Event Driven Programming

Traditional Programming:
- Start at beginning
- Execute statements until the program ends

Event Driven Programming:
- Start at beginning
- Draw page and setup Graphical Controls
- Attach “Event Handlers” (functions) to the controls
- Sit back and wait...
- And wait...
- If an Event is triggered, run the handler
- Now go back to waiting for another event...

Recall: Example of “Event”

Mouse clicks can trigger function calls:

<p onclick="...">Click this</p>

Example: SimpleClickEvent.html
Mouse Events

onclick – user clicks element
ondblclick – user double-clicks element
onmousedown – mouse button has been pressed (but not released)
onmouseup – mouse button has been released
onmouseover – mouse cursor passes over element
onmousemove – mouse cursor has moved while over element
onmouseout – mouse cursor has been moved off element

Example: MouseEvents.html

Page Loading/Unloading

- onload – triggered when page is loaded
- onunload – triggered when page is unloaded
  (Some browsers may disable this.)

Example: LoadEvents.html
Events we will use with “Forms”

onkeypress – key has been pressed and released  
onkeydown – key is held down  
onkeyup – key is released  

onchange – the state of a “control” has been changed (like a checkbox or radio button)

Dynamic Updates to Elements

We can modify an existing element, as long as it has an identifier.

Assume the following element already appears on the page:

```
<img id="Joe" src="flower.jpg" alt="flower" height="50" />
```

We can use Javascript to modify ANY attribute of the element dynamically!
Dynamic Updates to Elements

```html
<img id="Joe" src="flower.jpg" alt="flower" height="50" />
```

Old method:
```javascript
var x = document.getElementById("Joe");
x.src="dog.jpg";
x.height="100";
```

New method*:
```javascript
Joe.src="dog.jpg";
Joe.height="100";
```

*May not work on all browsers (yet)!

Example: DynamicUpdates1.html

---

Dynamic Updates to Style

```html
<p id="Bozo">Hello!</p>
```

In Javascript, we can change the style at will:
```javascript
Bozo.style.color = "green";
Bozo.style.fontFamily = "arial";
Bozo.style.textAlign = "center";
```

Caution: Property names don’t always match with CSS names!
For example: fontFamily vs. font-family

Example: DynamicStyleUpdates.html
Dynamic Updates to “Inner HTML”

<p id="Bozo">Hello!</p>

In Javascript, we can change what’s between the tags at will:

Bozo.innerHTML = "Goodbye!";

Example: DynamicInnerHTMLUpdates.html
BetterRouletteWheel.html