

JavaScript Timing Events

- description : JavaScript Timing Events
- author :
- email : shlim@repia.com
- lastupdate : 2021-04-12

Source of the article

- [JavaScript Timing Events](#)

JavaScript

_____ (Timing Events)

Timing Events

window

JavaScript

가

- setTimeout(function, milliseconds)
 - ()
- setInterval(function, milliseconds)
 - setTimeout()

setTimeout() setInterval() HTML DOM Window

The setTimeout() Method

```

window.setTimeout(function, milliseconds);

```

window.setTimeout() window prefix

. 3 "Hello" window 가 .

Click "Try it". Wati 3 seconds, and the page will alert "Hello".

Try it

How to Stop the Execution?

`clearTimeout()` `setTimeout()` .

```
window.clearTimeout(timeoutVariable)
```

`window.clearTimeout()` window prefix .

`clearTimeout()` `setTimeout()` .

```
myVar = setTimeout(function, milliseconds);  
clearTimeout(myVar);
```

가 , `clearTimeout()` .

, "Stop" 가 .

```
<!DOCTYPE html>  
<html>  
<body>  
  <p>Click "Try it". Wati 3 seconds, and the page will alert "Hello".</p>  
  <p>Click "Stop" to prevent the first function to execute.</p>  
  <p>(You must click "Stop" before the 3 seconds are up.)</p>  
  <button onclick="myVar = setTimeout(myFunction, 3000)">Try it</button>  
  <button onclick="clearTimeout(myVar)">Stop it</button>  
  <script>  
    function myFunction() {  
      alert("Hello");  
    }  
  </script>  
</body>  
</html>
```

The setInterval() Method

setInterval()

```
window.setInterval(function, milliseconds);
```

window.setInterval()

window prefix

1

"myTimer"

```
<!DOCTYPE html>
<html>
<body>
  <p>A script on this page starts this clock:</p>
  <p id="demo"></p>
  <script>
    var myVar = setInterval(myTimer, 1000);
    function myTimer() {
      var d = new Date();
      document.getElementById("demo").innerHTML =
d.toLocaleTimeString();
    }
  </script>
</body>
</html>
```

1

1000

가

How to Stop the Execution?

clearInterval()

setInterval()

```
window.clearInterval(timerVariable)
```

window.clearInterval() window prefix .
clearInterval() setInterval() .

```
myVar = setInterval(function, milliseconds);  
clearInterval(myVar);
```

, "Stop time" 가 .

```
<!DOCTYPE html>  
<html>  
<body>  
  <p>A script on this page starts this clock:</p>  
  <p id="demo"></p>  
  <button onclick="clearInterval(myVar)">Stop time</button>  
  <script>  
    var myVar = setInterval(myTimer, 1000);  
    function myTimer() {  
      var d = new Date();  
      document.getElementById('demo').innerHTML = d.toLocaleTimeString();  
    }  
  </script>  
</body>  
</html>
```

More Examples

Another simple timing

```
<!DOCTYPE html>  
<html>  
<body>  
  <button onclick="timedText()">Try it</button>  
  <p id="demo">Click on "Try it". I will display when two, four, and six  
seconds have passed.</p>  
  <script>  
    function timedText() {  
      setTimeout(myTimeout1, 2000)  
      setTimeout(myTimeout2, 4000)  
      setTimeout(myTimeout3, 6000)  
    }  
    function myTimeout1() {  
      document.getElementById('demo').innerHTML = "2 seconds";  
    }  
  </script>
```

```

    function myTimeout2() {
        document.getElementById('demo').innerHTML = "4 seconds";
    }
    function myTimeout3() {
        document.getElementById('demo').innerHTML = "6 seconds";
    }
</script>
</body>
</html>

```

A clock created with a timing event

```

<!DOCTYPE html>
<html>
<head>
  <script>
    function startTime() {
      var today = new Date();
      var h = today.getHours();
      var m = today.getMinutes();
      var s = today.getSeconds();
      m = checkTime(m);
      s = checkTime(s);
      document.getElementById('txt').innerHTML = h + ":" + m + ":" +
s;

      var t = setTimeout(startTime, 500);
    }
    function checkTime(i) {
      if (i < 10) { i = "0" + i }; // add zero in front of numbers <
10
      return i;
    }
  </script>
</head>
<body onload="startTime()">
  <div id="txt"></div>
</body>
</html>

```

„ Javascript, Timing, Events

From:
<http://rwiki.repia.com/> -

. - 2023.12

Permanent link:
http://rwiki.repia.com/doku.php?id=wiki:javascript:javascript_note:js_timing&rev=1618215159 

Last update: **2022/03/10 19:52**