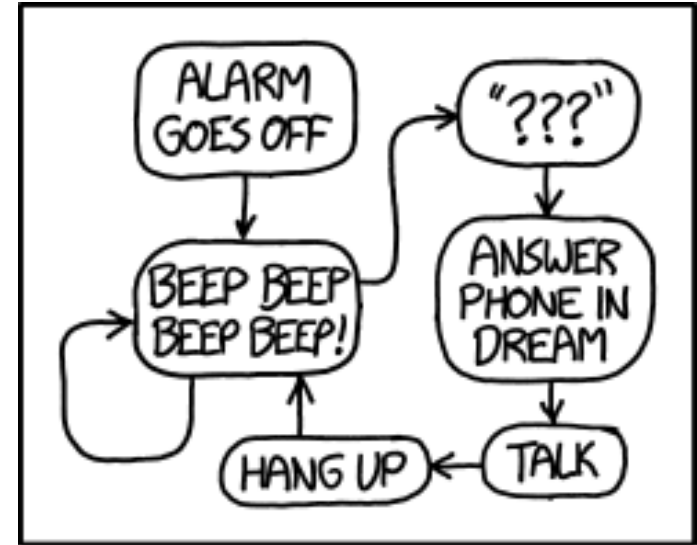


CSc 337

LECTURE 8: EVENTS AND TIMERS



MY PROBLEM WITH PHONE ALARMS

Exercise: tip calculator

Create a page that allows the user to input a price and a percentage they would like to tip. Your page should show the user the total cost (original price plus tip) when the user clicks a button.

Turn the tip red if the percentage is less than 15.

Hint: use `console.log()` to output variables and help you find bugs in your page

Checkboxes: <input>

yes/no choices that can be checked and unchecked (inline)

```
<input type="checkbox" name="lettuce" /> Lettuce  
<input type="checkbox" name="tomato" checked="checked" /> Tomato  
<input type="checkbox" name="pickles" checked="checked" /> Pickles HTML
```

Lettuce Tomato Pickles

output

- none, 1, or many checkboxes can be checked at same time
- when sent to server, any checked boxes will be sent with value on:
 - <http://webster.cs.washington.edu/params.php?tomato=on&pickles=on>
- use checked="checked" attribute in HTML to initially check the box

Radio buttons: <input>

sets of mutually exclusive choices (inline)

```
<input type="radio" name="cc" value="visa" checked="checked" /> Visa  
<input type="radio" name="cc" value="mastercard" /> MasterCard  
<input type="radio" name="cc" value="amex" /> American Express
```

HTML

Visa MasterCard American Express

output

- grouped by name attribute (only one can be checked at a time)
- must specify a value for each one or else it will be sent as value on

Text labels: <label>

```
<label><input type="radio" name="cc" value="visa"
checked="checked" /> Visa</label>
```

```
<label><input type="radio" name="cc" value="mastercard" />
MasterCard</label>
```

```
<label><input type="radio" name="cc" value="amex" /> American
Express</label>
```

HTML

Visa MasterCard American Express

output

- associates nearby text with control, so you can click text to activate control
- can be used with checkboxes or radio buttons
- label element can be targeted by CSS style rules

Drop-down list: <select>, <option>

menus of choices that collapse and expand (inline)

```
<select name="favoritecharacter">
  <option>Jerry</option>
  <option>George</option>
  <option selected="selected">Kramer</option>
  <option>Elaine</option>
</select>
```

HTML

Kramer ▾ Submit Query

output

- option element represents each choice
- select optional attributes: disabled, multiple, size
- optional selected attribute sets which one is initially chosen

Grouping input: <fieldset>, <legend>

groups of input fields with optional caption (block)

```
<fieldset>
  <legend>Credit cards:</legend>
  <input type="radio" name="cc" value="visa" checked="checked" /> Visa
  <input type="radio" name="cc" value="mastercard" /> MasterCard
  <input type="radio" name="cc" value="amex" /> American Express
</fieldset>
```

HTML

Credit cards:

Visa MasterCard American Express

Submit Query

output

- fieldset groups related input fields, adds a border; legend supplies a caption

Styling form controls

```
element [attribute="value"] {  
  property : value;  
  property : value;  
  ...  
  property : value;  
}
```

CSS

```
input [type="text"] {  
  background-color: yellow;  
  font-weight: bold;  
}
```

CSS

Borat

output

- attribute selector: matches only elements that have a particular attribute value
- useful for controls because many share the same element (input)

More about form controls

```
<select id="captain">
  <option value="kirk">James T. Kirk</option>
  <option value="picard">Jean-Luc Picard</option>
  <option value="cisco">Benjamin Cisco</option>
</select>
<label> <input id="trekkie" type="checkbox" /> I'm a Trekkie
</label>
```

HTML

James T. Kirk ▾ I'm a Trekkie

output

- when talking to a text box or `select`, you usually want its `value`
- when talking to a checkbox or radio button, you probably want to know if it's `checked` (`true/false`)

Exercise: stop watch

Create a page that allows the user to input an amount of time and, when the user clicks a button, counts down one second at a time.

An "all done" message should be displayed when the time is up.

Setting a timer

method	description
<code>setTimeout(<i>function</i>, <i>delayMS</i>);</code>	arranges to call given function after given delay in ms
<code>setInterval(<i>function</i>, <i>delayMS</i>);</code>	arranges to call function repeatedly every <i>delayMS</i> ms
<code>clearTimeout(<i>timerID</i>);</code> <code>clearInterval(<i>timerID</i>);</code>	stops the given timer

- both `setTimeout` and `setInterval` return an ID representing the timer
 - this ID can be passed to `clearTimeout/Interval` later to stop the timer



setTimeout example

```
<button id="clickme">Click me!</button>  
<span id="output"></span>
```

HTML

```
window.onload = function() {  
  document.getElementById("clickme").onclick = delayedMessage;  
};  
  
function delayedMessage() {  
  document.getElementById("output").innerHTML = "Wait for it...";  
  setTimeout(sayBooyah, 5000);  
}  
  
function sayBooyah() { // called when the timer goes off  
  document.getElementById("output").innerHTML = "BOOYAH!";  
}
```

JS

Click me!

output

setInterval example

```
var timer = null; // stores ID of interval timer

function delayMsg2() {
  if (timer === null) {
    timer = setInterval(rudy, 1000);
  } else {
    clearInterval(timer);
    timer = null;
  }
}

function rudy() { // called each time the timer goes off
  document.getElementById("output").innerHTML += " Rudy!";
}
```

JS

Click me!

output

Passing parameters to timers

```
function delayedMultiply() {  
    // 6 and 7 are passed to multiply when timer goes off  
    setTimeout(multiply, 2000, 6, 7);  
}  
function multiply(a, b) {  
    alert(a * b);  
}
```

JS

Click me!

output

- any parameters after the delay are eventually passed to the timer function
 - doesn't work in IE; must create an intermediate function to pass the parameters
- why not just write this?

```
setTimeout(multiply(6 * 7), 2000);
```

JS

Common timer errors

- many students mistakenly write `()` when passing the function

```
setTimeout(booyah(), 2000);  
setTimeout(booyah, 2000);
```

```
setTimeout(multiply(num1 * num2), 2000);  
setTimeout(multiply, 2000, num1, num2);
```

JS

- what does it actually do if you have the `()` ?
 - it calls the function immediately, rather than waiting the 2000ms!

The innerHTML property

```
<button onclick="addText();" >Click me!</button>  
<span id="output">Hello </span>
```

HTML

```
function addText() {  
  var span = document.getElementById("output");  
  span.innerHTML += " bro";  
}
```

JS

Click me! Hello

output

- can change the text inside most elements by setting the `innerHTML` property

Abuse of innerHTML

```
// bad style!  
var paragraph = document.getElementById("welcome");  
paragraph.innerHTML =  
    "<p>text and <a href=\"page.html\">link</a>";
```



JS

- `innerHTML` can inject arbitrary HTML content into the page
- however, this is prone to bugs and errors and is considered poor style
- we forbid using `innerHTML` to inject HTML tags; inject plain text only
 - (later, we'll see a better way to inject content with HTML tags in it)