

JavaScript 动手实验

主讲人： 和凌志

Agenda

- ① JavaScript 常用函数
- ② JavaScript 动手实验

Case1 : Event Handler

Why use Event Handler ?

- Event handlers allow your web page to detect when a given “event” has occurred, so that it can run some JavaScript code.
- An example of an event could be when the user clicks on a particular HTML

Case 1 完整代码:

```
<input type="button"  
  onclick="alert(' Hello World');"   
  value="Display Example"  
>
```

when the user clicks on our button, a JavaScript alert box is displayed. This is done by adding an **onClick** attribute and placing the JavaScript alert box code into it.

Placement of the External JavaScript

- For performance reasons, JavaScript is often placed at the bottom of the document, just before the closing `<body>` tag.
- This allows the rest of the page to load first, before making the call to the external file.
- However, not all JavaScript can be placed at the bottom of the document. If another script on the page needs that particular piece of JavaScript at load time, you will need to place the external JavaScript before the script that needs it.

JavaScript functions

```
function displayWelcomeMessage(FirstName) {  
    ...code goes here...  
}
```

```
displayWelcomeMessage('Homer');
```

Case1 : function usage

First name: 您好

Display Message

Case 1 完整代码:

```
<script>
  function displayMessage(firstName) {
    alert("Hello " + firstName );
  }
</script>

<form>
  First name:
  <input type="input" name="yourName">
  <input type="button" onclick="displayMessage(form.yourName.value);"
  value="Display Message">
</form>
```

代码改造： 使用 外部 js 文件 （注意路径）

创建外部 js 文件，注意路径
js/myjavascript.js （js 后缀）



```
JS myjavascript.js ●
1
2 function displayMessage(firstName) {
3     alert("Hello " + firstName );
4 }
```

HTML 引入 外部 js 文件，如下。 调用方式不变。

```
<script src="./js/myjavascript.js"> </script>
```

Case 2 : getElementById() 应用

Show Message 1

Show Message 2

InnerHTML In JavaScript

- When you use innerHTML, you can change the page's content without refreshing the page. This can make your website feel quicker and more responsive to user input.
- The innerHTML property can be used along with getElementById() within your JavaScript code to refer to an HTML element and change its contents.

InnerHTML Syntax:

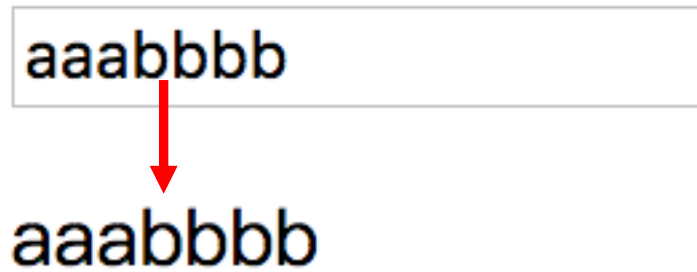
```
document.getElementById('{ID of element}').innerHTML = '{content}';
```

Case2 : 完整代码

```
<script>
    function Msg1(){
        document.getElementById('myText').innerHTML = 'Message 1!';
    }
    function Msg2(){
        document.getElementById('myText').innerHTML = 'Message 2!';
    }
</script>
```

```
<input type="button" onclick="Msg1()" value="Show Message 1">
<input type="button" onclick="Msg2()" value="Show Message 2">
<p id="myText"></p>
```

Case3 : innerHTML With User Input



Case3 : 完整代码

```
<script>
    function showMsg(){
        var userInput = document.getElementById('userInput')
        .value;
        document.getElementById('userMsg').innerHTML =
        userInput;
    }
</script>

<input type="input" maxlength="40" id="userInput"
onkeyup="showMsg()" placeholder="Enter text here...">
<p id="userMsg"></p>
```



感谢聆听