

**TUGAS PELATIHAN CODE JAVASCRIPT
PADA W3SCHOOL**



Disusun oleh:

NAMA : PIONA ROSKHA

NIM : 2000148

KELAS : SIK A

Dosen Pengampu:

Wildan Aprizal Arifin, S.Pd., M.Kom,

**UNIVERSITAS PENDIDIKAN INDONESIA
KAMDA SERANG
2021**

JAVASCRIPT

1. Javascript Comments

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Comments</h2>

<h1 id="myH"></h1>

<p id="myP"></p>

<script>
//document.getElementById("myH").innerHTML = "My First Page";
document.getElementById("myP").innerHTML = "Hai namaku Piona Roskha.";
</script>

<p>Aku berasal dari Bogor // Salam kenal yaa.</p>

</body>
</html>

```

JavaScript Comments

Hai namaku Piona Roskha.

Aku berasal dari Bogor // Salam kenal yaa.

2. Javascript Variables

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Variables</h2>

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

<script>
var price1 = 5;
var price2 = 6;
var price3 = 4;
var total = price1 + price2 + price3;
document.getElementById("demo").innerHTML =
"Totalnya adalah: " + total;
</script>

</body>
</html>

```

JavaScript Variables

Totalnya adalah: 15



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

<h2>JavaScript Variables</h2>

<p>Hai namaku Piona Roskha.</p>
<p>Aku dari UPI kampus Serang.</p>

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

<script>
var pi = 3.14;
var person = "Piona";
var answer = 'Ya sayaa';

document.getElementById("demo").innerHTML =
pi + "<br>" + person + "<br>" + answer;
</script>

</body>
</html>
```

JavaScript Variables

Hai namaku Piona Roskha.

Aku dari UPI kampus Serang.

3.14
Piona
Ya sayaa

3. Javascript Let



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

<h2>Redeclaring a Variable Using var</h2>

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

<script>
var x = 10;
// Here x is 10

{
var x = 3;
// Here x is 2
}

// Here x is 2
document.getElementById("demo").innerHTML = x;
</script>

</body>
</html>
```

Redeclaring a Variable Using var

3



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

<h2>JavaScript Hoisting</h2>

<p>With <b>var</b>, you can use a variable before it is declared:</p>

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

<script>
carName = "Piona";
document.getElementById("demo").innerHTML = carName;
var carName;
</script>

</body>
</html>
```

JavaScript Hoisting

With `var`, you can use a variable before it is declared:

Piona

4. Javascript Aritmethic



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

<h2>JavaScript Arithmetic</h2>
<h3>X + Y</h3>

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

<script>
let x = 50;
let y = 25;
let z = x + y;
document.getElementById("demo").innerHTML = z;
</script>

</body>
</html>
```

JavaScript Arithmetic

X + Y

75

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Arithmetic</h2>
<h3>Bilangan ** Bilangan</h3>

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

<script>
let x = 10;
document.getElementById("demo").innerHTML = x ** 2;
</script>

</body>
</html>

```

JavaScript Arithmetic

Bilangan ** Bilangan

100

5. Javascript Assignment

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Assignments</h2>
<h3>Bilangan %= Bilangan</h3>

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

<script>
let x = 8;
x %= 4;
document.getElementById("demo").innerHTML = x;
</script>

</body>
</html>

```

JavaScript Assignments

Bilangan %= Bilangan

0

6. Javascript Data Types

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Objects</h2>

<p id="pio"></p>

<script>
const person = {
  firstName : "Piona",
  lastName  : "Roskha",
  age       : 19,
  eyeColor  : "brown"
};

document.getElementById("pio").innerHTML =
person.firstName + " berumur " + person.age + " tahun.";
</script>

</body>
</html>

```

JavaScript Objects

Piona berumur 19 tahun.

7. Javascript Functoins

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Functions</h2>

<p>Disini kita belajar mengenai Function javascript:</p>

<p id="pio"></p>

<script>
var x = myFunction(6, 2);
document.getElementById("pio").innerHTML = x;

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

</body>
</html>

```

JavaScript Functions

Disini kita belajar mengenai Function javascript:

12

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Functions</h2>

<p id="pio"></p>

<script>
document.getElementById("pio").innerHTML =
"The temperature is " + toCelsius(77) + " Celsius";

function toCelsius(fahrenheit) {
  return (5/9) * (fahrenheit-32);
}
</script>

</body>
</html>

```

JavaScript Functions

The temperature is 25 Celsius

8. Javascript Objects

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Objects</h2>
<p>An object method is a function definition, stored as a property value.</p>

<p id="pio"></p>

<script>
// Create an object:
const person = {
  firstName: "Piona",
  lastName: "Roskha",
  id: 5566,
  fullName: function() {
    return this.firstName + " " + this.lastName;
  }
};

// Display data from the object:
document.getElementById("pio").innerHTML = person.fullName();
</script>

</body>
</html>

```

JavaScript Objects

An object method is a function definition, stored as a property value.

Piona Roskha

9. Javascript Strings

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript String Properties</h2>

<p>The length property returns the length of a string:</p>

<p id="pio"></p>

<script>
let text = "ABCDEFGHI";
document.getElementById("pio").innerHTML = text.length;
</script>

</body>
</html>

```

JavaScript String Properties

The length property returns the length of a string:

9

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Strings</h2>

<p id="pio"></p>

<script>
let x = "John";           // x is a string
let y = new String("John"); // y is an object

document.getElementById("pio").innerHTML =
typeof x + "<br>" + typeof y;
</script>

</body>
</html>

```

JavaScript Strings

string
object

10. Javascript Tempaltes Literals

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Template Literals</h2>

<p>With back-ticks, you can use both single and double quotes inside a string:
</p>

<p id="pio"></p>

<p>Template literals are not supported in Internet Explorer.</p>

<script>
let text = `He's often called "Pion"`;
document.getElementById("pio").innerHTML = text;
</script>

</body>
</html>

```

JavaScript Template Literals

With back-ticks, you can use both single and double quotes inside a string:

He's often called "Pion"

Template literals are not supported in Internet Explorer.

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Template Literals</h2>

<p>Template literals allows variables in strings:</p>

<p id="pio"></p>

<p>Template literals are not supported in Internet Explorer.</p>

<script>
let header = "Templates Literals";
let tags = ["template literals", "javascript", "es6"];

let html = `<h2>${header}</h2><ul>`;

for (const x of tags) {
  html += `<li>${x}</li>`;
}

html += `</ul>`;
document.getElementById("pio").innerHTML = html;
</script>

</body>
</html>

```

JavaScript Template Literals

Template literals allows variables in strings:

Templates Literals

- template literals
- javascript
- es6

Template literals are not supported in Internet Explorer.

11. Javascript Number Methods



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

<h2>JavaScript Number Methods</h2>

<p>The toFixed() method rounds a number to a given number of digits.</p>
<p>For working with money, toFixed(2) is perfect.</p>

<p id="pio"></p>

<script>
let x = 9.656;
document.getElementById("pio").innerHTML =
  x.toFixed(0) + "<br>" +
  x.toFixed(2) + "<br>" +
  x.toFixed(4) + "<br>" +
  x.toFixed(6);
</script>

</body>
</html>
```

JavaScript Number Methods

The toFixed() method rounds a number to a given number of digits.

For working with money, toFixed(2) is perfect.

```
10
9.66
9.6560
9.656000
```

12. Javascript Array Methods

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>pop()</h2>
<p>The pop() method removes the last element from an array.</p>

<p id="demo1"></p>
<p id="demo2"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits;
fruits.pop();
document.getElementById("demo2").innerHTML = fruits;
</script>

</body>
</html>

```

JavaScript Array Methods

pop()

The pop() method removes the last element from an array.

Banana,Orange,Apple,Mango

Banana,Orange,Apple

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<h2>push()</h2>
<p>The push() method appends a new element to an array.</p>

<button onclick="myFunction()">Try it</button>
<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML = fruits;

function myFunction() {
  fruits.push("Kiwi");
  document.getElementById("demo").innerHTML = fruits;
}
</script>

</body>
</html>

```

JavaScript Array Methods

push()

The push() method appends a new element to an array.

[Try it](#)

Banana,Orange,Apple,Mango

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Methods</h2>
<p>Array elements are accessed using their index number:</p>

<p id="demo1"></p>
<p id="demo2"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits;
fruits[0] = "Kiwi";
document.getElementById("demo2").innerHTML = fruits;
</script>

</body>
</html>

```

JavaScript Array Methods

Array elements are accessed using their index number:

Banana,Orange,Apple,Mango

Kiwi,Orange,Apple,Mango

13. Javascript Booleans

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Booleans</h2>
<p id="pio"></p>

<script>
document.getElementById("pio").innerHTML =
"100 is " + Boolean(100) + "<br>" +
"3.14 is " + Boolean(3.14) + "<br>" +
"-15 is " + Boolean(-15) + "<br>" +
"Any (not empty) string is " + Boolean("Hello") + "<br>" +
"Even the string 'false' is " + Boolean('false') + "<br>" +
"Any expression (except zero) is " + Boolean(1 + 7 + 3.14);
</script>

</body>
</html>

```

JavaScript Booleans

100 is true







3.14 is true

-15 is true

Any (not empty) string is true

Even the string 'false' is true

Any expression (except zero) is true



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

<h2>JavaScript Booleans</h2>
<p>Never create booleans as objects.</p>
<p>Booleans and objects cannot be safely compared.</p>

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

<script>
let x = false;           // x is a boolean
let y = new Boolean(false); // y is an object
document.getElementById("demo").innerHTML = typeof x + "<br>" + typeof y;
</script>

</body>
</html>
```

JavaScript Booleans

Never create booleans as objects.

Booleans and objects cannot be safely compared.

boolean
object

14. Javascript Switch Statement



```
<h2>JavaScript switch</h2>

<p id="pio"></p>

<script>
let day;
switch (new Date().getDay()) {
  case 0:
    day = "Minggu";
    break;
  case 1:
    day = "Senin";
    break;
  case 2:
    day = "Selasa";
    break;
  case 3:
    day = "Rabu";
    break;
  case 4:
    day = "Kamis";
    break;
  case 5:
    day = "Jumat";
    break;
  case 6:
    day = "Sabtu";
    break;
}
```

JavaScript switch

Hari ini adalah Jumat

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript switch</h2>

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

<script>
let text;
switch (new Date().getDay()) {
  case 4:
  case 5:
    text = "Soon it is Weekend";
    break;
  case 0:
  case 6:
    text = "It is Weekend";
    break;
  default:
    text = "Looking forward to the Weekend";
}
document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>

```

JavaScript switch

Soon it is Weekend

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript switch</h2>

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

<script>
let x = "0";

switch (x) {
  case 0:
    text = "Off";
    break;
  case 1:
    text = "On";
    break;
  default:
    text = "No value found";
}
document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>

```

JavaScript switch

No value found

15. Javascript Break and Continue

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Loops</h2>

<p>A loop with a <b>break</b> statement.</p>

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

<script>
let text = "";
for (let i = 0; i < 10; i++) {
  if (i === 3) { break; }
  text += "The number is " + i + "<br>";
}

document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>

```

JavaScript Loops

A loop with a **break** statement.

The number is 0
The number is 1
The number is 2

```

<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Loops</h2>

<p>A loop with a <b>continue</b> statement.</p>

<p>A loop which will skip the step where i = 3.</p>

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

<script>
let text = "";
for (let i = 0; i < 10; i++) {
  if (i === 3) { continue; }
  text += "The number is " + i + "<br>";
}

document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>

```

JavaScript Loops

A loop with a **continue** statement.

A loop which will skip the step where i = 3.

The number is 0
The number is 1
The number is 2
The number is 4
The number is 5
The number is 6
The number is 7
The number is 8
The number is 9



Run >

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

<h2>JavaScript break</h2>

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

<script>
const cars = ["BMW", "Volvo", "Saab", "Ford"];
let text = "";

list: {
  text += cars[0] + "<br>";
  text += cars[1] + "<br>";
  break list;
  text += cars[2] + "<br>";
  text += cars[3] + "<br>";
}




document.getElementById("demo").innerHTML = text;
</script>


</body>
</html>
```

JavaScript break

BMW
Volvo

EXERCISE





Completed 5 of 67 Exercises:

JS Variables ✓

✓ Exercise 1




✓ Exercise 2


✓ Exercise 3

✓ Exercise 4

✓ Exercise 5

[Go to JS Variables Tutorial](#)





Completed 10 of 67 Exercises:

JS Variables ✓

JS Operators ✓

✓ Exercise 1




✓ Exercise 2


✓ Exercise 3

✓ Exercise 4

✓ Exercise 5

[Go to JS Operators Tutorial](#)





Completed 11 of 67 Exercises:

JS Variables ✓

JS Operators ✓

JS Data Types ✓

✓ Exercise 1

[Go to JS Data Types Tutorial](#)

<div> </div> <div> </div> <div>Completed 15 of 67 Exercises:</div> <div> <div>JS Variables ✓</div> <div>JS Operators ✓</div> <div>JS Data Types ✓</div> <div>JS Functions ✓</div> </div> <div> <div>✓ Exercise 1</div> <div>✓ Exercise 2</div> <div>✓ Exercise 3</div> <div>✓ Exercise 4</div> </div> <div> Go to JS Functions Tutorial </div>	<div> </div> <div> </div> <div>Completed 18 of 67 Exercises:</div> <div> <div>JS Variables ✓</div> <div>JS Operators ✓</div> <div>JS Data Types ✓</div> <div>JS Functions ✓</div> <div>JS Objects ✓</div> </div> <div> <div>✓ Exercise 1</div> <div>✓ Exercise 2</div> <div>✓ Exercise 3</div> </div> <div> Go to JS Objects Tutorial </div>	<div> </div> <div> </div> <div>Completed 21 of 67 Exercises:</div> <div> <div>JS Variables ✓</div> <div>JS Operators ✓</div> <div>JS Data Types ✓</div> <div>JS Functions ✓</div> <div>JS Objects ✓</div> <div>JS Events ✓</div> <div>JS Strings ✓</div> </div> <div> <div>✓ Exercise 1</div> <div>✓ Exercise 2</div> <div>✓ Exercise 3</div> </div>
<div> </div> <div> </div> <div>Completed 24 of 67 Exercises:</div> <div> <div>JS String Methods</div> <div>JS Arrays</div> <div>JS Array Methods ✓</div> </div> <div> <div>✓ Exercise 1</div> <div>✓ Exercise 2</div> <div>✓ Exercise 3</div> </div> <div> Go to JS Array Methods Tutorial </div>	<div> </div> <div> </div> <div>Completed 26 of 67 Exercises:</div> <div> <div>JS Math</div> <div>JS Comparisons</div> <div>JS Conditions</div> <div>JS Switch ✓</div> </div> <div> <div>✓ Exercise 1</div> <div>✓ Exercise 2</div> </div> <div> Go to JS Switch Tutorial </div>	<div> </div> <div> </div> <div>Completed 28 of 67 Exercises:</div> <div> <div>JS Conditions</div> <div>JS Switch ✓</div> <div>JS For Loops</div> <div>JS While Loops</div> <div>JS Break Loops ✓</div> </div> <div> <div>✓ Exercise 1</div> <div>✓ Exercise 2</div> </div> <div> Go to JS Break Loops Tutorial </div>