SetTimeout():

Slide Show:

So far the user has controlled the action. However, you can make a slide show that automatically cycles through each image in an array and goes to the next image at an incremental time. You do this by adding a call to a "setTimeout()" function (already written by Javascript) that basically does nothing for a set amount of time, then calls your first function over again. So you've already written a JavaScript that cycles through each image in an array. The following code should look familiar:

```
<script type = "text/javascript">
           imgArray = new Array()
           imgArray [0] = "kittyfur-back.jpg"
           imgArray [1] = "kittyhimself.jpg"
           imgArray [2] = "kittybreakfast.jpg"
           imgArray [3] = "kittyno-regrets.jpg"
           imgArray [4] = "kttylemon.jpg"
           imgArray [5] = "kittypanda.jpg"

           arraycounter = 0

           function ChangePic()
          {
               document.images["kittypic1"].src = imgArray[arraycounter]
               arraycounter = arraycounter + 1
               if (arraycounter == imgArray.length)
               {
                      arraycounter = 0
               }
          }
 </script>
```

First, we'll modify this by adding a call to a function setTimeout().

What the setTimeout() function does:

setTimeout() is a function that in essence causes the JavaScript code to pause working. No matter what the JavaScript code is doing, when it hits the line that calls setTimeout(), it pauses. How long does it pause? As long as we tell it to. One of the parameters we send into the setTimeout() function is the number of milliseconds we want the javascript to pause for. So the call looks like this:

```
      setTimeout("Function1()",5000)
```

the pause will be for 5000 milliseconds. Then after the pause, setTimeout() will call the function "Function1()". Here is an example of using setTimeOut to change the color of a button back to black after 2000 msec. Try it:

```
<script type="text/javascript">

function setToRed ( )
{
  document.getElementById("colorButton").style.color = "#FF0000";
  setTimeout ( "setToBlack()", 2000 );
}
```

```
function setToBlack ( )
{
  document.getElementById("colorButton").style.color = "#000000";
}

</script>

<input type="button" name="clickMe" id="colorButton" value="Click me and
wait!" onclick="setToRed()"/>
```

If this is working correctly, the button should turn red, and then after 2000 msec it should turn back to black.

____
_

For our slide show, I'll make the timeout last for 3000 milliseconds, and when it's over, I want to call the ChangePic() function so that the next picture comes up. To do that, at the end of the ChangePic function (from page 1), I'll add:

```
      setTimeout(“ChangePic()”,3000)
```

So my function now looks like this:

```
    function ChangePic()
    {
       document.images[“kittypic1”].src = imgArray[arraycounter]
       arraycounter = arraycounter + 1
       if (arraycounter == imgArray.length)
       {
          arraycounter = 0
       }
       setTimeout(“ChangePic()”,3000)
     }
```

Now what happens is whenever the ChangePic() function is called, it sets the image with the id "kittypic1"'s source to whatever is in the imgArray[arraycounter]. I then add one to arraycounter and check to see if the arraycounter is equal to the length of the imgArray. If it is, I reset the arraycounter back to 0. And now I pause the javascript for 3000 msec. When 3000 msec is up, the function ChangePic() is started over from the beginning. Make sense?

The only thing left is to add an image in my web page with the id of "kittypic1". I could make the slideshow start when I clicked on the image associated with the id "kittypic1" (you know how to do this, right?), but I think in this case I would like the slideshow to start as soon as the web page loads. So I am going to add to the <body> tag an event handler for onload. It will look like this:

```
<body onload = “ChangePic()”>
<img src = “kittyjumping.jpg” width = “500” height = “250” alt = “a pic of a
kitty jumping” id = “kittpic1” />
</body>
```

Now when my web page loads into the browser, the function ChangePic() will be called and the slideshow will start.

Your Turn: Make a slide show. Put together the JavaScript at the top, with an array, a counter, and a change image function, then add the setTimeout() call. Make sure there's an image in your web page with the correct id, and then add the onload event to the body.

Text for slideshow:

You may wish to cycle through text as you cycle through your slideshow (think of vacation pictures, each with their own text associated with them).

To do this, you will need to create an array of text. The array will need to be the exact same length as the array of images if you want them to synch up. If you don't, it doesn't matter how long the text array is. For now, I'll make them be the same length:

```
textArray = new Array()
textArray[0] = "picture of grumpy cat and a big pile of fur"
textArray[1] = "baby kitten wanting to hold bottle of milk"
textArray[2] = "cat looking at clock and demanding breakfast"
textArray[3] = "kitty has gone through a whole roll of toilet paper"
textArray[4] = "kitty with a very sour look on his face because he ate a lemon"
textArray[5] = "Panda fell head-first off a slide"
```

Then inside the ChangePic() function, you need to add the code that changes the innerHTML of the id associated with the text you want to change. Thus if, in your code you've got

```
<p id = "piclabels">Text associated with pictures </p>
```

You'll add to your ChangePic() function:

```
function ChangePic()
{
   document.images["kittypic1"].src = imgArray[arraycounter]
   document.getElementById("piclabels").innerHTML = textArray[arraycounter]
   arraycounter = arraycounter + 1
   if (arraycounter == imgArray.length)
   {
       arraycounter = 0
   }
   setTimeout("ChangePic()",3000)
}
```

That should do it. Now when you load your page, both the picture and the text below it should cycle through

Your turn: Make sure this works for you. Get a page with an image and text rotating through a slide show when the web page loads into the browser.

Question: What happens if we add another picture to the end of our image array, but don't add anything to the textArray?