

# **Dahsyatkan Aplikasi PHP dengan Sentuhan JavaScript**

**Adi Chandra Setiawan**



**CV. LOKOMEDIA**

# **Dahsyatkan Aplikasi PHP dgn Sentuhan JavaScript**

Perpustakaan Nasional : Katalog Dalam Terbitan (KDT)

Penulis : Adi Chandra Setiawan

Dahsyatkan Aplikasi PHP dengan Sentuhan JavaScript

- Cet. I. - Yogyakarta : Penerbit Lokomedia, 2015

142 halaman; 14 x 21 cm

ISBN : 978-602-71905-3-5

Penerbit Lokomedia,

Cetakan Pertama : Juni 2015

Editor : Lukmanul Hakim

Cover : Subkhan Anshori

Layout : Lukmanul Hakim

Diterbitkan pertama kali oleh :

**CV. LOKOMEDIA**

Jl. Jambon, Perum. Pesona Alam Hijau 2 Kav. B-4, Kricak  
Yogyakarta 55242.

email : [redaksi@bukulokomedia.com](mailto:redaksi@bukulokomedia.com)

website : [www.bukulokomedia.com](http://www.bukulokomedia.com)

Copyright © Lokomedia, 2015

Hak Cipta dilindungi oleh Undang-Undang

Dilarang memperbanyak, mencetak ataupun menerbitkan sebagian maupun seluruh isi buku ini tanpa izin tertulis dari penerbit.

# UCAPAN TERIMAKASIH

Pertama-tama, saya panjatkan puji syukur kehadirat **ALLAH SWT** atas segala rahmat dan hidayah -Nya sehingga terbitnya buku ini.

Ucapan terimakasih sebesar-besarnya saya berikan kepada orang tua saya, **Rachmat Hidayat** dan **Sri Wahyu Ningsih** atas berkat Doa dan Dukungnya.

Istri, **Ira Handayani** makasih atas pengertian dan dukungnya.

Kakak dan kedua adik, **Wahyu Dewi Rahmawati**, **Astrit Tri Wulandari** dan **Anton Yuniarto** yang selalu memberikan semangat.

Muhamad Hanafi dan **I Gusti Bayu Artayoga**, terimakasih atas bantuan desain aplikasinya.

**Ana Lia Agustina**, terimakasih bantuan translatenya.

Serta para **Dosen STMIK AMIKOM YOGYAKARTA** dan **STMIK NUSA MANDIRI**, terimakasih atas ilmunya.

dan seluruh keluarga Besar **Bina Sarana Informatika**.

Akhir kata, saya ucapkan terimakasih kepada semua yang telah membantu dalam terbitnya buku ini.

*Halaman ini Sengaja Dikosongkan*

**[www.bukulokomedia.com](http://www.bukulokomedia.com)**

# KATA PENGANTAR

Kehadiran teknologi saat ini begitu pesat, seperti yang bisa kita lihat hampir disetiap perusahaan menggunakan teknologi dalam menunjang kegiatan operasional perusahaan. Kebutuhan akan efisiensi waktu dan bank data menyebabkan perusahaan merasa perlu menerapkan teknologi.

Tahun 2000an merupakan tahun booming website. Di tahun itu pula semakin banyak orang mengenal dan mempelajari bahasa pemrograman HTML dan PHP, bahkan jika mendengar kata website akan mengarah ke bahasa pemrograman PHP. Begitu pula jika mendengar PHP, PHP dikenal bahasa pemrograman untuk membuat website. Tetapi berbeda hal jika mendengar bahasa pemrograman Java, Visual Basic, DELPHI dll merupakan bahasa pemrograman aplikasi desktop.

Dengan hadirnya buku ini diharapkan dapat merubah pemikiran bahwa bahasa pemrograman PHP tidak hanya dapat digunakan untuk membuat website, tetapi aplikasinya dapat di buat dengan bahasa pemrograman PHP.

Sebagian besar buku ini bercerita tentang Javascript, walaupun dengan kombinasi JQuery dan Ajax di dalamnya. Pembahasan buku dimulai dari pengenalan javascript, JavaScript HTML DOM, Basis data, JqGrid, Barcode Scanner, trik-trik dan proyek aplikasi bengkel mobil.

Akhir kata, semoga buku dan ilmu yang sedikit ini bisa bermanfaat untuk pembaca. Penulis menyadari dalam buku ini tentu masih jauh dari sempurna. Oleh karena itu, pertanyaan, kritik, saran bisa ke [adichandraweb@gmail.com](mailto:adichandraweb@gmail.com) dan tunggu buku selanjutnya.

Cikarang, Desember 2014

Adi Chandra Setiawan

*Halaman ini Sengaja Dikosongkan*

**[www.bukulokomedia.com](http://www.bukulokomedia.com)**

# DAFTAR ISI

<b>BAB 1. Dasar-Dasar JavaScript</b> .....	1
1.1. Apa itu JavaScript .....	2
1.2. Variabel.....	3
1.3. Operator .....	4
1.4. Statement (Kondisional).....	8
1.5. Looping (Perulangan) .....	12
<b>BAB 2. JavaScript HTML DOM</b> .....	15
2.1. Pengenalan HTML DOM .....	16
2.2. Properti HTML DOM.....	17
2.3. Fungsi HTML DOM.....	17
2.4. Object HTML DOM .....	18
2.4.1. Document Object .....	18
2.4.2. Event Object.....	20
2.4.3. Element Object.....	24
<b>BAB 3. Basis Data dan Latihan Query</b> .....	27
3.1. Pengenalan Basis Data.....	28
3.2. SQL.....	30
3.2.1. Tipe-Tipe Data.....	30
3.2.2. Operator.....	32
3.2.3. Data Definition Language (DDL) .....	34
3.2.4. Data Manipulation Language (DML) .....	34

3.2.4.1. INSERT .....	35
3.2.4.2. UPDATE .....	35
3.2.4.3. DELETE.....	38
3.2.4.4. SELECT .....	38
3.2.5. Join antar Tabel .....	52
3.2.5.1. Cross Join.....	52
3.2.5.2. Inner Join.....	52
3.2.5.3. Outer Join.....	53
<b>BAB 4. JqGrid.....</b>	<b>55</b>
4.1. Pengenalan Grid.....	56
4.2. Contoh Grid.....	57
4.3. Grid dengan Pencarian.....	61
4.4. Grid dengan ComboBox Ajax .....	64
4.5. Grid dengan Button View .....	65
4.6. Grid dengan List Menu.....	67
4.7. Grid dengan Tanggal.....	68
4.8. Grid dengan Input Form .....	69
4.9. Grid dengan Popup Form .....	71
4.10. Grid dengan Live Edit.....	72
4.11. Grid dengan Multi Select.....	73
4.12. Grid dengan Sub Grid.....	74
4.13. Grid dengan Multi Sub .....	75
<b>BAB 5. Barcode Scanner .....</b>	<b>79</b>
5.1. Barcode.....	80
5.2. Barcode Scanner .....	81



5.3. Tipe Barcode .....	81
5.4. Cara Kerja Barcode Scanner.....	82
5.5. Cetak Label Barcode.....	83
<b>BAB 6. Aneka Trik Pilihan .....</b>	<b>87</b>
6.1. Membuat PopUp Search.....	88
6.2. Membuat PopUp Search dengan Mengirim Parameter.....	95
6.3. Membuat Search AutoSuggest .....	97
6.4. Membuat AddRow Menggunakan JavaScript.....	100
6.5. Membuat DropDown Menu pada AddRow dengan JavaScript.....	102
6.6. Membuat Perhitungan pada AddRow Menggunakan JavaScript .....	103
6.7. Membuat AddRow PopUp Menggunakan JqGrid .....	105
6.8. Membuat Kolaborasi AddRow Menggunakan JqGrid.....	107
6.9. Membuat AddRow Otomatis untuk Barcode.....	109
6.10. Membuat Fungsi Number (Tahun-Bulan-Urutan).....	111
<b>BAB 7. Proyek Aplikasi Bengkel.....</b>	<b>113</b>
7.1. Langkah Penerapan Aplikasi Bengkel .....	114
7.1.1. Copy File Aplikasi ke Localhost .....	114
7.1.2. Mengaktifkan Service Apache dan MySQL di Xampp.....	115
7.1.3. Setting Konfigurasi Apache.....	116
7.1.4. Mengimpor Database.....	118
7.1.5. Menjalankan Aplikasi Bengkel .....	120
7.2. Fitur-Fitur Aplikasi Bengkel.....	122
7.3. Use Case Diagram.....	126
7.4. Use Case Actor.....	127
7.5. Use Case Glossary .....	128

7.6. Use Case Spesification.....	129
7.7. Tools dan Bahasa Pemrograman Pendukung.....	139
7.7.1. Twitter Bootstrap.....	139
7.7.2. jQuery.....	139
7.7.3. AJAX.....	140
7.7.4. JSON.....	141
7.7.5. XML.....	141
<b>Daftar Pustaka.....</b>	<b>143</b>
<b>Tentang Penulis.....</b>	<b>145</b>

**BAB I**



---

# **MENGENAL JAVAScript**

## BAB 1

# Mengenal JavaScript



### 1.1 Apa itu JavaScript?

Apakah yang ada dipikiran Anda setelah mendengar JavaScript? Samakah Javascript dengan Java? Ayo siapa yang tau tunjuk tangan.heeee....ya tepat jawabannya beda, Java merupakan bahasa pemrograman berorientasi objek. Berbeda dengan JavaScript yang bukan bahasa pemrograman berorientasi objek. Java memerlukan Java Virtual Machine untuk menterjemahkannya, sedangkan JavaScript, bahasa yang di intepretasikan langsung oleh browser.

Piye wes mudeng? (#sudah mengerti.heeee..) yuk kita cari tau lebih dalam mengenai JavaScript.

**JavaScript** adalah bahasa pemrograman yang bersifat client side scripting yang eksekusinya berada di sisi client. Client disini adalah browser, seperti Internet Explorer (IE), Mozilla Firefox, Google Crome, Netscape, dan Opera.

#### **Kenapa kita harus mengerti JavaScript?**

Menurut pengalaman penulis dalam membuat aplikasi, JavaScript menjadi modal utama dalam menunjang pembuatan aplikasi, ada beberapa teknik dimana PHP tidak dapat mengatasinya, dengan kata lain membutuhkan bantuan JavaScript, seperti membuat pop-up, addrow validasi, dan sebagainya.

Penulis menyarankan menggunakan browser Mozilla Firefox, karena browser tersebut yang paling support dengan JavaScript.



JavaScript bersifat **case sensitive**, artinya huruf kecil dan huruf besar adalah berbeda. Setiap baris kode JavaScript dipisahkan baris baru atau bisa juga titik koma (;). Komentar dalam javascript diawali dengan // atau ditulis diantara /\* dan \*/.

## 1.2 Variabel

Penamaan variabel di Javascript harus berdasarkan aturan berikut:

- ✓ **Harus diawali huruf kecil, huruf besar, garis bawah (\_) atau \$.**

Sebagai contoh:

```
iniVariabel;  
INI_VARIABEL;  
_variabelKu;  
$ini_variabel;  
ini_var1a123;
```

- ✓ **Tidak boleh diawali dengan angka, @, \*, #, !, %, (, ), -, +.**

Sebagai contoh:

```
@variabel_salah;  
123_salah;  
*_VARIABEL_saLah;  
!_variabel_salah;  
%_variabel_salah;  
#$variabel_salah;
```

- ✓ **Tidak boleh mengandung spasi.**

Sebagai contoh:

```
va ri a_bel;  
ini bukan_variabel;
```

- ✓ **Tidak diperkenankan menggunakan keywords yang ada di JavaScript sebagai nama variabel.**

Untuk keywords yang dimaksud adalah: Abstract, boolean, break, byte,

case, catch, do, double, else, extends, false, final, implements, import, in, true, try, typeof, var, void, while, with, this, throw, throws, transient, Char, class, const, continue, default, delete, finally, float, for, function, goto, if, instanceof, int, interface, long, native, new, null, package, private, protected, static, super, switch, synchronized.

**Variabel dalam JavaScript harus dideklarasikan terlebih dahulu**, lalu di inialisasi dan inialisasi dilakukan secara bersamaan.

```
<script>
  var x = 10;
  var nama = "Bejo";
</script>
```

## 1.3 Operator

### 1.3.1 Operator Aritmatika

Berikut tabel untuk operator aritmatika.

Operator	Definisi	Contoh
+	Penambahan	$2 + 2 = 4$
-	Pengurangan	$4 - 2 = 2$
*	Pengalian	$4 * 2 = 8$
/	Pembagian	$4 / 2 = 2$
%	Modulus (sisa hasil pembagian)	$5 \% 2 = 1$

Berikut beberapa contoh script untuk operator aritmatika:

#### Script OA\_1.php

```
<script>
var a=2;
var b=3;
var c=a+b;
alert(c);
</script>
```

*Penjelasan Script:*

- ✓ Nilai 2 disimpan ke dalam var a.



- ✓ Nilai 3 disimpan ke dalam var b.
- ✓ Penambahan nilai 2 dan 3 disimpan di var c.
- ✓ Alert untuk menampilkan hasil.

### Script OA\_2.php

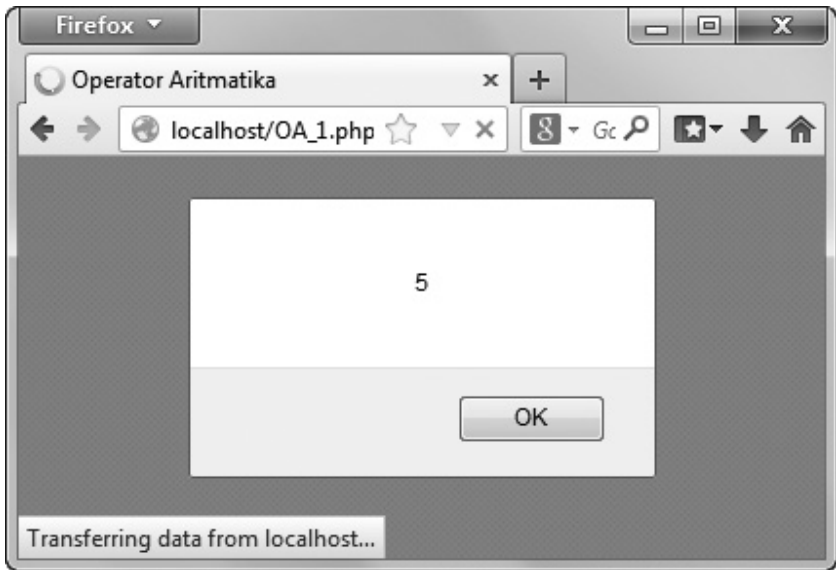
```
<script>  
var c=2+3;  
alert(c);  
</script>
```

#### Penjelasan Script:

Contoh script OA\_1.php dengan OA\_2.php mempunyai output yang sama, perbedaannya hanya pada OA\_1.php akan menyimpan nilai dahulu pada suatu variabel, lalu di tampilkan pada variabel baru, sedangkan pada OA\_2.php nilai langsung dihasilkan dari satu variabel.

#### Catatan:

Untuk menjalankan script melalui browser bisa dengan mengetikkan alamat **http://localhost/OA\_1.php**, maka tampil hasilnya dalam bentuk alert/kotak dialog seperti pada gambar 1.1.



Gambar 1.1 Hasil script OA\_1.php

Semua script yang dibahas pada Bab 1 ini bisa Anda temukan di CD terlampir, tepatnya pada folder **source-code/bab1**.

### 1.3.2 Operator Assignment

Berikut tabel untuk operator assignment.

Shorthand Operator	Definisi	Artinya
$x += y$	Ditambah dengan	$x = x + y$
$x -= y$	Dikurangi dengan	$x = x - y$
$x *= y$	Dikali dengan	$x = x * y$
$x /= y$	Dibagi dengan	$x = x / y$

Berikut contoh script untuk operator assignment:

#### Script OAS\_1.php

```
<script>
var x = 4;
var y = 2;
x -= y
alert(x);
</script>
```

*Penjelasan Script:*

- ✓ Nilai 4 disimpan ke dalam var x.
- ✓ Nilai 2 disimpan ke dalam var y.
- ✓ Alert untuk menampilkan hasil.

### 1.3.3 Operator Pembandingan

Berikut tabel untuk operator pembandingan.

Operator	Definisi	Contoh
$==$	Sama dengan	$var1 == \text{"tes"}$
$!=$	Tidak sama dengan	$x != y$





>	Lebih besar dari	$x > y$
<	Lebih kecil dari	$x < 6$
>=	Lebih besar sama dengan	$x \geq y$
<=	Lebih kecil sama dengan	$x < 5$

Berikut contoh script untuk operator perbandingan:

### Script OP\_1.php

```
<script>
var x = 4;
var y = 2;
if(x > y) {
  alert("x lebih besar dari y");
}
else{
  alert("x lebih kecil dari y");
}
</script>
```

### Penjelasan Script:

- ✓ var x = 4; //nilai 4 disimpan ke dalam var x.
- ✓ var y = 2; //nilai 2 disimpan ke dalam var y.
- ✓ if(x > y) //kondisi akan mengecek apakah x lebih besar dari y.
- ✓ alert("x lebih besar dari y"); //menampilkan kalimat x lebih besar dari y.
- ✓ alert("x lebih kecil dari y"); //menampilkan kalimat x lebih kecil dari y.

### Operator Logika

Berikut tabel untuk operator logika.

Operator	Definisi	Contoh
&&	Dan	$x \geq 5 \ \&\& \ x < 10$
	Atau	$x == 6 \    \ x == 12$
!	Bukan	!expression

Berikut contoh script untuk operator logika:



## Script OL\_1.php

```
<script>
var x = 76;
if(x >= 80){
alert("Nilai = A");
}else if(x >= 70 && x < 80){
alert("Nilai = B");
}else if(x >= 60 && x < 70){
alert("Nilai = C");
}else{ alert("Nilai = D"); }
</script>
```

### Penjelasan Script:

- ✓ **var x = 76;** // nilai 76 disimpan ke dalam var x.
- ✓ **if(x >= 80)** // kondisi mengecek apakah x lebih besar sama dengan 80.
- ✓ **alert("Nilai = A");** // menampilkan Nilai = A.
- ✓ **else if(x >= 70 && x < 80){** // kondisi x lebih besar sama dengan 70 dan x kurang dari 80, berjalan jika kondisi **if(x >= 80)** tidak terpenuhi.
- ✓ **alert("Nilai = B");** // menampilkan Nilai = B.
- ✓ **}else if(x >= 60 && x < 70){** // kondisi x lebih besar sama dengan 60 dan x kurang dari 70, berjalan jika kondisi **if(x >= 70 && x < 80)** tidak terpenuhi.
- ✓ **alert("Nilai = C");** // menampilkan Nilai = C.
- ✓ **alert("Nilai = D");** // menampilkan Nilai = D kondisi ini akan berjalan jika semua kondisi di atas tidak terpenuhi.

## 1.4 Statement (Kondisional)

Kondisional berguna untuk mengecek suatu kondisi dan melakukan suatu kode jika kondisi tersebut benar atau salah.

### 1.4.1 if

Sintaks-nya:

```
if(kondisi){
    kode yang dijalankan jika benar
}
```



## Script **if.php**

```
<script>
var x=1;
if(x==1){
alert("x berisi nilai 1"); }
</script>
```

### *Penjelasan Script:*

- ✓ **var x=1;** // variabel x berisi nilai 1.
- ✓ **if(x==1){** // kondisi x sama dengan 1.
- ✓ **alert("x berisi nilai 1");** // menampilkan x berisi nilai 1.

## 1.4.2 if - else

Sintaks-nya:

```
if(kondisi){
    kode yang dijalankan jika benar
}else{
    kode yang dijalankan jika salah
}
```

## Script **ifelse.php**

```
<script>
var x=5;
if(x==10){
alert("Sama");
}else{
alert("Tidak Sama");
}
</script>
```

### *Penjelasan Script:*

- ✓ **var x=5;** // variabel x berisi nilai 5.
- ✓ **if(x==10){** // kondisi jika x sama dengan 10.
- ✓ **alert("Sama");** // menampilkan Sama.
- ✓ **alert("Tidak Sama");** // menampilkan Tidak Sama.



### 1.4.3 if - else if - else

Jika Anda membutuhkan kondisi yang banyak.

Sintaks-nya:

```
if(kondisi 1){
    kode yang dijalankan jika kondisi 1 benar
}else if(kondisi 2){
    kode yang dijalankan jika kondisi 2 benar
}else if(kondisi 3){
    kode yang dijalankan jika kondisi 3 benar
}else{
    kode jika salah satu kondisi di atas tidak ada yang benar
}
```

#### Script `ifelseifelse.php`

```
<script>
var nilai = 80;
if(nilai >= 85){
    alert("A");
}else if(nilai >= 70 && nilai < 85){
    alert("B");
}else if(nilai >= 60 && nilai < 70){
    alert("C");
}else{
    alert("D");
}
</script>
```

#### *Penjelasan Script:*

- ✓ `var nilai = 80;` // nilai 80 disimpan ke dalam `var nilai`.
- ✓ `if(nilai >= 85)` // kondisi yang apakah nilai lebih besar sama dengan 85.
- ✓ `alert("A");` // menampilkan A.
- ✓ `else if(nilai >= 70 && x < 85){` // kondisi nilai lebih besar sama dengan 70 dan nilai kurang dari 85, berjalan jika kondisi `if(nilai >= 85)` tidak terpenuhi.
- ✓ `alert("B");` // menampilkan B.



- ✓  `}else if(nilai >= 60 && x < 70){ // kondisi nilai lebih besar sama dengan 60 dan nilai kurang dari 70, berjalan jika kondisi if(nilai >= 70 && nilai < 85) tidak terpenuhi.`
- ✓  `alert("C"); // menampilkan C.`
- ✓  `alert("D"); // menampilkan D, kondisi ini akan berjalan jika semua kondisi diatas tidak terpenuhi.`

### 1.4.4 Switch

Sama seperti if - else if - else, berguna jika membutuhkan kondisi yang banyak.

Sintaks-nya:

```
switch(ekspresi){
    case kondisi1 :
        kode yang dijalankan jika kondisi1 benar;
        break;
    case kondisi2 :
        kode yang dijalankan jika kondisi2 benar;
        break;
    case kondisi3 :
        kode yang dijalankan jika kondisi3 benar;
        break;
}
```

#### Script switch.php

```
<script>
var hobi = "sepakbola";
switch (hobi) {
case "musik":
alert("hobinya adalah musik");
break;
case "sepakbola":
alert("hobinya adalah sepakbola");
break;
case "olahraga":
alert("hobinya adalah olahraga");
break;
}
```

```
}  
</script>
```

*Penjelasan Script:*

- ✓ `var hobi = "sepakbola";` // variabel hobi di isikan nilai sepakbola.
- ✓ `switch (hobi) { case "musik":` // jika statement hobi adalah musik.
- ✓ `alert("hobinya adalah musik");` // menampilkan hobinya adalah musik.
- ✓ `case "sepakbola":` // jika statement hobi adalah sepakbola.
- ✓ `alert("hobinya adalah sepakbola");` // menampilkan hobinya adalah sepakbola.
- ✓ `case "olahraga":` // jika statement hobi adalah olahraga.
- ✓ `alert("hobinya adalah olahraga");` //menampilkan hobinya adalah jambu.
- ✓ `break;` // perintah untuk mengakhiri statement.

## 1.5 Perulangan (Looping)

Perulangan berguna untuk melakukan sesuatu secara berulang-ulang sebanyak jumlah yang ditentukan dan akan berakhir pada kondisi yang telah ditentukan.

### 1.5.1 For

Berguna untuk pengulangan yang sudah ditentukan terlebih dahulu jumlah perulangannya.

Sintaks-nya:

```
for(awal; kondisi; penambahan){  
    kode untuk dijalankan  
}
```

Script **for.php**

```
<script>  
for (i = 1; i <= 10; i++) {  
    document.write(i);  
}  
</script>
```



*Penjelasan Script:*

- ✓ `i = 1;` // i yang di isi nilai 1.
- ✓ `i <= 10` // i akan terus berulang ketika nilai i kurang atau sama dengan 10.
- ✓ `i++` // variabel counter agar bisa memenuhi kondisi akhir perulangan, akan berhenti jika i kurang atau sama dengan 10.
- ✓ `document.write(i);` // menampilkan hasil i.

## 1.5.2 While

Berguna untuk menjalankan suatu kode terus menerus selama kondisi bernilai TRUE.

Sintaks-nya:

```
while(kondisi){  
    kode untuk dijalankan;  
}
```

Script **while.php**

```
<script>  
var i=1;  
while(i<=5)  
{  
document.write("Nomor : "+i + "<br />");  
i++;  
}  
</script>
```

*Penjelasan Script:*

- ✓ `var i=1;` // var i berisi nilai 1.
- ✓ `while(i<=5)` // kondisi perulangan i kurang atau sama dengan 5.
- ✓ `document.write("Nomor : "+i + "<br />");` // menampilkan perulangan i.
- ✓ `i++;` // variabel counter.