

# Client Side Scripting } *Events*

# Events

- { Events
- { Events and ID's
- { Passing Arguments

# Events



# What is an event?

Notification that a condition is true.

# Events can be...

- { User based
- { Logic based
- { Functionality based



# We will focus on HTML DOM Events

# List of Built in - DOM Events

- { onblur
- { onchange
- { onclick
- { ondblclick
- { onerror
- { onfocus
- { onkeydown
- { onkeypress
- { onkeyup
- { onmousedown
- { onmousemove
- { onmouseout
- { onmouseover
- { onmouseup
- { onresize
- { onselect
- { onunload
- { onload

# List of Built in - DOM Events

## CLICK BASED

- { onblur
- { onchange
- { **onclick**
- { **ondblclick**
- { onerror
- { onfocus
- { onkeydown
- { onkeypress
- { onkeyup
- { **onmousedown**
- { onmousemove
- { onmouseout
- { onmouseover
- { **onmouseup**
- { onresize
- { onselect
- { onunload
- { onload

# List of Built in - DOM Events

## MOUSE BASED

- { onblur
- { onchange
- { onclick
- { ondblclick
- { onerror
- { onfocus
- { onkeydown
- { onkeypress
- { onkeyup
- { onmousedown
- { onmousemove
- { onmouseout
- { onmouseover
- { onmouseup
- { onresize
- { onselect
- { onunload
- { onload

# List of Built in - DOM Events

## KEYBOARD BASED

- { onblur
- { onchange
- { onclick
- { ondblclick
- { onerror
- { onfocus
- { onkeydown
- { onkeypress
- { onkeyup
- { onmousedown
- { onmousemove
- { onmouseout
- { onmouseover
- { onmouseup
- { onresize
- { onselect
- { onunload
- { onload

# List of Built in - DOM Events

## LOAD BASED

- { onblur
- { onchange
- { onclick
- { ondblclick
- { **onerror**
- { onfocus
- { onkeydown
- { onkeypress
- { onkeyup
- { onmousedown
- { onmousemove
- { onmouseout
- { onmouseover
- { onmouseup
- { onresize
- { onselect
- { **onunload**
- { **onload**

# List of Built in - DOM Events

## MISC

- { onblur
- { onchange
- { onclick
- { ondblclick
- { onerror
- { onfocus
- { onkeydown
- { onkeypress
- { onkeyup
- { onmousedown
- { onmousemove
- { onmouseout
- { onmouseover
- { onmouseup
- { onresize
- { onselect
- { onunload
- { onload

# Event Process

- { Register or listen for an event.
- { Define a function, aka **event handler**, to run when the condition of the event has been met.

# Create an Event

**onload**

```
// listen for when the page has  
// loaded
```

# Create an Event

```
onload = init;
```

# Create an Event

```
onload = init;  
// When the page has loaded run the  
// function init.
```

# Create an Event

```
onload = init;  
// What is init?
```

# Create an Event

```
onload = init;  
// It's a function defined by us.  
  
function init(){  
    alert('Hello World');  
}
```

# Create an Event - Final Code

```
// Event Listener  
onload = init;  
  
// Event Handler  
function init(){  
    alert( 'Hello World' );  
}
```

# Events and ID's



# A Little Tricky

Registering events tied to id's within your HTML page can be a little tricky.

# Events for Id's

```
document.getElementById( 'btn' )
```

```
// an id of an element in the HTML document
```

# Events for Id's

```
document.getElementById('btn').onclick
```

```
// a click event has been added
```

```
// javascript runtime is listening for any clicks
```

```
// on the id btn
```

# Events for Id's

```
document.getElementById('btn').onclick = changeStyle;  
  
// when the click on the btn has happend  
// run the function changeStyle
```

# Events for Id's

```
document.getElementById('btn').onclick = changeStyle;  
  
// what is changeStyle?
```

# Events for Id's

```
document.getElementById('btn').onclick = changeStyle;

// changeStyle is a function defined by us that will
// alter the font of the btn to red when it's clicked

function changeStyle(){
    document.getElementById(this.id).style.color="#F00000";
}
```

# Events for Id's

```
document.getElementById('btn').onclick = changeStyle;

// changeStyle is a function defined below that will
// alter the style of the btn to red when it's clicked

function changeStyle()
{
    document.getElementById(this).style.color="#F00000";
}
```

# Events for Id's

```
document.getElementById('btn').onclick = changeStyle;

// changeStyle is a function defined below that will
// alter the style of the btn to red when it's clicked

function changeStyle() {
    document.getElementById(this).style.color="#F00000";
}
```

**HUGE PROBLEM**

# Events for Id's

```
document.getElementById('btn').onclick = changeStyle;  
  
// changeStyle is a function defined by us that will  
// alter the font of the btn to red when it's clicked  
  
function changeStyle(){  
    document.getElementById(this.id).style.color="#F00000";  
}
```

**btn is null**

# Events for Id's

```
document.getElementById('btn').onclick = changeStyle;  
  
// changeStyle is a function defined by us that will  
// alter the font of the btn to red when it's clicked  
  
function changeStyle(){  
    document.getElementById(this.id).style.color="#F00000";  
}
```

**Why?**

# Events for Id's

```
document.getElementById('btn').onclick = changeStyle;  
  
// changeStyle is a function defined by us that will  
// alter the font of the btn to red when it's clicked  
  
function changeStyle(){  
    document.getElementById(this.id).style.color="#F00000";  
}
```

**btn has not loaded**

# Events for Id's

```
document.getElementById('btn').onclick = changeStyle;

// changeStyle is a function defined by us that will
// alter the font of the btn to red when it's clicked

function changeStyle(){
    document.getElementById(this.id).style.color="#F00000";
}
```

**Really? How can that be?**

# Events for Id's

```
<head>  
<script type="text/javacript" src="js/script.js"></script>  
</head>  
  
<body>  
<div id="btn">button</div>  
</body>
```

# Events for Id's

```
<head>  
<script type="text/javascript" src="js/script.js"></script>  
</head>  
  
<body>  
<div id="btn">button</div>  
</body>
```

**The javascript loads here**

# Events for Id's

```
<head>  
<script type="text/javacript" src="js/script.js"></script>  
</head>  
  
<body>  
<div id="btn">button</div>  
</body>
```

**But the div doesn't exist until here.**

# Events for Id's

```
<head>  
<script type="text/javacript" src="js/script.js"></script>  
</head>  
  
<body>  
<div id="btn">button</div>  
</body>
```

**How can you fix this?**

# Events for Id's

```
document.getElementById('btn').onclick = changeStyle;

function changeStyle(){
    document.getElementById(this.id).style.color="#F00000";
}
```

**Wrap this listener in an onload statement**

# Events for Id's

```
onload = init;
```

```
function init(){  
    document.getElementById('btn').onclick = changeStyle;  
}
```

```
function changeStyle(){  
    document.getElementById(this.id).style.color="#F00000";  
}
```

**Wrap this listener in an onload statement**

# Events for Id's

```
onload = init;
```

```
function init(){  
    document.getElementById('btn').onclick = changeStyle;  
}
```

```
function changeStyle(){  
    document.getElementById(this.id).style.color="#F00000";  
}
```

**Now the btn will only be registered when the entire page loads.**

# Events for Id's

```
onload = init;

function init(){
    document.getElementById('btn').onclick = changeStyle;
}

function changeStyle(){
    document.getElementById(this.id).style.color="#F00000";
}
```

**All is well!**



# Passing Arguments through Events



# Passing Arguments

```
onload = init;

function init(){
    // this is to make it easier to reference the btn
    var btn = document.getElementById( 'btn' );

    // Note the nested function
    btn.onclick = function () {changeStyle( 'bob.jpg' );}
}

function changeStyle(img){
    this.id.src=img;
}
```

# Passing Arguments

```
onload = init;

function init(){
  // this is to make it easier to reference the btn
  var btn = document.getElementById( 'btn' );

  // Note the nested function
  btn.onclick = function () {changeStyle( 'bob.jpg' );}
}

function changeStyle(img){
  this.id.src=img;
}
```

**The unnamed function is wrapped around the changeStyle function**

# Passing Arguments

```
onload = init;

function init(){
  // this is to make it easier to reference the btn
  var btn = document.getElementById( 'btn' );

  // Note the nested function
  btn.onclick = function () {changeStyle( 'bob.jpg' );}
}

function changeStyle(img){
  this.id.src=img;
}
```

**A flexible technique with dynamic referneces.**

# Events

- { Events
- { Events and ID's
- { Passing Arguments