

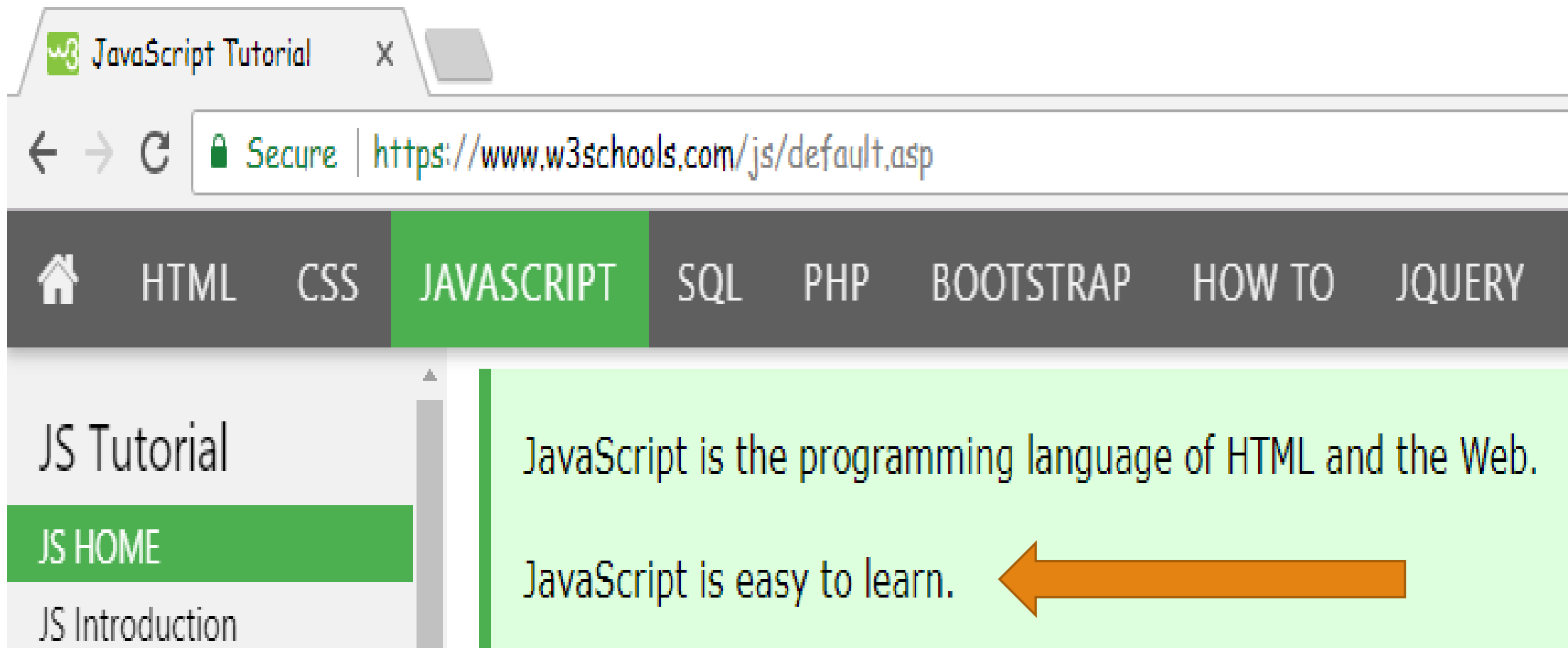
# JAVASCRIPT

---

Referensi:

<https://developer.mozilla.org/id/docs/web/javascript>

<https://www.w3schools.com/js/default.asp>



# DEFINISI

---

Javascript adalah bahasa pemrograman tingkat tinggi dan dinamis. Dengan Javascript, kita dapat membuat website menjadi interaktif.

# PENGUNAAN JAVASCRIPT

---

- ❑ Client Side / Front End
- ❑ Server Side / Back End
- ❑ Mobile Application
- ❑ Add On
- ❑ JSON

All Respondents

Professional Developers

Overview

Key Results

Developer Profile

**Technology**

**I. Most Popular Technologies**

II. Most Loved, Dreaded, and Wanted

III. Development Environments and Tools

IV. Blockchain in the Real World

V. Top Paying Technologies

VI. Correlated Technologies

Work

Community

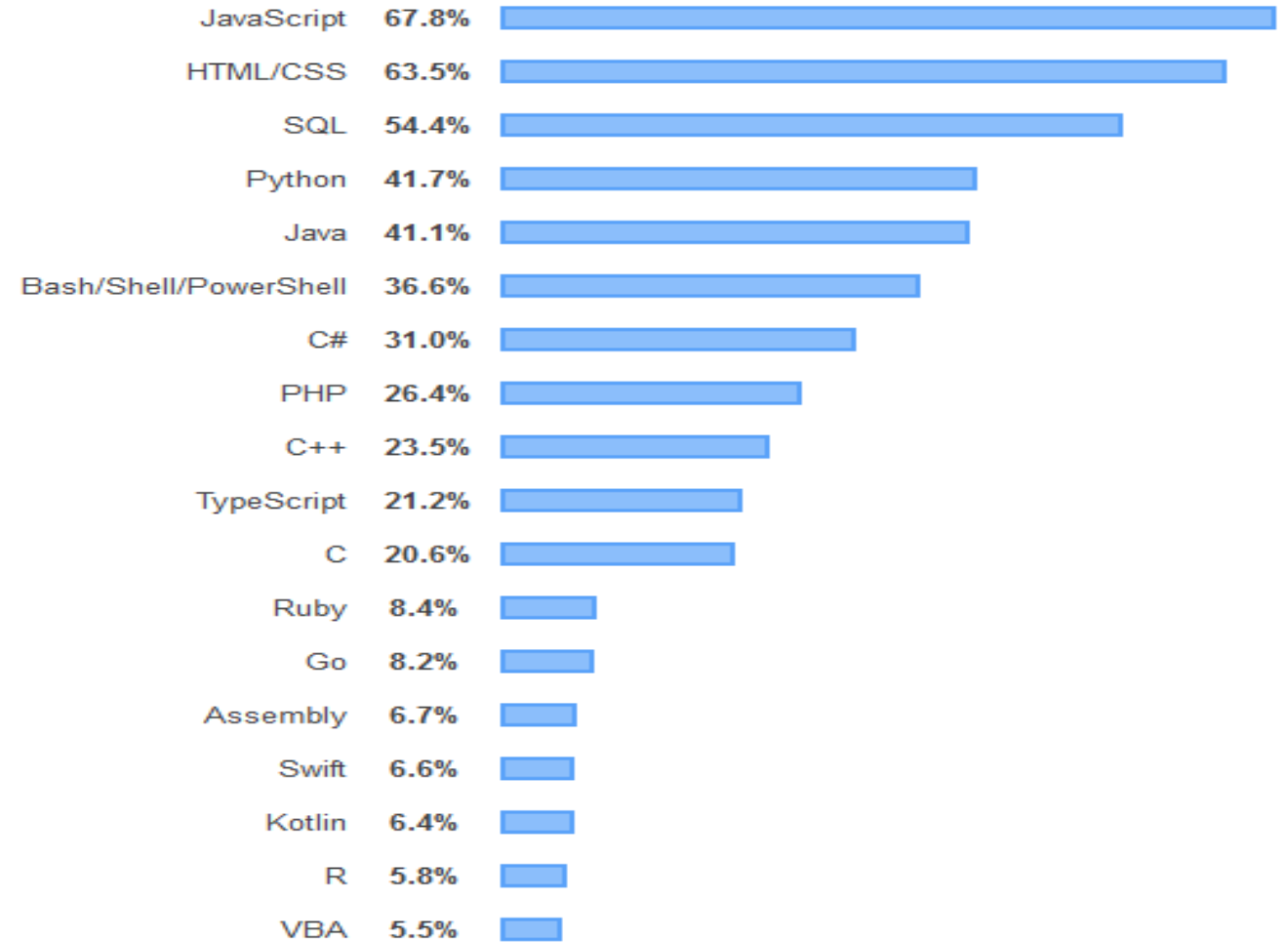
Methodology

Back to top ↱

Take control of your job search.

Stack Overflow Jobs puts developers first. No recruiter spam or fake job listings.

Browse jobs



<https://insights.stackoverflow.com/survey/2019>

# PENULISAN JAVASCRIPT

---

Penempatan elemen `<script>` yaitu di akhir elemen html. Jika JavaScript dimuat pertama kali dan seharusnya mempengaruhi HTML di bawahnya, kemungkinan ini tidak bisa bekerja, oleh karena itu JavaScript dimuat sebelum HTML bekerja seperti seharusnya. Oleh karena itu, strategi yang terbaik biasanya adalah di akhir halaman.

# VARIABEL

---

Variables merupakan wadah yang dapat diberi nilai. Kita bisa memulai mendeklarasikan variabel dengan keyword `var`, diikuti nama yang anda inginkan. Aturan penulisan variabel:

- ❑ Tidak boleh menggunakan spasi
- ❑ Tidak boleh dimulai dengan angka tapi boleh berisi angka
- ❑ Tidak boleh mengandung operator matematika dan logika
- ❑ Tidak boleh mengandung tanda baca kecuali garis bawah ( `_` )
- ❑ Huruf besar/kecil berpengaruh

Contoh penulisan yang dianjurkan : `var tamusBinTahir`

# TIPE DATA STRING

---

- ❑ Penulisannya dibungkus menggunakan " " atau ' ', contoh: "teknik" 'informatika'
- ❑ Jika kita memerlukan (') di dalam penulisan maka gunakan " " untuk membungkusnya begitupun sebaliknya, contoh "luffy's"
- ❑ Jika kita membutuhkan keduanya dalam penulisan maka tambahkan \ sehingga kutipnya terbaca sebagai string kutip. Jadi penulisannya \' atau \", contoh:  
"I\'ve thought of a way to fight with all my strength so I won\'t lose to anybody!" Monkey D Luffy arc Enies Lobby'



# TIPE DATA NUMBER

---

- ❖ Untuk tipe data number dapat ditulis langsung
- ❖ Untuk bilangan decimal digunakan tanda (.) contoh 3.14
- ❖ Jika penulisan angka dimulai dari 0 maka akan dianggap sebagai bilangan octal contoh: 077

# TIPE DATA BOOLEAN

---

- ❑ Boolean menghasilkan nilai true atau false
- ❑ Biasanya digunakan pada pengkondisian (if else) dan function

Contoh :

- ❑ `10 > 20`
- ❑ `10 > 20`
- ❑ `"tahir" == "TAHIR"`

# ARITHMETIC OPERATORS

---

- + : digunakan untuk penjumlahan
- - : digunakan untuk pengurangan
- / : digunakan untuk pembagian
- \* : digunakan untuk perkalian
- % : digunakan untuk mencari sisa hasil bagi

# STRING OPERATOR

---

➤ + : digunakan untuk menggabungkan string

# ASSIGNMENT OPERATORS

---

□ = : digunakan untuk menugaskan pada variabel

Untuk menggabungkan nilai variabel sebelumnya dan variabel berikutnya yang sama nama variabelnya kita menggunakan operator berikut:

□ \*= :  $\text{var}_{\text{awal}} * \text{var}_{\text{berikutnya}}$

□ /= :  $\text{var}_{\text{awal}} / \text{var}_{\text{berikutnya}}$



□ += :  $\text{var}_{\text{awal}} + \text{var}_{\text{berikutnya}}$

□ -= :  $\text{var}_{\text{awal}} - \text{var}_{\text{berikutnya}}$

□ %= :  $\text{var}_{\text{awal}} \% \text{var}_{\text{berikutnya}}$

# COMPARISON OPERATORS

---

<input type="checkbox"/> ==	: sama dengan		(tidak membandingkan tipe data)
<input type="checkbox"/> !=	: tidak sama dengan		
<input type="checkbox"/> ===	: strict sama dengan		(membandingkan tipe data)
<input type="checkbox"/> !==	: strict tidak sama dengan		
<input type="checkbox"/> <	: lebih kecil dari		
<input type="checkbox"/> >	: lebih besar dari		
<input type="checkbox"/> <=	: lebih kecil sama dengan		
<input type="checkbox"/> >=	: lebih besar sama dengan		

# LOGICAL OPERATORS

---

- && : logika and
- || : logika or
- ! : logika not

# CONDITIONAL (TERNARY) OPERATORS

- ❑ Operator ini memerlukan 3 operan
- ❑ Operator ini digunakan untuk mengecek kondisi, dan jika kondisinya bernilai true akan mengambil kondisi kedua dan jika bernilai false akan mengambil kondisi ketiga.

true  
 ↓  
 (Expression) ? expression : expression



# FUNCTION

---

- ❑ Function merupakan suatu pondasi dasar dalam javascript
- ❑ Function adalah prosedur JavaScript melakukan tugas atau menghitung nilai.
- ❑ Untuk menggunakan function, kita harus mendefinisikannya kemudian memanggilnya.

# BUILT IN FUNCTION

---

- ❑ Function yang sudah disediakan oleh javascript
- ❑ Untuk menggunakannya, kita tinggal memanggilnya.
- ❑ Contoh : `alert()`

# USER DEFINED FUNCTION

---

- ❑ Function yang dibuat oleh user dengan menggunakan keyword function
- ❑ Nama function boleh ada boleh juga tidak
- ❑ Function boleh memiliki parameter boleh juga tidak
- ❑ Parameternya disimpan di dalam ()
- ❑ Parameter boleh lebih dari satu dan penulisannya dipisahkan oleh koma
- ❑ Body function dibungkus dengan {}
- ❑ Function mengembalikan nilai atau boleh juga tidak

# CONTOH USER DEFINED FUNCTION

---

Pembuatan function

```
function penjumlahan(x, y) {  
  
    var hasil;  
  
    hasil = x + y;  
  
    return hasil;  
  
}
```

Pemanggilan function

```
console.log(penjumlahan(7, 8));
```

# USER DEFINED FUNCTION

---

- ❑ Jika argument lebih banyak dari parameter maka argument kelebihannya akan diabaikan
- ❑ Jika parameter lebih banyak dari argument maka parameter kelebihannya akan diisi undefined

# CONDITIONS IF

---

```
if (kondisi) {  
    blok kode yang akan dieksekusi jika bernilai true  
}
```




```
var number = prompt('input number');  
if (number % 2 === 0 ) {  
    alert(number + " adalah bilangan genap");  
}
```

# CONDITIONS IF ELSE

---

```
if (kondisi) {  
    blok kode yang akan dieksekusi jika bernilai true  
} else {  
    blok kode yang akan dieksekusi jika bernilai false  
}
```



```
var number = prompt('input number');  
if (number % 2 === 0) {  
    alert(number + " adalah bilangan genap");  
} else {  
    alert(number + " adalah bilangan ganjil");  
}
```

# ARRAY

---

Array Adalah Tipe Data Yang Berisi Kumpulan Dari Nilai Atau Tipe Data Lain. Nilai Di Dalam Array Disebut Dengan Elemen, Dan Setiap *Elemen* Memiliki '*Nomor Urut*' Yang Dikenal Dengan Istilah Index.



# ARRAY

---

Elemen dari array bisa bertipe data string, number dan boolean dalam sebuah array yang sama, bahkan elemen dari array bisa berupa objek atau array yang lain.

# ARRAY

---

```
Var prodi = ['Informatika', 'Mesin', 'Elektro'];
```

```
Var angkatan = [2019, 2018, 2017];
```

```
Var mhs = ['Andi', 2019, true];
```

```
Var mhs = ['Andi', [2019, 'Informatika', 'UPA']];
```

# DOCUMENT OBJECT MODEL

---

Secara sederhananya, **DOM** adalah kumpulan aturan atau cara bagi programmer untuk '*memanipulasi*' apapun yang tampil dalam halaman web.

# DOM SELECTION

---

1. `getElementById( )`
2. `getElementsByTagName( )`
3. `getElementsByClassName( )`
4. `querySelector( )`
5. `querySelectorAll( )`

# EVENT

---

Event adalah sesuatu yang terjadi di elemen HTML. Sebuah event bisa dipicu oleh browser, bisa juga dipicu oleh pengguna.

Adapun jenis-jenis event di HTML antara lain :

1. `OnClick`: saat pengguna melakukan klik pada sebuah elemen HTML.
2. `Ondblclick`: saat pengguna melakukan double klik pada sebuah elemen HTML.
3. `Onmouseover`: saat pengguna mengarahkan pointer pada sebuah elemen HTML.
4. `Onmouseout`: saat pengguna memindahkan pointer dari sebuah elemen HTML

Next level → php