## 对事件做出反应

## 本例从事件处理器调用一个函数：

<!DOCTYPE html>

<html>

<head>

<script>

function changetext(id)

{

id.innerHTML="Ooops!";

}

</script>

</head>

<body>

<h1 onclick="changetext(this)">Click on this text!</h1>

</body>

</html>

## HTML 事件属性

<button onclick="displayDate()">Try it</button>

## 使用 HTML DOM 来分配事件

<script>

document.getElementById("myBtn").onclick=function(){displayDate()};

</script>

## onload 和 onunload 事件

onload 和 onunload 事件会在用户进入或离开页面时被触发。

onload 事件可用于检测访问者的浏览器类型和浏览器版本，并基于这些信息来加载网页的正确版本。

<body onload="checkCookies()">

## onchange 事件

onchange 事件常结合对输入字段的验证来使用。

下面是一个如何使用 onchange 的例子。当用户改变输入字段的内容时，会调用 upperCase() 函数。

<input type="text" id="fname" onchange="upperCase()">

## onmouseover 和 onmouseout 事件

onmouseover 和 onmouseout 事件可用于在用户的鼠标移至 HTML 元素上方或移出元素时触发函数。

## onmousedown、onmouseup 以及 onclick 事件

首先当点击鼠标按钮时，会触发 onmousedown 事件，当释放鼠标按钮时，会触发 onmouseup 事件，最后，当完成鼠标点击时，会触发 onclick 事件。

**Js生成事件和分发**

<script>

let div = document.createElement("div");

document.body.appendChild(div);

let event = document.createEvent("MouseEvent"); // 事件类型

event.initEvent(

"mouseover", // 鼠标事件类型

true // 是否冒泡

)

div.dispatchEvent(event)

</script>