...

Suppose we want "new" to revolve once ever 5 seconds ...

- `animate()` sets timer for two different times
 - When animating, 30 ms
 - When waiting, 5000
 - Use an `if`-statement
- ```
if (frame == 0)
 setTimeout("animate()", 5000);
else
 setTimeout("animate()", 30);
```



## Watch It Go

```
var i, pref = new Array(4);
var frame=1, timerID;
for (i=0; i<4; i++){
 pref[i] = new Image;
}
for (i=0; i<4; i++){
 pref[i].src="new" + i + ".gif";
}
timerID=setTimeout("animate()", 5000);
function animate() {
 frame = (frame+1)%4;
 document.images[0].src=pref[frame].src;
 if (frame == 0)
 timeID = setTimeout("animate()", 5000);
 else
 timerID = setTimeout("animate()", 30);
}
```



## Chapter 21

Chapter 21 illustrates solving a large problem -- an animated page

- The main topic is how to decompose a large problem into pieces and reassemble
- Project 2 is a large project (that was divided for you to be an assignment)
- When you have time -- end of term? -- try your creating your own animations



## Summary

Animation requires a 3 step process

- Place the initial image(s)
- Prefetch the series of images that will be the animation
- Setup the animation function to draw the next item in the series

When creating your own GIFs make sure that the sizes are all the same