

# Fungsi dan Event Javascript

Suprayogi,M.Kom

---

- fungsi adalah **blok kode** yang dibuat untuk melakukan tugas tertentu.
- fungsi dieksekusi ketika nama dari fungsi tersebut dipanggil.

## Fungsi

---

- Definisi fungsi
 

```
function name(parameter1, parameter2, parameter3) {
    // code to be executed
}
```
- Pemanggilan fungsi
 

```
name(par1,par2,par3)
```

## Syntax

### lat\_fungsi1.html

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Functions</h2>

<p>This example calls a function which performs a calculation and returns the
result:</p>

<p id="demo"></p>

<script>
var x = myFunction(4, 3);
document.getElementById("demo").innerHTML = x;

function myFunction(a, b) {
    return a * b;
}
</script>

</body>
</html>
```

#### JavaScript Functions

This example calls a function which performs a calculation and returns the result:

12

## Contoh

# pendefinisian fungsi

Menggunakan constructor new

```
<!DOCTYPE html>
<html>
<body>

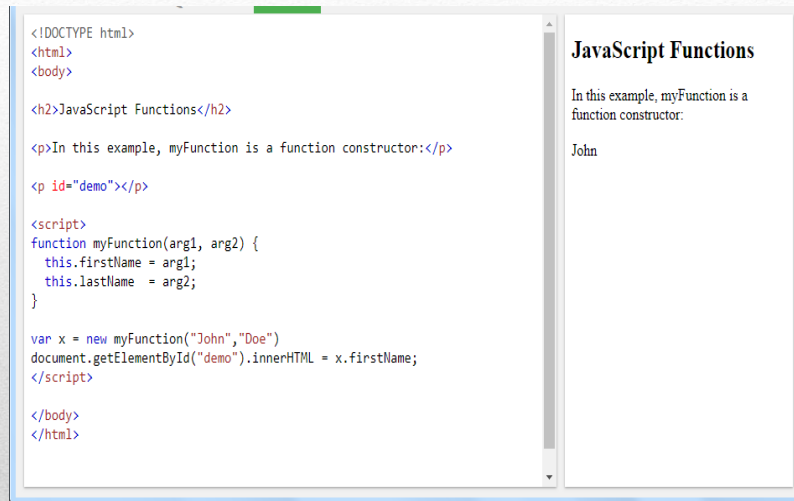
<p>JavaScript has an built-in function constructor.</p>
<p id="demo"></p>

<script>
var myFunction = new Function("a", "b", "return a * b");
document.getElementById("demo").innerHTML =
  myFunction(4, 3);
</script>

</body>
</html>
```

```
JavaScript has an built-in fun
12
```

## Constructor Function



```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Functions</h2>

<p>In this example, myFunction is a function constructor:</p>

<p id="demo"></p>

<script>
function myFunction(arg1, arg2) {
  this.firstName = arg1;
  this.lastName = arg2;
}

var x = new myFunction("John","Doe")
document.getElementById("demo").innerHTML = x.firstName;
</script>

</body>
</html>

```

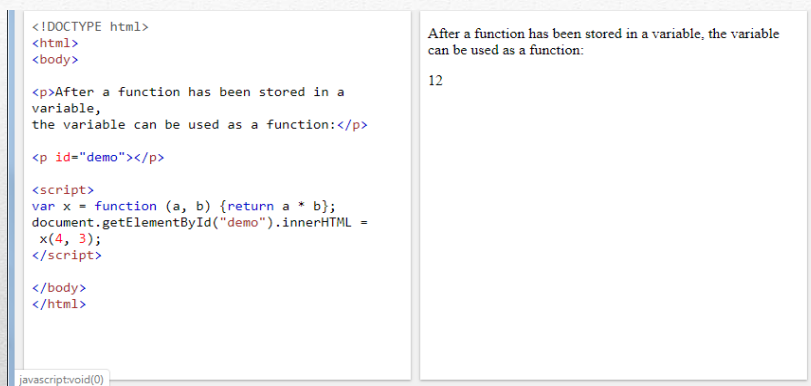
**JavaScript Functions**

In this example, myFunction is a function constructor:

John

## Akses variabel fungsi

lat\_fungsi2.html



```

<!DOCTYPE html>
<html>
<body>

<p>After a function has been stored in a
variable,
the variable can be used as a function:</p>

<p id="demo"></p>

<script>
var x = function (a, b) {return a * b};
document.getElementById("demo").innerHTML =
x(4, 3);
</script>

</body>
</html>

```

After a function has been stored in a variable, the variable can be used as a function:

12

## fungsi anonymous

## lat\_fungsi5.html

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;h2&gt;JavaScript Arrow Functions&lt;/h2&gt;  &lt;p&gt;With arrow functions, you don't have to type the function keyword, the return keyword, and the curly brackets.&lt;/p&gt;  &lt;p&gt;Arrow functions are not supported in IE11 or earlier.&lt;/p&gt;  &lt;p id="demo"&gt;&lt;/p&gt;  &lt;script&gt; const x = (x, y) =&gt; x * y; document.getElementById("demo").innerHTML = x(5, 5); &lt;/script&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<h3>JavaScript Arrow Functions</h3> <p>With arrow functions, you don't have to type the function keyword, the return keyword, and the curly brackets.</p> <p>Arrow functions are not supported in IE11 or earlier.</p> <p>25</p>
---	--

## Fungsi menggunakan =>

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;p&gt;Functions can be used in expressions.&lt;/p&gt; &lt;p id="demo"&gt;&lt;/p&gt;  &lt;script&gt; function myFunction(a, b) {   return a * b; } var x = myFunction(4, 3) * 2; document.getElementById("demo").innerHTML = x; &lt;/script&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<p>Functions can be used in expressions.</p> <p>24</p>
--	--

## Fungsi di dlm ekpresi





The screenshot shows a web browser window with a red header bar. The browser's address bar is empty. The page content is displayed in two parts: a code editor on the left and a rendered output on the right. The code editor shows the following HTML and JavaScript code:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Functions</h2>

<p>myObject.fullName() will return John Doe:</p>

<p id="demo"></p>

<script>
var myObject = {
  firstName: "John",
  lastName: "Doe",
  fullName: function() {
    return this.firstName + " " + this.lastName;
  }
}
document.getElementById("demo").innerHTML = myObject.fullName();
</script>
</body>
</html>
```

The rendered output on the right shows the title "JavaScript Functions" and the text "myObject.fullName() will return John Doe:" followed by the output "John Doe" in a larger font.

## pemanggilan fungsi sbg sebuah method

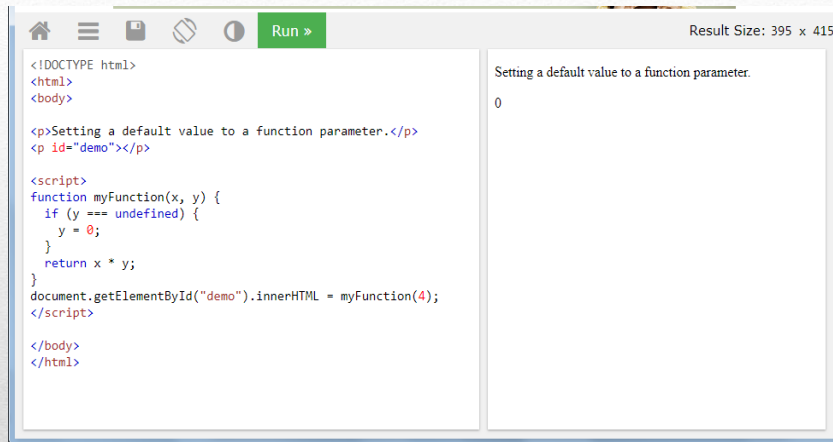
---

Aturan Parameter pd fungsi di JavaScript

- tipe data pada parameter tdk perlu didefinisikan.
- tidak melakukan check tipe data pd argumen yg dilewatkan
- tidak melakukan check thd jml parameter yang diterima

## Parameter fungsi

---



The screenshot shows a web browser window with a red header bar. The browser's address bar shows "Result Size: 395 x 415". The main content area displays the text "Setting a default value to a function parameter." and the number "0". The browser's developer tools are open, showing the following HTML and JavaScript code:

```
<!DOCTYPE html>
<html>
<body>

<p>Setting a default value to a function parameter.</p>
<p id="demo"></p>

<script>
function myFunction(x, y) {
  if (y == undefined) {
    y = 0;
  }
  return x * y;
}
document.getElementById("demo").innerHTML = myFunction(4);
</script>

</body>
</html>
```

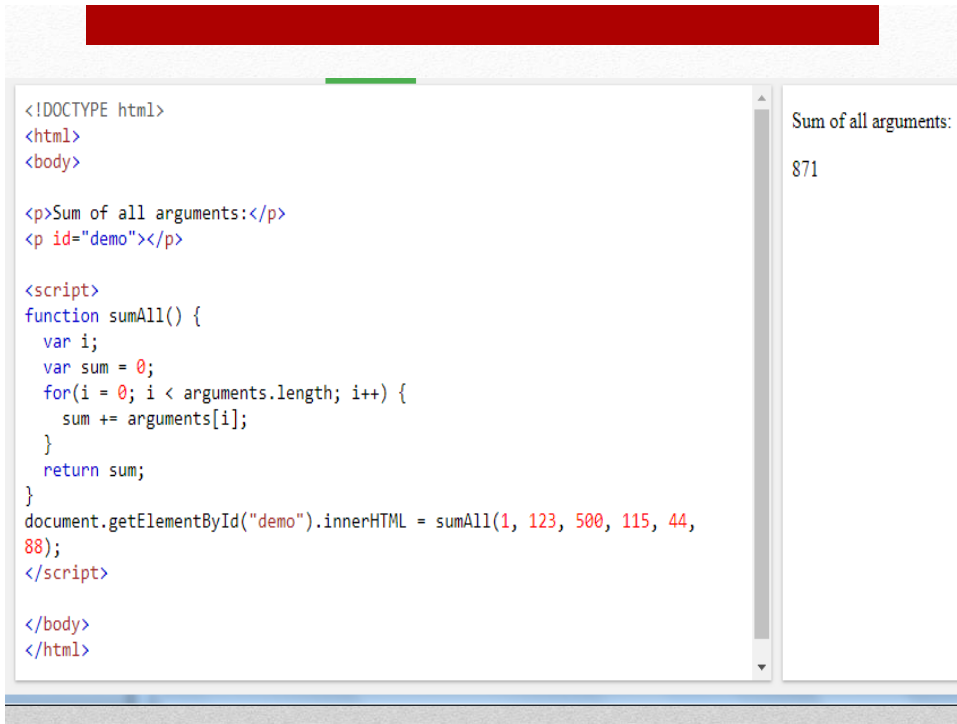
## Parameter default

---

- Fungsi JavaScript memiliki objek built-in yang disebut **object arguments**.
- **object arguments** berisi array argumen yang digunakan ketika fungsi dipanggil.

## objek argumen

---



The screenshot shows a web browser window with a white background. At the top, there is a red horizontal bar. Below it, the browser's address bar is empty. The main content area displays the output of a JavaScript function. On the left, a code editor shows the following HTML and JavaScript code:

```
<!DOCTYPE html>
<html>
<body>

<p>Sum of all arguments:</p>
<p id="demo"></p>

<script>
function sumAll() {
  var i;
  var sum = 0;
  for(i = 0; i < arguments.length; i++) {
    sum += arguments[i];
  }
  return sum;
}
document.getElementById("demo").innerHTML = sumAll(1, 123, 500, 115, 44,
88);
</script>

</body>
</html>
```

On the right side of the browser window, the text "Sum of all arguments:" is displayed above the number "871".



# Event

---



- Event HTML adalah sesuatu hal yang terjadi pada elemen HTML.
- Ketika JavaScript digunakan di halaman HTML, JavaScript dapat "bereaksi" pada event ini.

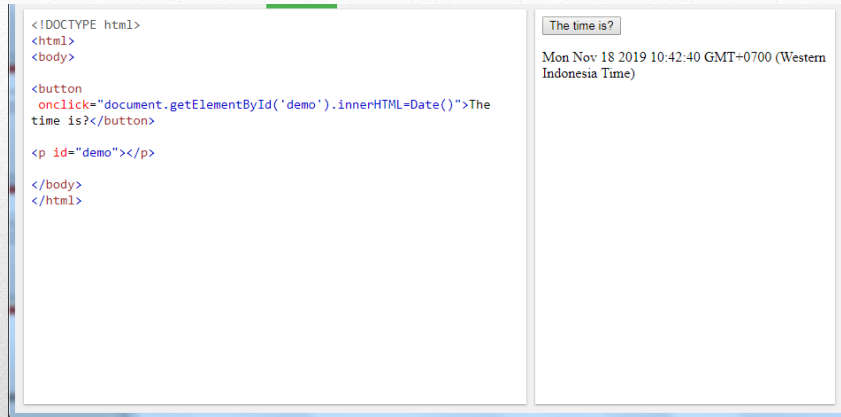
## Event

---

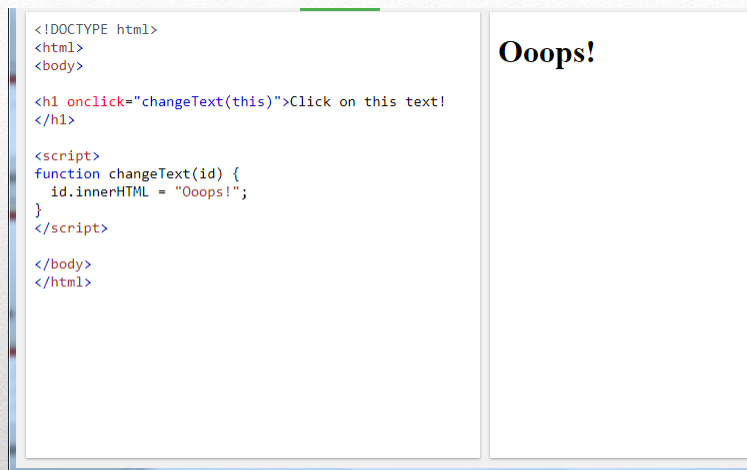
- Event HTML: sesuatu yang dilakukan oleh browser/user.
  - contoh: halaman web selesai loading, text input berubah isinya, button di klik
- atribut **event handler** pd suatu **elemen HTML**:  
`<element event='some JavaScript'>`

## Event

---



## Contoh Event click



## onclick

Event	Description
onchange	An HTML element has been changed
onclick	The user clicks an HTML element
onmouseover	The user moves the mouse over an HTML element
onmouseout	The user moves the mouse away from an HTML element
onkeydown	The user pushes a keyboard key
onload	The browser has finished loading the page

## Event HTML umum

```
<!DOCTYPE html>
<html>
<body onload="loadTeks()">

  <p id="demo"></p>

  <script>
function loadTeks() {
  text="load teks ....."
  document.getElementById("demo").innerHTML = text;
}
</script>

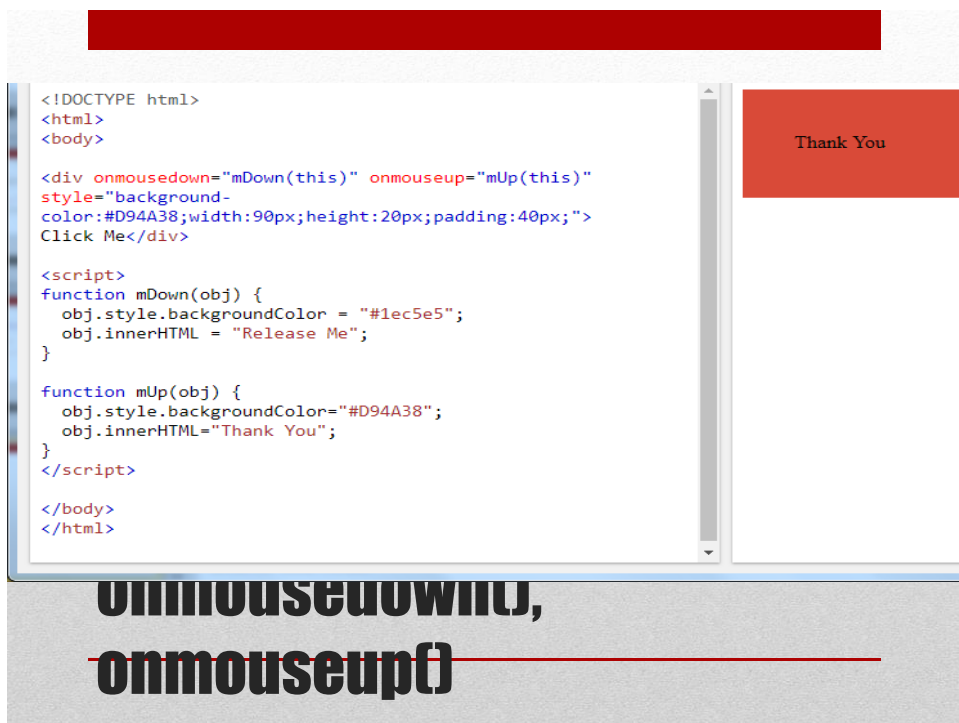
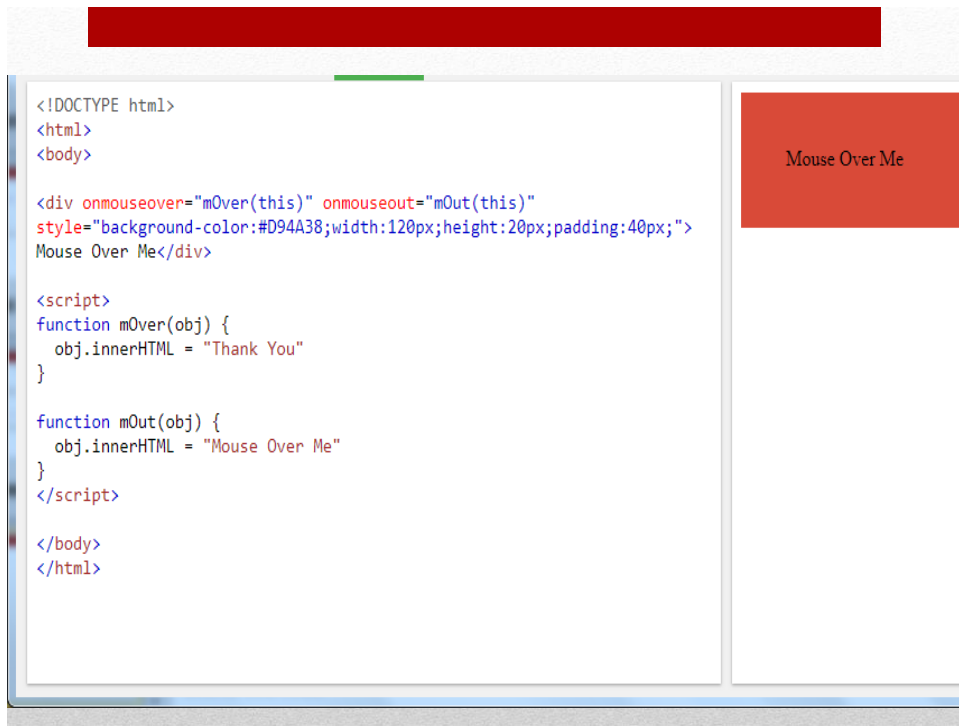
</body>
</html>
```

load teks .....

## onload()

<pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt; &lt;script&gt; function myFunction() {     var x = document.getElementById("fname");     x.value = x.value.toUpperCase(); } &lt;/script&gt; &lt;/head&gt; &lt;body&gt;  Enter your name: &lt;input type="text" id="fname" onchange="myFunction()"&gt;  &lt;p&gt;When you leave the input field, a function is triggered which transforms the input text to upper case.&lt;/p&gt;  &lt;/body&gt; &lt;/html&gt; </pre>	<p>Enter your name: <input type="text" value="HALO"/></p> <p>When you leave the input field, a function is triggered which transforms the input text to upper case.</p>
--	---

<pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;p&gt;This example uses the HTML DOM to assign an "onchange" event to an input element.&lt;/p&gt;  Enter your name: &lt;input type="text" id="fname"&gt;  &lt;p&gt;When you leave the input field, a function is triggered which transforms the input text to upper case.&lt;/p&gt;  &lt;script&gt; document.getElementById("fname").onchange = function() {myFunction()};  function myFunction() {     var x = document.getElementById("fname");     x.value = x.value.toUpperCase(); } &lt;/script&gt;  &lt;/body&gt; &lt;/html&gt; </pre>	<p>This example uses the HTML DOM to assign an "onchange" event to an input element.</p> <p>Enter your name: <input type="text"/></p> <p>When you leave the input field, a function is triggered which transforms the input text to upper case.</p>
--	---





```

<!DOCTYPE html>
<html>
<head>
<script>
function myFunction(x) {
  x.style.background = "yellow";
}
</script>
</head>
<body>

Enter your name: <input type="text"
onfocus="myFunction(this)">

<p>When the input field gets focus, a function is triggered
which changes the background-color.</p>

</body>
</html>

```

Enter your name:

When the input field gets focus, a function is triggered which changes the background-color.

# onfocus()

```

<!DOCTYPE html>
<html>
<body>

<p>Select a new car from the list.</p>

<select id="mySelect" onchange="myFunction()">
  <option value="Audi">Audi</option>
  <option value="BMW">BMW</option>
  <option value="Mercedes">Mercedes</option>
  <option value="Volvo">Volvo</option>
</select>

<p>When you select a new car, a function is triggered which outputs the
value of the selected car.</p>

<p id="demo"></p>

<script>
function myFunction() {
  var x = document.getElementById("mySelect").value;
  document.getElementById("demo").innerHTML = "You selected: " + x;
}
</script>

</body>
</html>

```

Select a new car from the list.

BMW ▼

When you select a new car, a function is triggered which outputs the value of the selected car.

You selected: BMW

# select - onchange()

```

<!DOCTYPE html>
<html>
<body>

<p>Write something in the text field to trigger a function.</p>

<input type="text" id="myInput" oninput="myFunction()">

<p id="demo"></p>

<script>
function myFunction() {
  var x = document.getElementById("myInput").value;
  document.getElementById("demo").innerHTML = "You wrote: " + x;
}
</script>

</body>
</html>

```

Write something in the text field to trigger a function.

You wrote: test

## oninput()

```

<!DOCTYPE html>
<html>
<body>

<p>A function is triggered when the user is pressing a
key in the input field.</p>

<input type="text" onkeydown="myFunction()">

<script>
function myFunction() {
  alert("You pressed a key inside the input field");
}
</script>

</body>
</html>

```

A function is triggered when the user is pressing a key in the input field.

## Onkeydown()