

University of Washington, CSE 190 M, Spring 2009

Lab 5: JavaScript for Interactive User Interfaces (Thursday, April 30, 2009)

original lab idea and code by Victoria Kirst and Kevin Wallace; revised by Brian Le and Marty Stepp

The purpose of this lab is to practice using basic JavaScript and UI controls to create interactive web pages. **You very likely won't finish all of the exercises. Just finish as much as you can within the allotted time. You do not need to work on this lab any more after you leave your lab session, though you may if you like.**

Lab 5: Pimp My Text!

The theme of this lab is that we'll be writing a page where the user can type some text into a box, and by clicking on various UI controls, we'll allow the user to "pimp out" the text by giving it some funny styling. You are given a skeleton HTML file named [pimpmytext.html](#) (right-click and select "Save Link As...") that contains a basic HTML shell and page header. This skeleton already links to a CSS file named `pimpmytext.css` that defines all the simple styles you should need for this lab. You do not have to edit this provided stylesheet or write any CSS code today. The skeleton also links to a JavaScript library named Prototype that we'll use in this course, which gives us the ability to use the `$()` function as shown in class. Please don't remove the links to the provided CSS or Prototype JS files from the page.

You will write a JavaScript file called `pimpmytext.js` that will manipulate text in various ways throughout the following exercises.

Lab 5 Resources

- [Lecture Slides](#)
- *Web Programming Step by Step* textbook Ch. 7
- [Install Firebug](#)
- [JSLint](#) JavaScript code checker
- [Runnable solution](#) to this lab (so you can see how your page is supposed to look and work... Don't look at the source code until after lab,

please!!)

Exercises for Today:

1. [Create UI Elements](#)
2. [Bigger Pimpin' Button](#)
3. [Bling Checkbox](#)
4. [Snoopify](#)
5. [Even More Gangsta](#)

Exercise 1: Create UI Elements (roughly 15 minutes)

The first task is to expand `pimpmytext.html` by adding UI controls. Add HTML code for the following:

- Add a field for users to enter *large* (multi-line) amounts of text. It should be 4 rows by 30 columns in size. Wrap it in a bordered field set box with the label "Text".
- Add a second bordered field set box labeled "Pimp It" that contains the following controls:
 - A button labeled "Bigger Pimpin'!"
 - A checkbox labeled "Bling"

You should roughly match the output below (between, but not including, the thick black lines). (Don't worry too much about getting the exact output; the important thing is to have the proper UI elements!)

Pimp My Text



Text

Pimpification

Bigger Pimpin!

☐ Bling

Exercise 2: Bigger Pimpin' Button (roughly 10 minutes)

Now you're going to create a basic JavaScript file so that when the user clicks "Bigger Pimpin!", the text in the main text area will get larger. Do the following:

1. Create a new file and saving it as `pimpmytext.js`. Link your XHTML page to this script file. You may want to make sure that this is working by simply putting an `alert` in your `.js` file and making sure that the alert pops up when you refresh the page. For example, the following could be the entire initial contents of the file:

```
alert("Hello World!");
```

2. Add JavaScript code (and any necessary modifications to the XHTML) so that when the user clicks the "Bigger Pimpin'!" button, the size of the text in the main text area changes to **24pt**.

Hint: Remember that most CSS properties translate directly into properties of the `.style` property within that element's DOM object. For example, the following CSS `color` declaration:

```
#id {  
    color: red;  
}
```

...translates into:

```
$("#id").style.color = "red";
```

For properties that have a hyphen in them, such as `background-color`, the hyphens are removed and subsequent words are capitalized:

```
$("#id").style.backgroundColor = "red";
```

Don't forget to set the property to a string with proper `pt` units, not an integer. If your code doesn't work, look for red Firebug error messages on the bottom-right in your browser, or try running [JSLint](#).

Clicking the button should cause an appearance like the one below.

Pimp My Text



Text

Don't hate the playa.
Hate the game.

Pimpification

Bigger Pimpin!

☐ Bling

Exercise 3: Bling Checkbox (roughly 15 minutes)

You are now going to add an event handler so that when the user checks the "Bling" checkbox, the main text area will have some styles applied to it.

1. Add JavaScript code (and any necessary modifications to the XHTML) so that when the user checks the box, the text in the text area becomes bold. You can test whether a given checkbox or radio button is checked by examining the `checked` property of the box's DOM object. When the box is unchecked, the style should go back to normal.
2. Once you get the bold aspect to work, add the following additional effects to also take place when the Bling checkbox is checked:
 - underline the text (this is the CSS `text-decoration` property)
 - change its color to **green**
 - make it blink (this is *also* the CSS `text-decoration` property)
 - any two other styles of your choice

The best design for this feature would be to place the above style rules into a class in your .css file and apply that class to the element when the box is checked. After checking the box, your text should look something like this:

Pimp My Text



Text

Don't hate the playa.
Hate the game.

Pimpification

Bigger Pimpin!

☒ Bling

Exercise 4: Snoopify (roughly 10 minutes)

Now we will transforming or "Snoopifying" the actual content of the text.

1. Write a new button named Snoopify that, when clicked, modifies the text in the text area by capitalizing it and adding an exclamation point to the end of it. You will want to use the `value` property of the text area's DOM element.
2. Modify your Snoopify button so that it also adds a suffix of "-izzle" to the last word of each sentence. (A sentence being a string of text

that ends in a period character, ".") Do this using the String/array methods `split` and `join`. For example, if you wanted to change all spaces with underscores in a string, you could write:

```
var str = "How are you?"
var parts = str.split(" "); // ["How", "are", "you?"]
str = parts.join("_");      // "How_are_you?"
```

After finishing this exercise and clicking the button, your text should look something like this:



Pimp My Text



Text

DON'T HATE THE PLAYA-izzle.
HATE THE GAME-izzle.

Pimpification

Bigger Pimpin!

☐ Bling

Snoopify

Exercise 5 (for uber-l33t h4x0rZ only): Even More Gangsta

If you finish all of the above, add the following to your page:

1. When the "Bling" checkbox is checked, also set your overall page to have a background image of:
<http://www.cs.washington.edu/education/courses/190m/09sp/labs/5-pimpify/hundred-dollar-bill.jpg>
2. Make it so that when the "Bigger Pimpin'!" button is clicked, rather than hard-setting the font size to 24pt, you'll make it 2pt larger than its current size. You may want to use the `parseInt` function to help you do this.
3. Set a timer so that when the Bigger Pimpin' button is clicked, instead of just increasing the font size once, it will repeatedly increase the font size by 2pt every 500ms.
4. (really tricky) Add a "Malkovitch" button that causes words of five characters or greater in length in your main text area to be replaced with the word "Malkovich". Consider compound words (i.e., contractions or hyphenated terms) to be separate words. (Inspired by the [Malkovitch Mediator](#) by Ka-Ping Yee, inspired by the film [Being John Malkovitch](#))
5. Add an "Igpay Atinlay" button that converts the text to Pig Latin. The rules of "Pig Latin" are:
 1. Words beginning in a consonant (or consonant cluster) have that consonant (or consonant cluster) moved to the end and -ay tacked on following.
 2. Words beginning in a vowel simply have -ay tacked on the end.
6. Add any other crazy styling or pimpin' you want to the page. Show us your best stuff, playa!

