

JS

dasar pemrograman

dengan Javascript

Javascript ?

Bahasa Pemrograman

variabel
tipe data

pengkondisian

pengulangan

object

array function

...

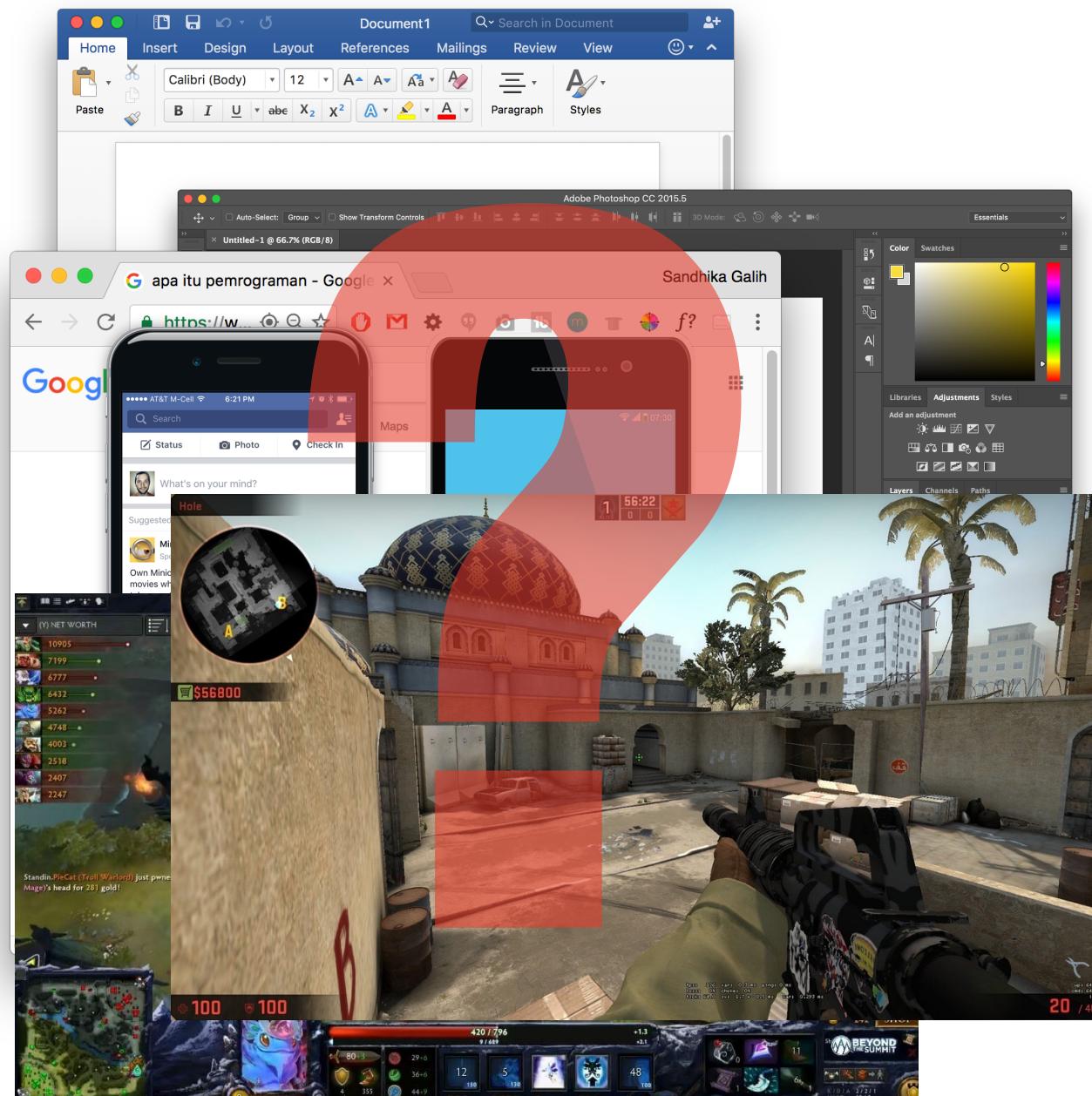
- Program
- Pemrograman
- Bahasa Pemrograman

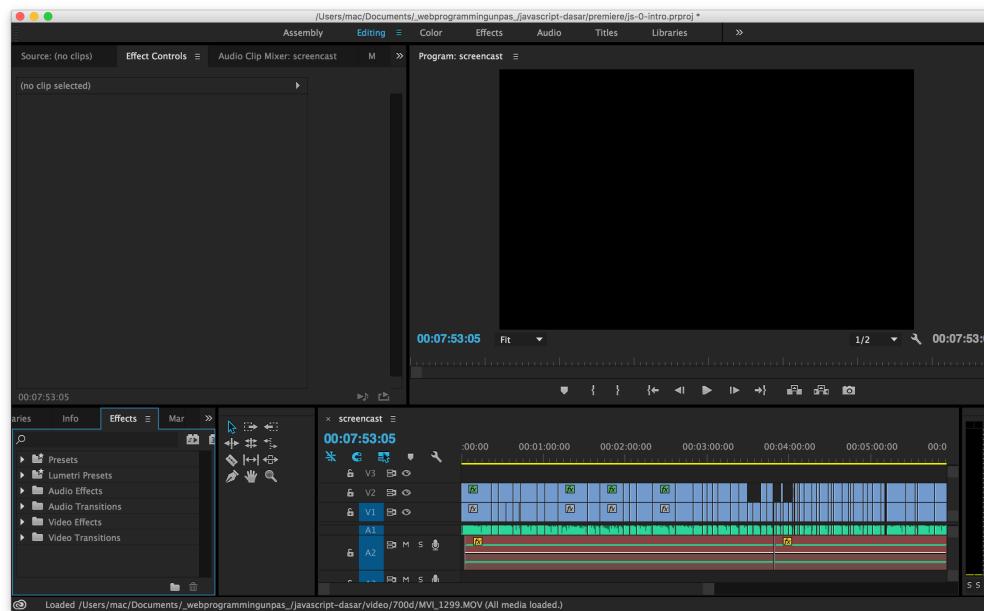
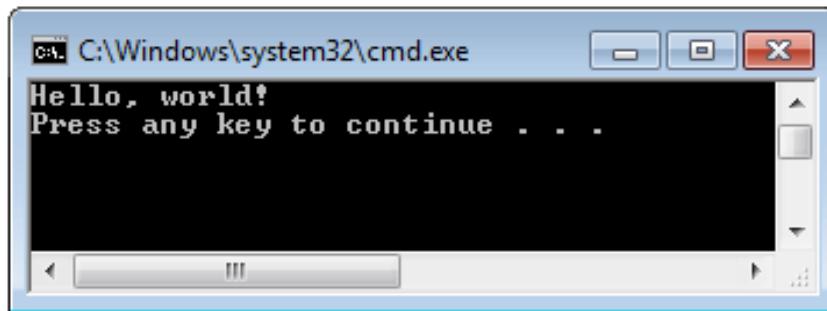
Program



"program adalah kumpulan instruksi.."







Bahasa Pemrograman



C

C++

C#

Java

Javascript

PHP

Python

Objective-C

Ruby

Visual Basic

kenapa banyak?

Bahasa Mesin

00000000	4F	UU	4B	UU												
000000e0	00	00	01	50	B4	00	7D	00	32	00	0E	00	02	00	00	00
000000f0	FF	FF	80	00	43	00	61	00	6E	00	63	00	65	00	6C	00
00000100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	01	50
00000110	EA	00	7D	00	32	00	0E	00	09	00	00	00	FF	FF	80	00
00000120	26	00	48	00	65	00	6C	00	70	00	00	00	00	00	00	00
00000130	00	00	00	00	00	00	00	00	80	08	81	50	0E	00	3A	00
00000140	3B	00	0E	00	2F	25	00	00	FF	FF	81	00	00	00	00	00
00000150	00	00	00	00	00	00	00	00	00	00	02	50	0E	00	30	00
00000160	1E	00	08	00	EE	25	00	00	FF	FF	82	00	46	00	69	00
00000170	6C	00	65	00	20	00	54	00	79	00	70	00	65	00	00	00
00000180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	02	50
00000190	54	00	30	00	2C	00	08	00	EF	25	00	00	FF	FF	82	00
000001a0	50	00	61	00	72	00	73	00	69	00	6E	00	67	00	20	00
000001b0	52	00	75	00	6C	00	65	00	73	00	00	00	00	00	00	00
000001c0	00	00	00	00	00	00	00	00	07	00	00	50	06	00	07	00
000001d0	1A	01	71	00	ED	25	00	00	FF	FF	80	00	00	00	00	00
000001e0	00	00	00	00	00	00	00	00	00	00	02	50	0E	00	11	00
000001f0	3E	00	08	00	EC	25	00	00	FF	FF	82	00	53	00	65	00
00000200	6C	00	65	00	63	00	74	00	20	00	52	00	75	00	6C	00
00000210	65	00	20	00	46	00	6F	00	72	00	20	00	46	00	69	00
00000220	6C	00	65	00	00	00	00	00	00	00	00	00	00	00	00	00
00000230	80	08	81	50	0E	00	1B	00	08	01	0E	00	EB	25	00	00
00000240	FF	FF	81	00	00	00	00	00	00	00	00	00	00	00	00	00
00000250	00	00	02	50	19	00	61	00	37	00	08	00	6B	26	00	00

Bahasa Manusia

JavaScript, ActionScript

PHP, Ruby, Python

Java, VB.NET

C++

C

Bahasa Assembly

Bahasa Mesin

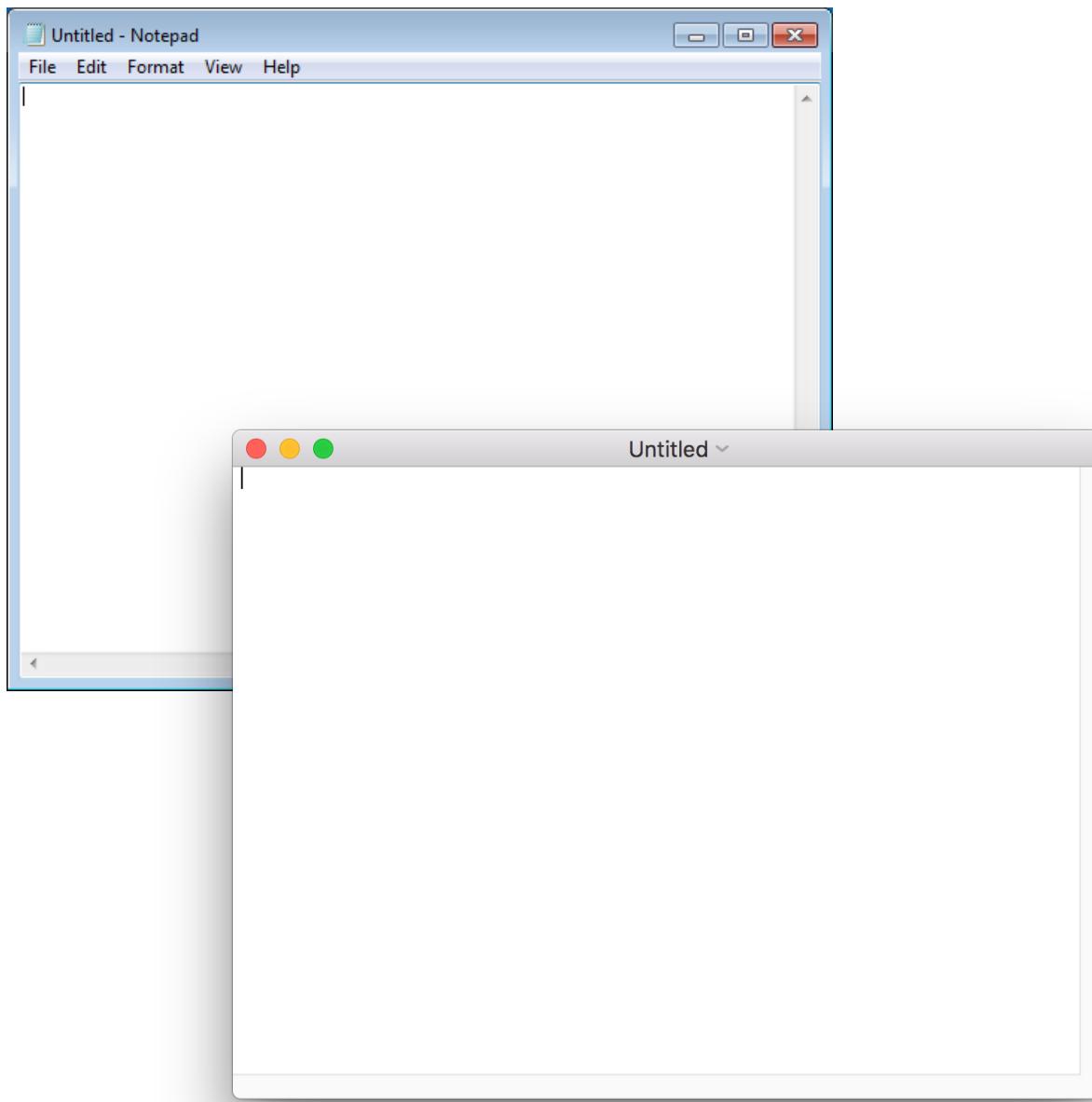
Bahasa Tingkat Tinggi /
High Level Language

Bahasa Tingkat Rendah /
Low Level Language



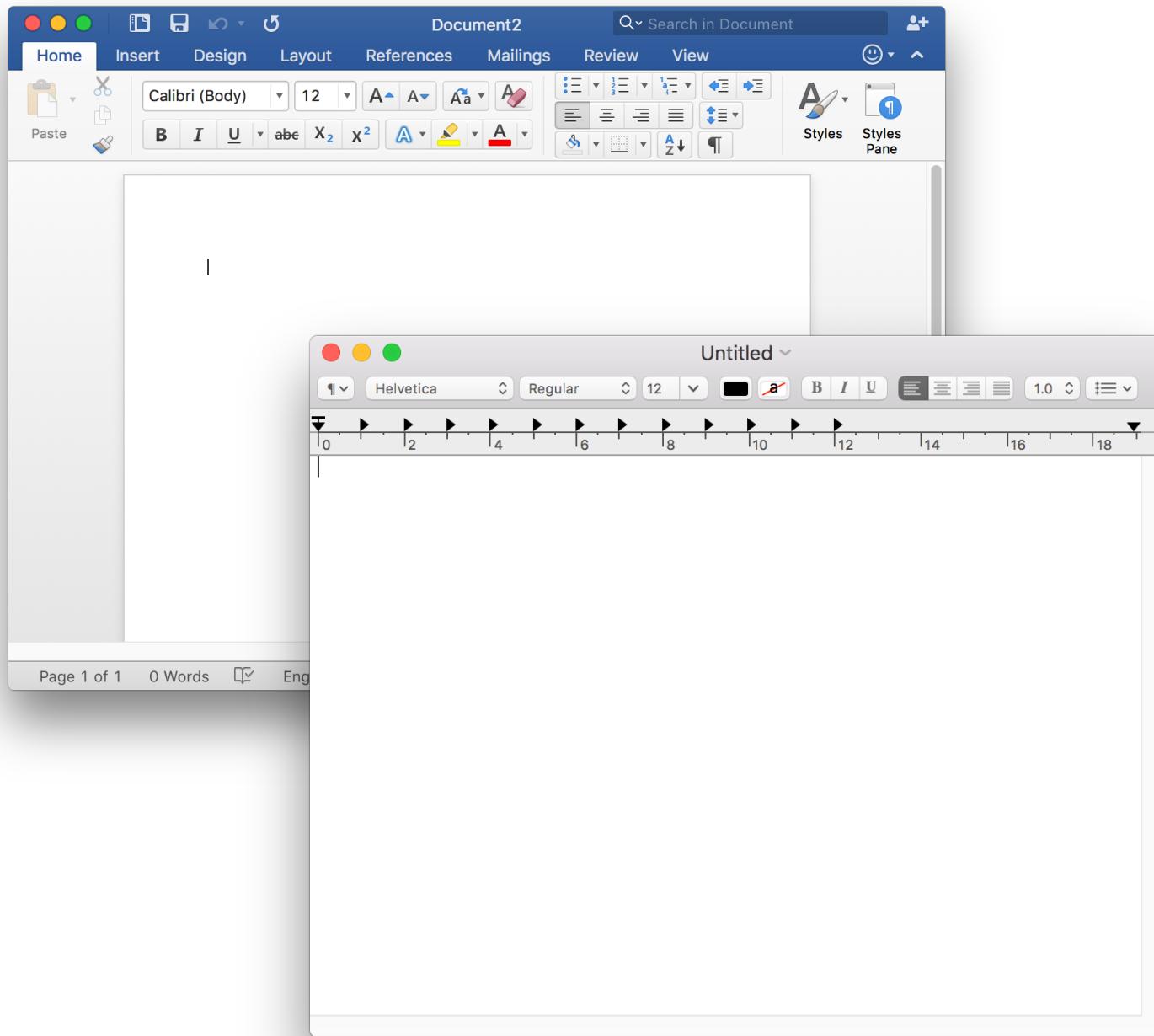
Source Code

Plain Text



The image shows four separate Mac OS X terminal windows, each displaying a different "Hello, world!" program. The windows are arranged vertically, overlapping slightly.

- Top window:** hello.js. Contains the JavaScript code: `alert("Hello, world!");`
- Second window from top:** hello.py. Contains the Python code: `print "Hello, world!";`
- Third window from top:** hello.ruby. Contains the Ruby code: `puts "Hello, world!";`
- Bottom window:** hello.pl. Contains the Perl code: `say "Hello, world!";`



The image shows four separate Mac OS X terminal windows, each displaying a different "Hello, world!" program. The windows are arranged vertically, overlapping slightly.

- Top window:** hello.js. Contains the JavaScript code: `alert("Hello, world!");`
- Second window from top:** hello.py. Contains the Python code: `print "Hello, world!";`
- Third window from top:** hello.ruby. Contains the Ruby code: `puts "Hello, world!";`
- Bottom window:** hello.pl. Contains the Perl code: `say "Hello, world!";`

Python

```
print "Hello, world!";
```

LUA

print “Hello, world!”;

PHP

print “Hello, world!”;

PHP

<?php

 print “Hello, world!”;

?>

PASCAL

```
program HelloWorld;  
  
begin  
    writeln('Hello, world!');  
end.
```

C

```
#include <stdio.h>

int main(void)
{
    printf("Hello, world!\n");
    return 0;
}
```

Java

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello, world!");  
    }  
}
```

Code Editor

A screenshot of a code editor window titled "script.js". The status bar at the bottom indicates "Line 16, Column 4", "Tab Size: 4", and "JavaScript". The code itself is as follows:

```
1 var keyword = document.getElementById('keyword'),
2     tujuan = document.getElementById('tujuan'),
3     tombol = document.getElementById('tombol-cari');
4
5 tombol.style.display = 'none';
6
7 keyword.addEventListener('keyup', function() {
8     var xhr = new XMLHttpRequest();
9     xhr.onreadystatechange = function() {
10         if( xhr.status === 200 && xhr.readyState === 4 ) {
11             tujuan.innerHTML = xhr.responseText;
12         }
13     }
14     xhr.open('GET', 'ajax/cari.php?keyword=' + keyword.value, true);
15     xhr.send();
16 });

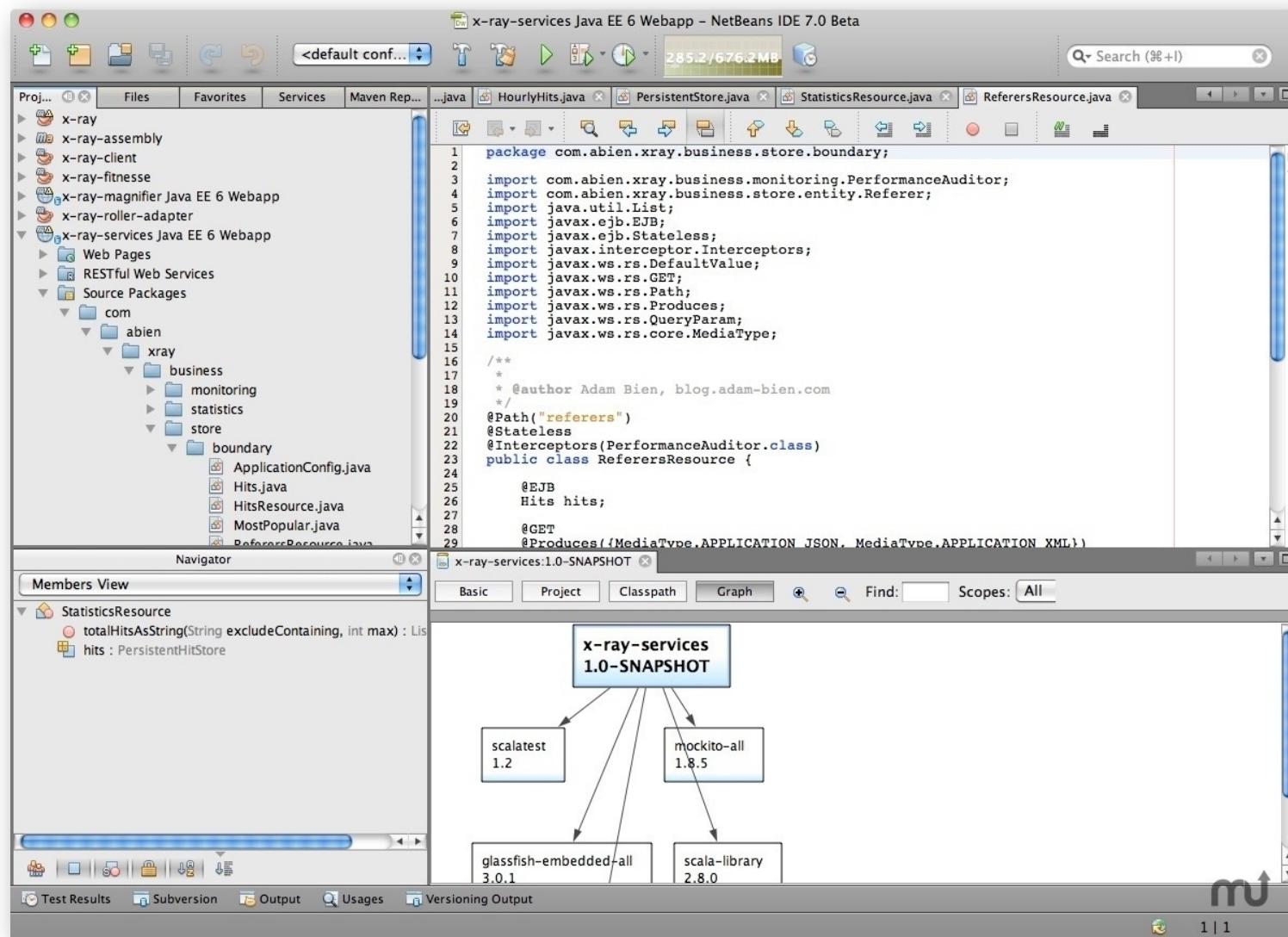
```

Code Editor

- Sublime Text 3
- Notepad++
- GEdit
- Atom
- UltraEdit
- Brackets
- Vim
- ...

IDE

Integrated Development Environment

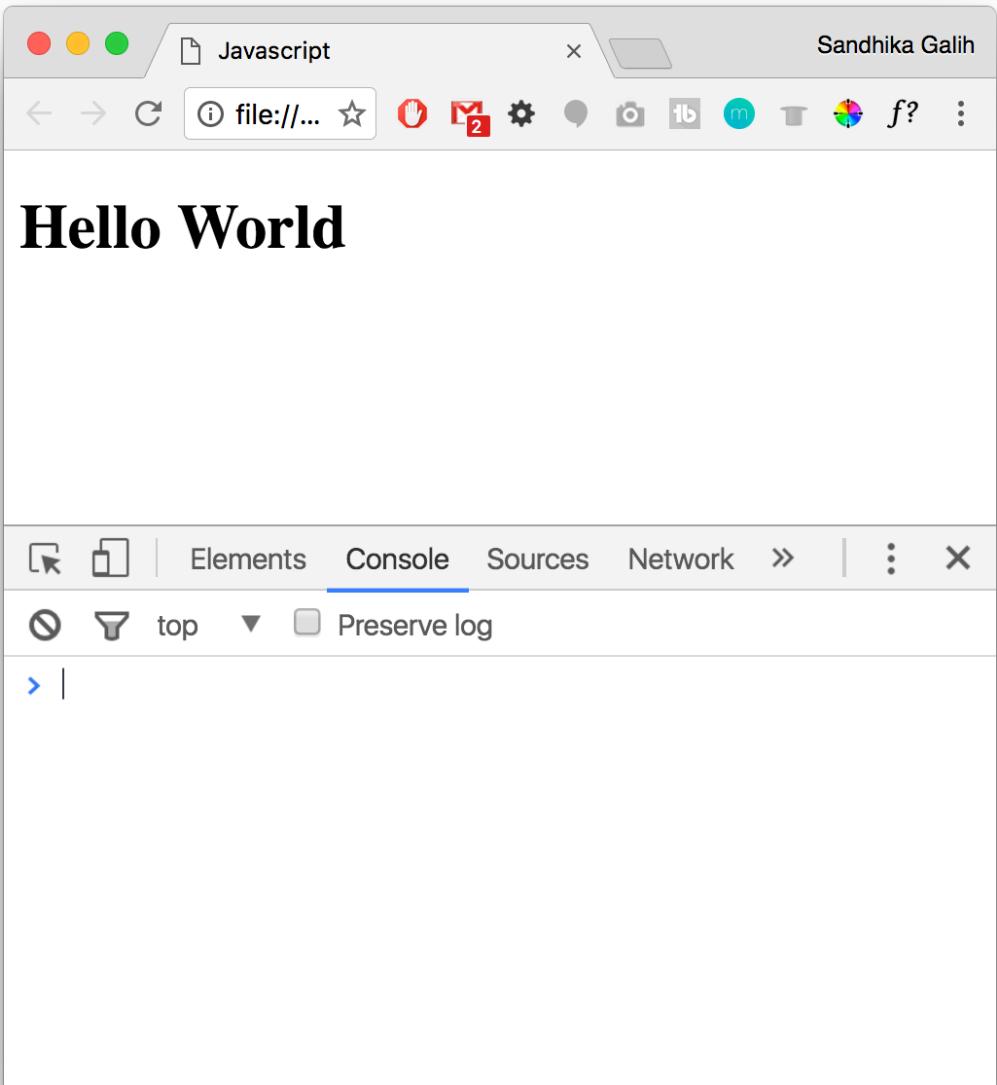


IDE

- XCode
- NetBeans
- Eclipse
- Dreamweaver
- Visual Studio
- PHP Storm
- ...

Kenapa Javascript ?

- Gratis
- Tidak perlu instalasi
- Hanya butuh web browser

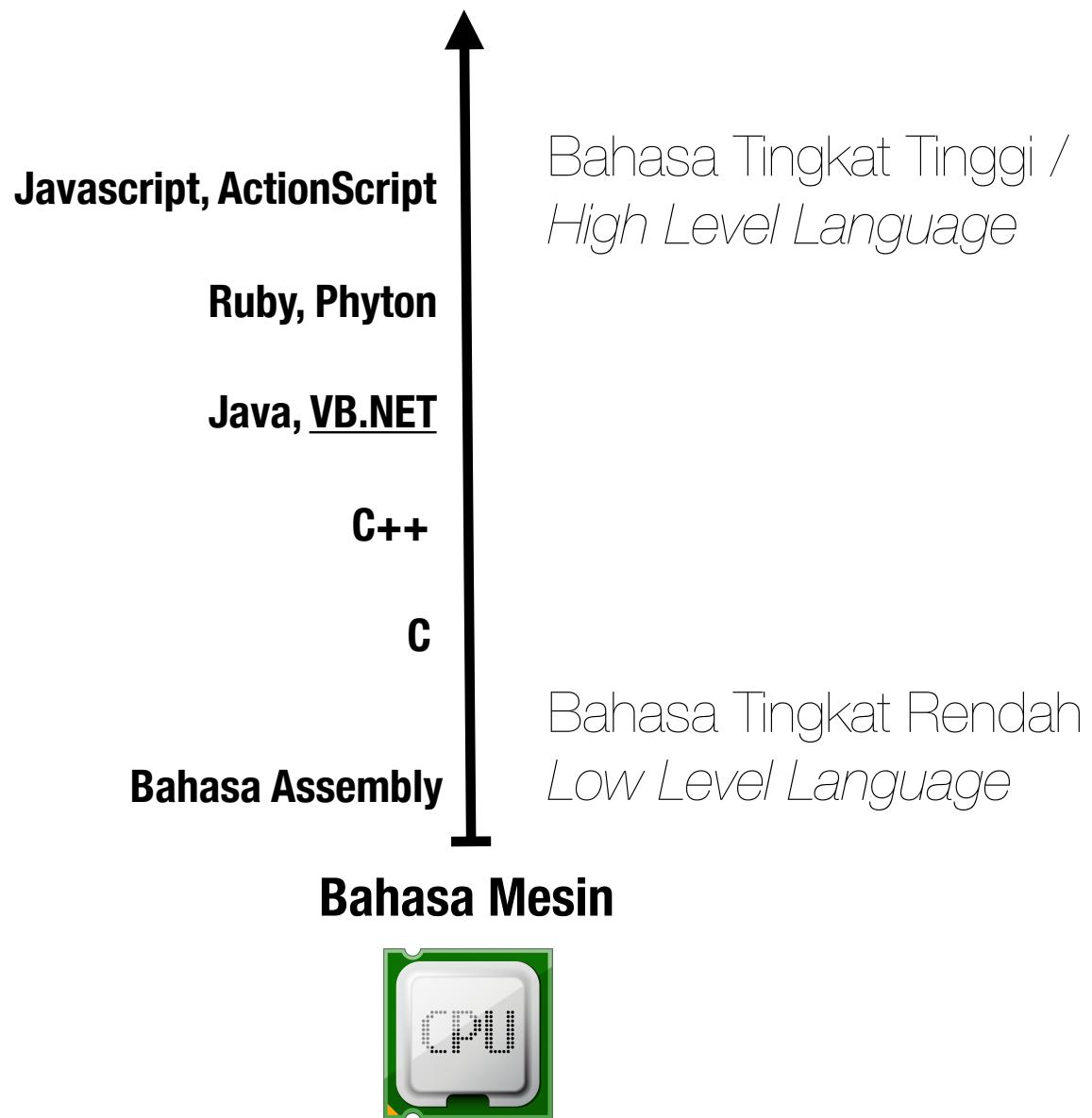


“
JavaScript is a **high-level**, **scripting**, **untyped**,
and **interpreted** programming language.
”

<https://en.wikipedia.org/wiki/JavaScript>

High-level Language

Bahasa Manusia



scripting

Sistem Operasi

Sistem Operasi

C++

Java

Flash

MS Office

Web Browser

C#

Obj-C

ActionScript

VBScript

JavaScript

untyped

untyped / dynamically
typed

untyped

```
var x = 12345;  
x = "hello world!";  
x = { hello: "world!" };
```



typed

```
int x = 12345;  
x = "hello world!";
```



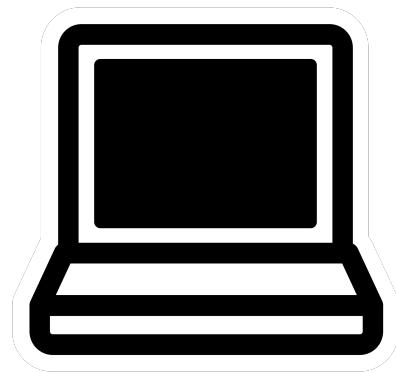
Interpreted

Compiler vs. Interpreter

compile



0	1	0
1	0	0
0	1	1
	1	1
1	1	0
0	0	0
	0	1
1	0	0



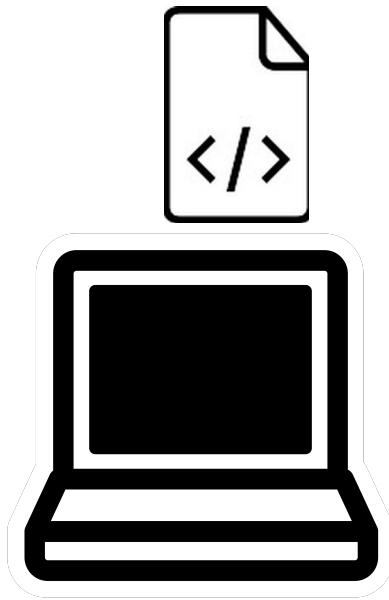
A

executable



B

Compiler



A



B

Interpreter

Compiler vs. Interpreter

Compiler		Interpreter	
siap untuk dijalankan	tidak lintas platform	lintas platform	butuh interpreter
lebih cepat	tidak fleksibel	fleksibel	lebih lambat
source code tidak terlihat	langkah tambahan	mudah untuk diuji	source code mudah diakses

Contoh Bahasa

compiler C, C++, Objective-C

interpreter javascript, PHP

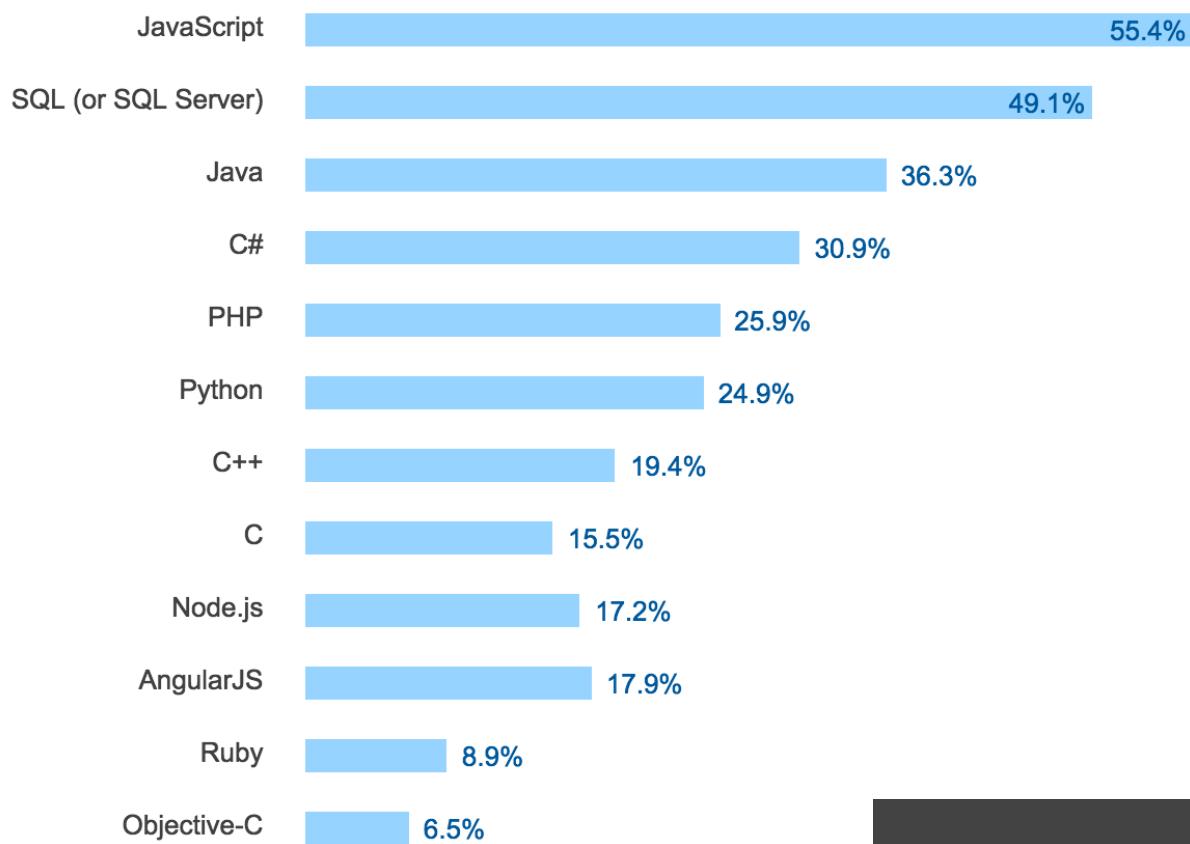
hybrid java, VB.NET, Python

“easy to learn, **hard to master.**”

@mpjme - funfunction

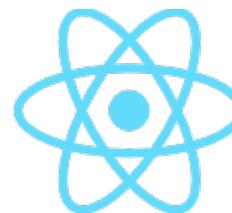
I. Most Popular Technologies

2016 2015 2014 2013

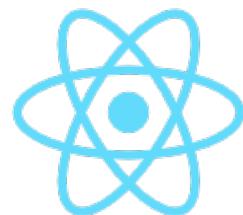


49,397 responses

[stackoverflow**.com](http://stackoverflow.com)**



frameworks & libraries





communities

Javascript bisa apa?

- **Client-side Web Development**

- Native Javascript / *Vanilla Javascript*
- JQuery
- AngularJS, React, Ember, Backbone
- dll.

- **Server-side Development**

- NodeJS
- ExpressJS

- **Browser Extension / Add-on**

- **Desktop Applications**

- Electron, AppJS

- **Mobile App Development**

- JQuery Mobile, Cordova / PhoneGap

- **IoT & Robotics**

- CyclonJS, Johnny-Five

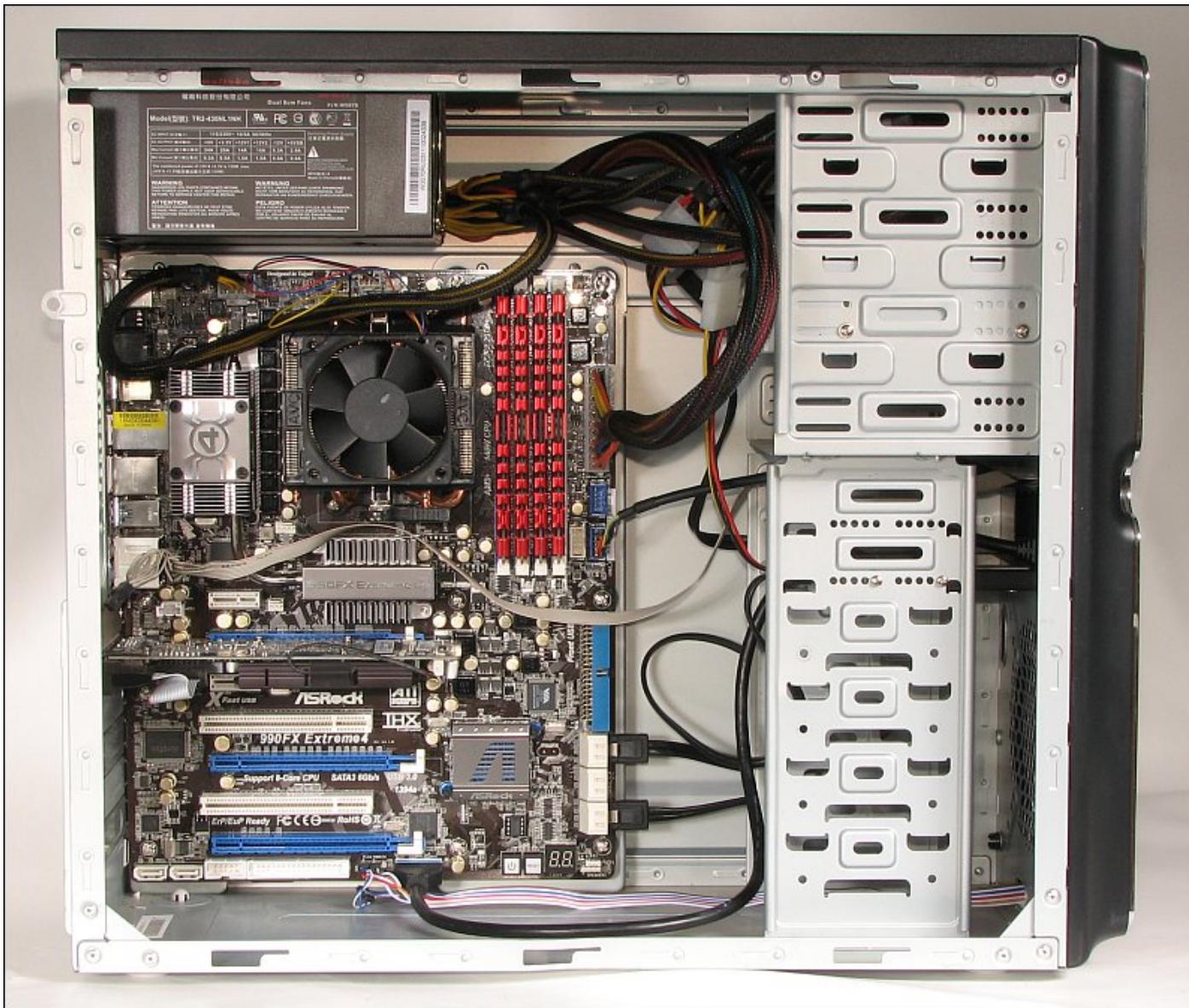
- **JSON**

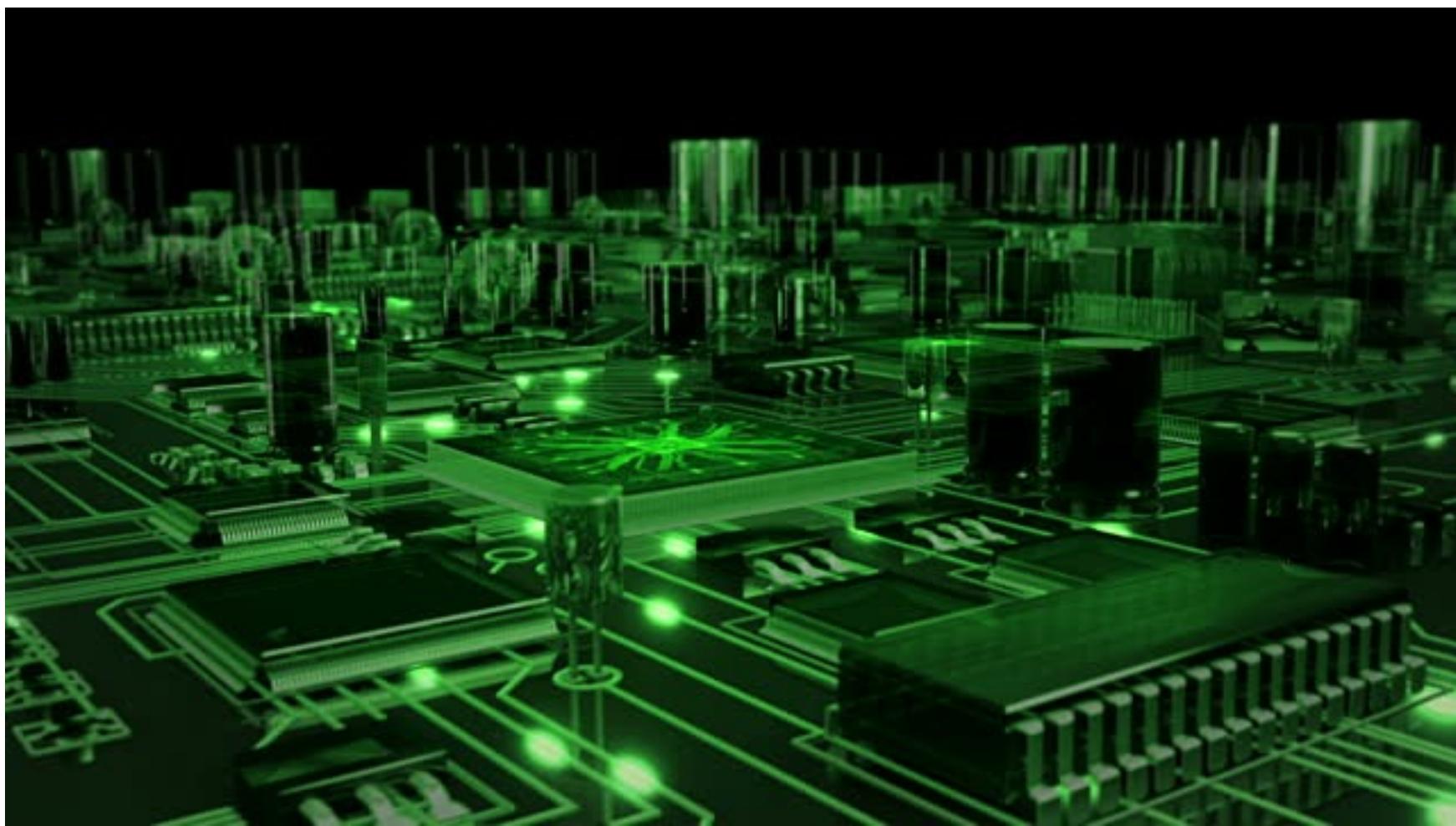
Dasar Pemrograman

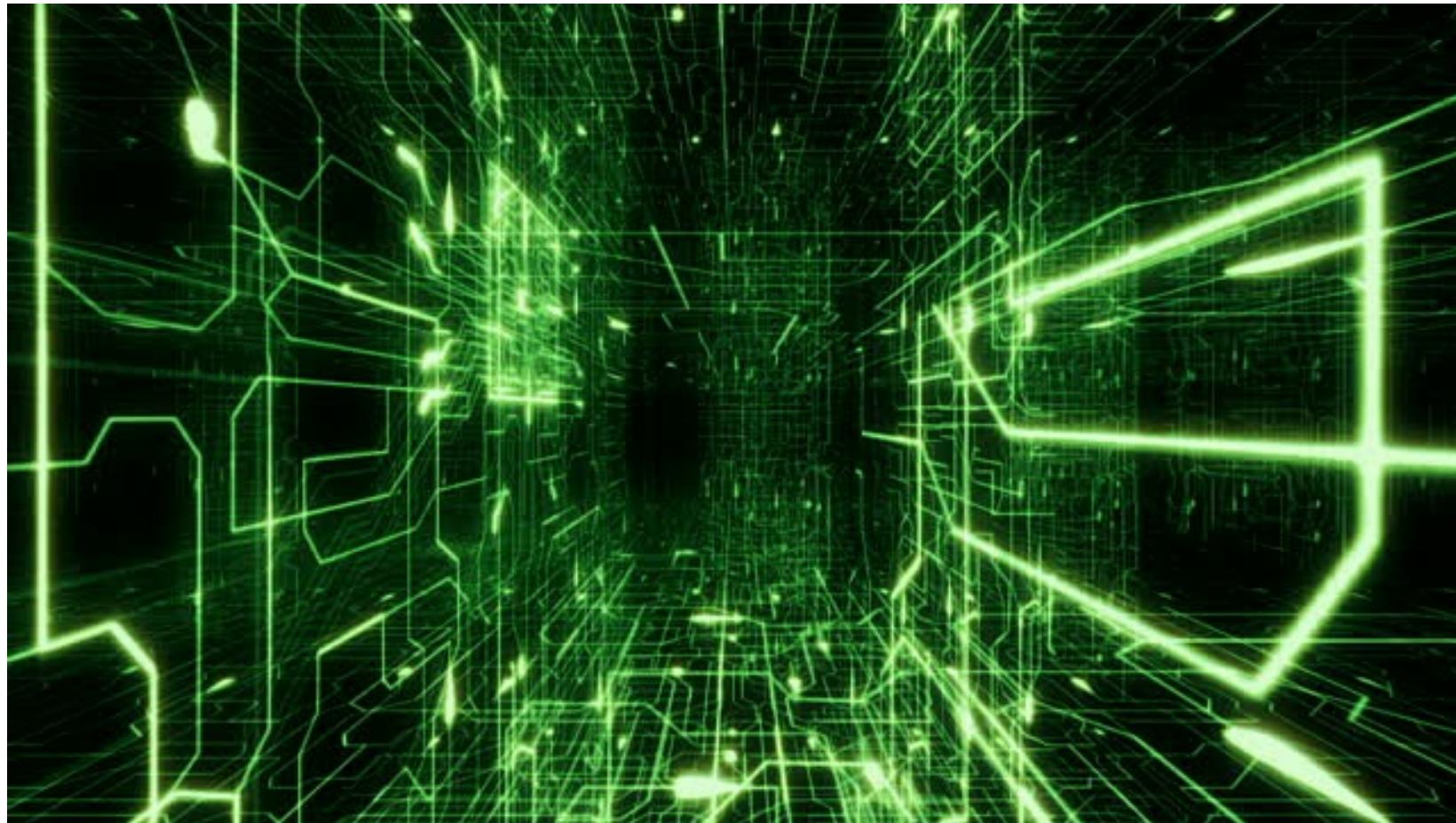
Data













Data

tipe data

tipe data

number

string

boolean

object

function

undefined

Number / Angka

- **Angka tanpa desimal**
 - 10, 1500, 123456
- **Angka dengan desimal**
 - 3.14, 0.5, 100.00
- **Eksponen**
 - 123e5 // 12300000
 - 123e-5 // 0.00123
- **Bilangan negatif**
 - -23, -0.52

String

- ‘Hello World!’
- “Sandhika Galih”

Operator

“ **operator** dalam javascript adalah sebuah **simbol** yang digunakan untuk melakukan **operasi**. ”

operator

aritmatika

penugasan

perbandingan

logika

string

kondisional

operator
aritmatika

+ **-** ***** **/** **%**

modulo / modulus / sisa bagi

operator precedence

urutan operasi matematika

KuKaBaTaKu

() * / + -

operator

penugasan / assignment



operator penugasan

sama saja dengan

x += y

x = x + y

x -= y

x = x - y

x *= y

x = x * y

x /= y

x = x / y

x %= y

x = x % y

operator
perbandingan / comparison

operator perbandingan	keterangan
<code>==</code>	sama dengan
<code>!=</code>	tidak sama dengan
<code>====</code>	strict sama dengan
<code>!==</code>	strict tidak sama dengan
<code>></code>	lebih besar dari
<code><</code>	lebih kecil dari
<code>>=</code>	lebih besar sama dengan
<code><=</code>	lebih kecil sama dengan

menghasilkan
boolean

true false

operator

logika / *logical*

& &

||

!

operator logika	keterangan
&&	AND
	OR
!	NOT

operator
string

+

Variabel

“ sebuah ‘**tempat / wadah**’ yang memiliki nama, yang digunakan untuk menyimpan **nilai** ,”



var x;

x = 20;

var ga boleh pake spasi;

var 1gaboleh;

var boleh1;

var ini_juga_boleh;

var boleh\$;

var \$\$;

var iniBolehBanget;

```
var namaVariabelPanjang;  
// Camel Case
```

keyword & reserved word

keyword & reserved word

**break case catch class const continue
debugger default delete do else enum
export extends false finally for
function if implements import in
instanceof interface let new null
package private protected public return
static super switch this throw true try
typeof var void while with yield**

shorthand

```
var nama;  
var umur;  
var lulus;
```

```
nama = "Sandhika";  
umur = 31;  
lulus = true;
```

```
var nama = "Sandhika",  
umur = 31,  
lulus = true;
```

Menulis Javascript

```
<script></script>
```

internal

external

```
console.log();
```

Popup

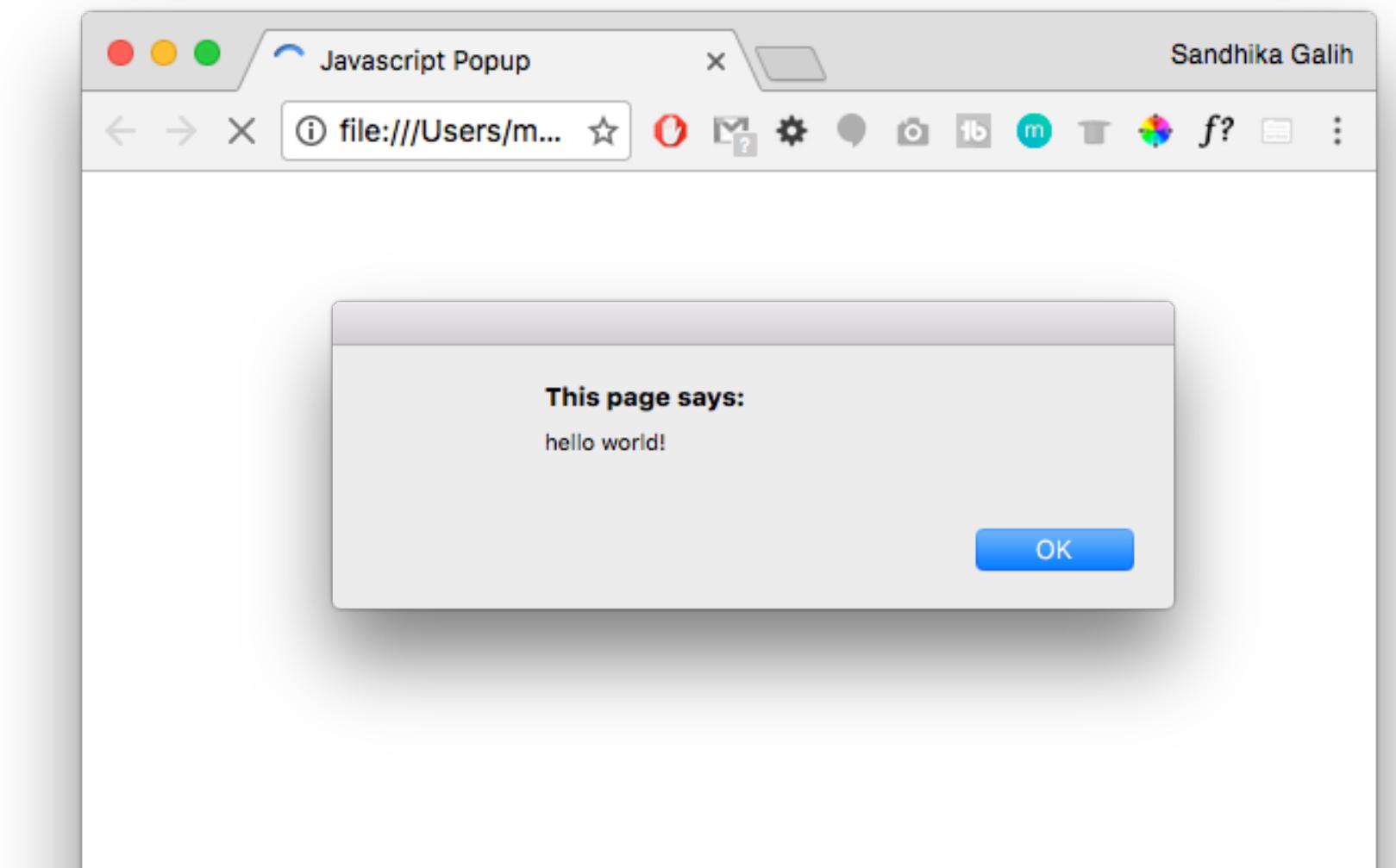
alert

prompt

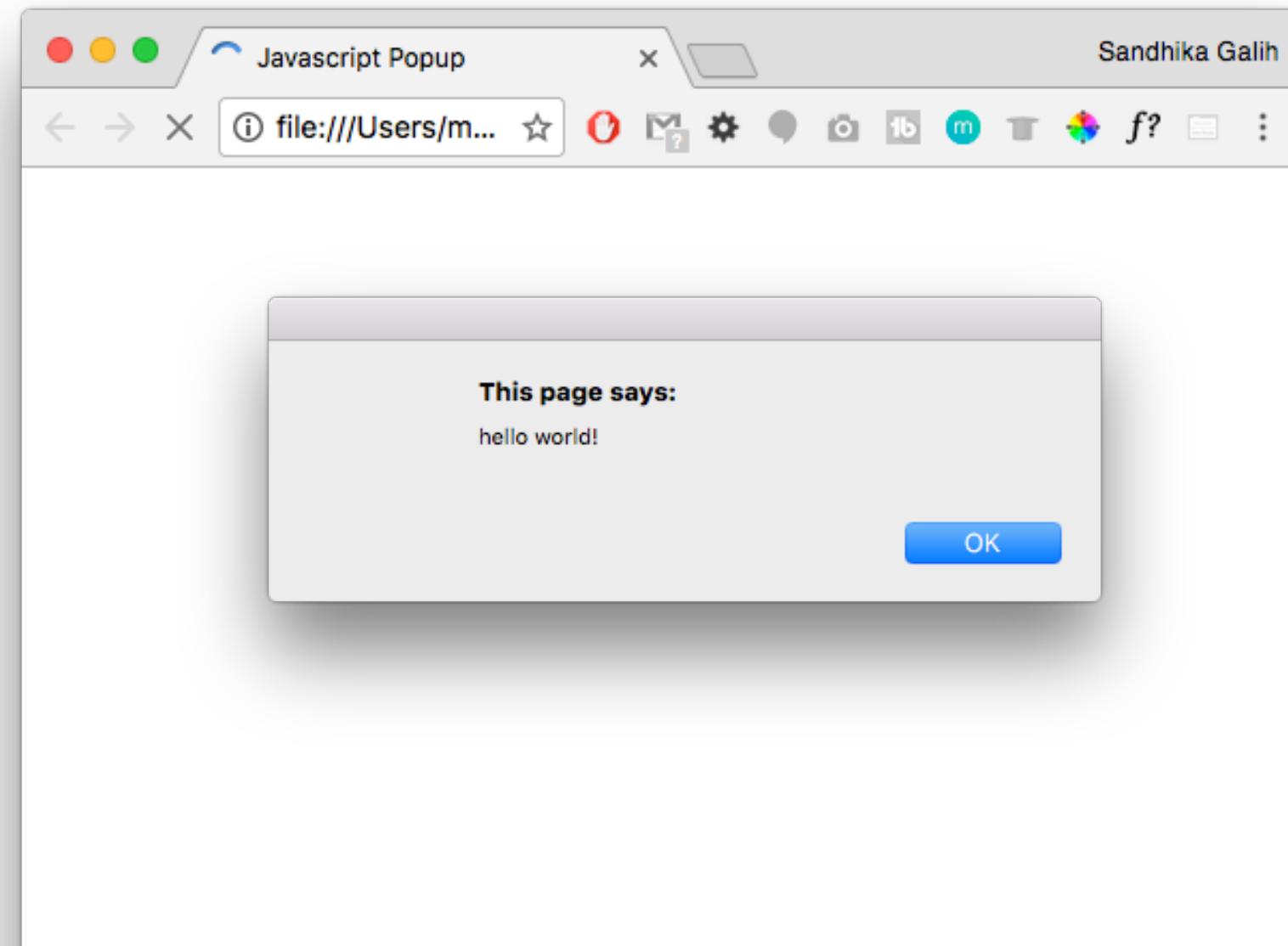
confirm

alert

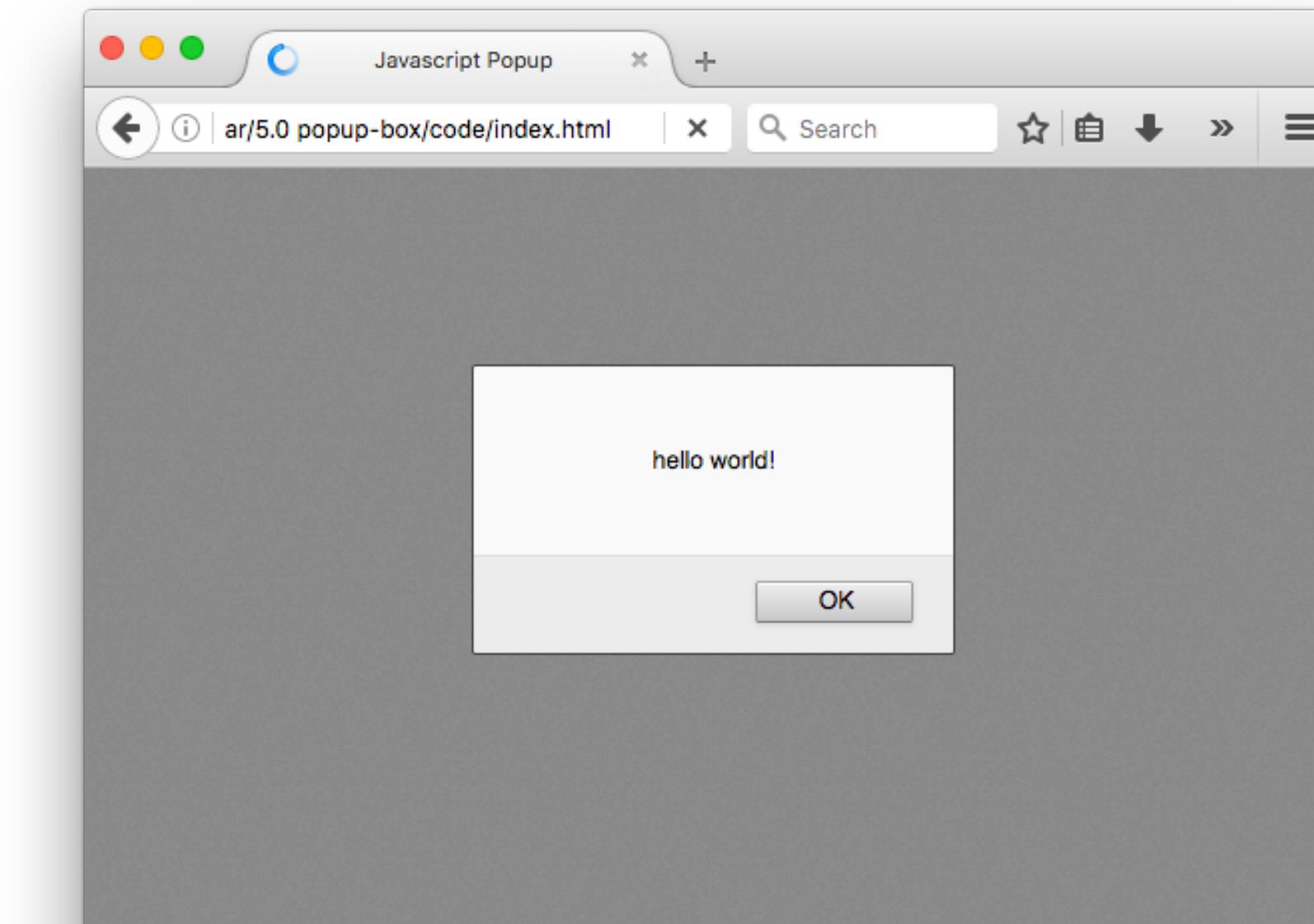
```
alert( 'hello world!' );
```



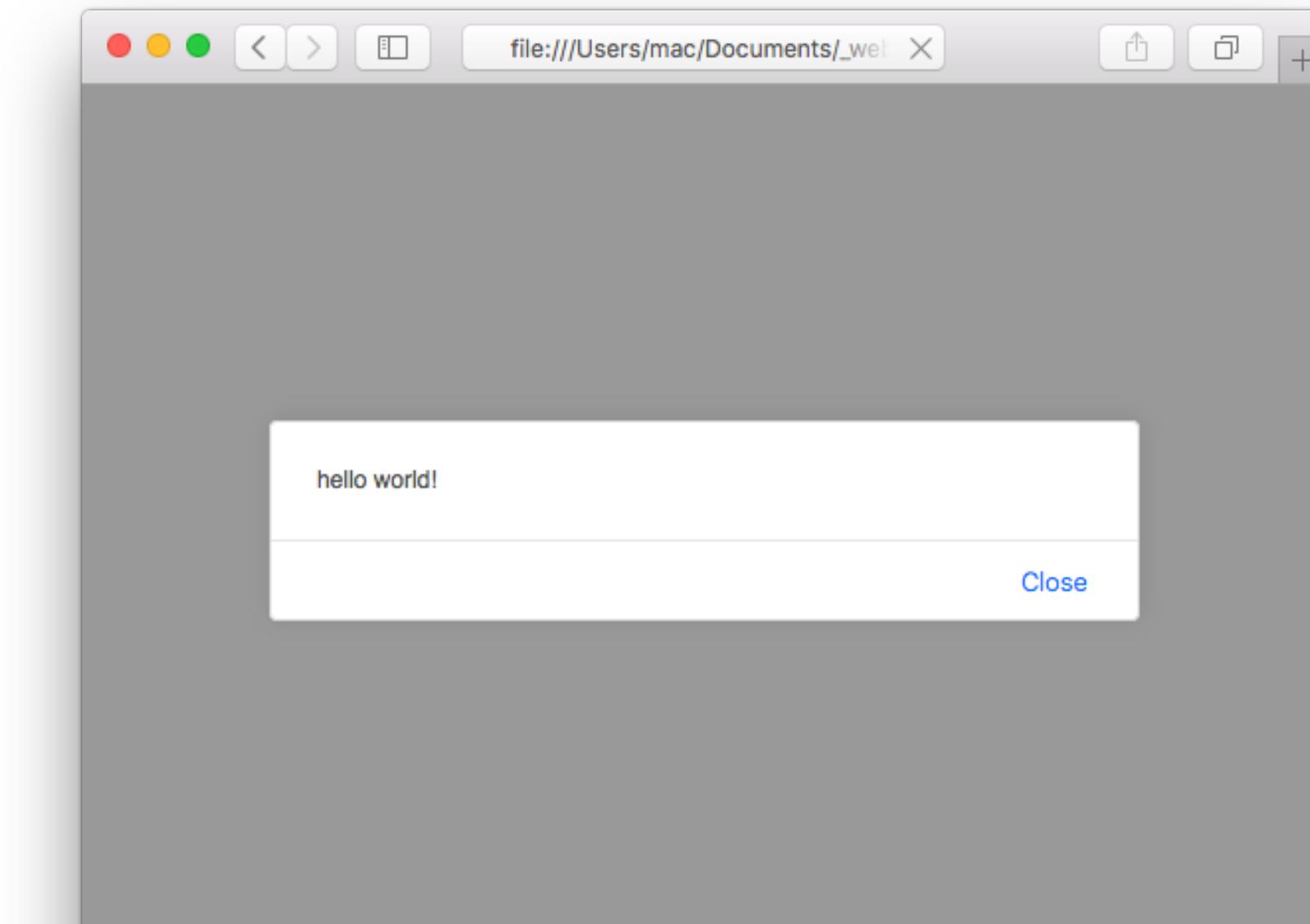
Google Chrome



Mozilla Firefox

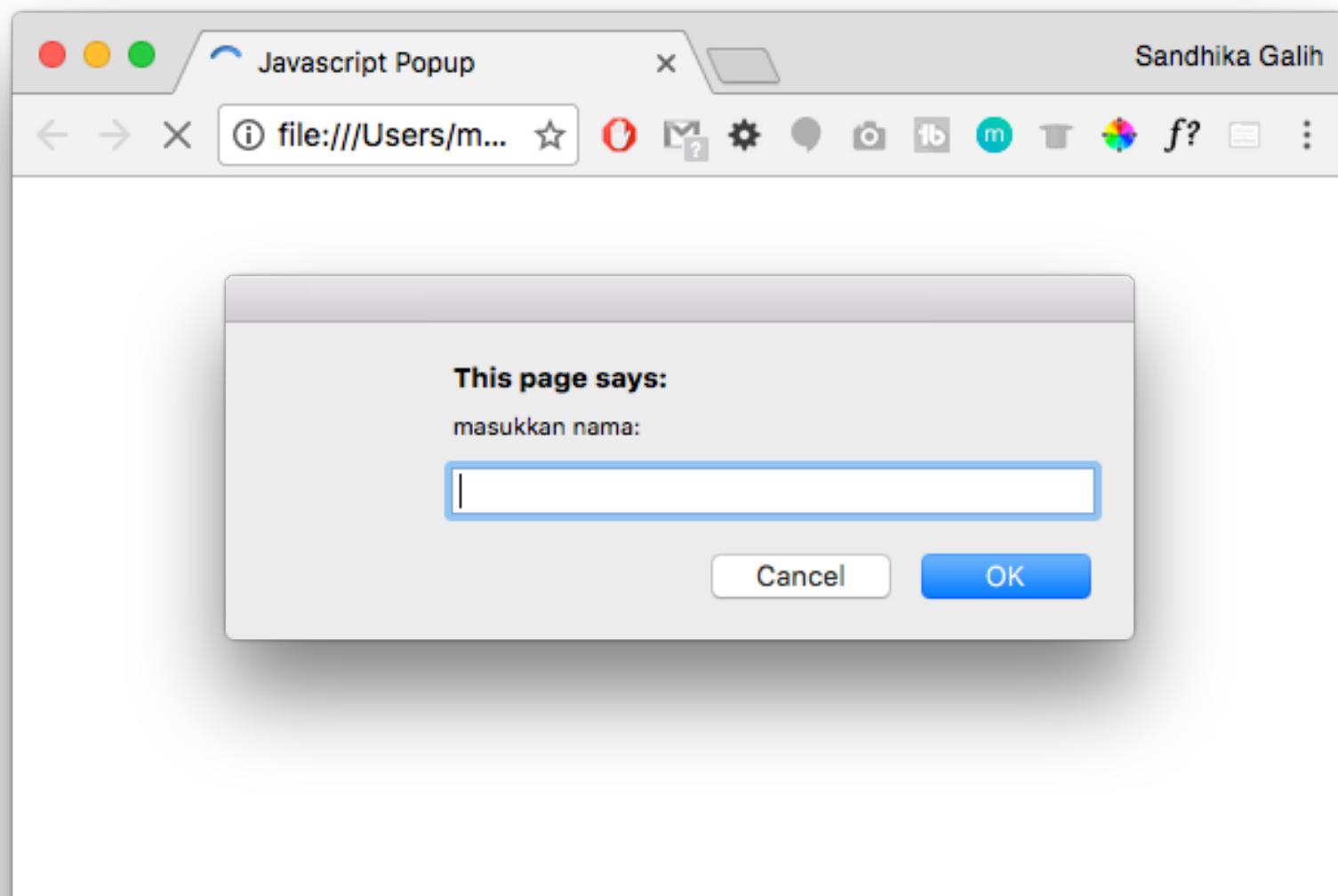


Safari



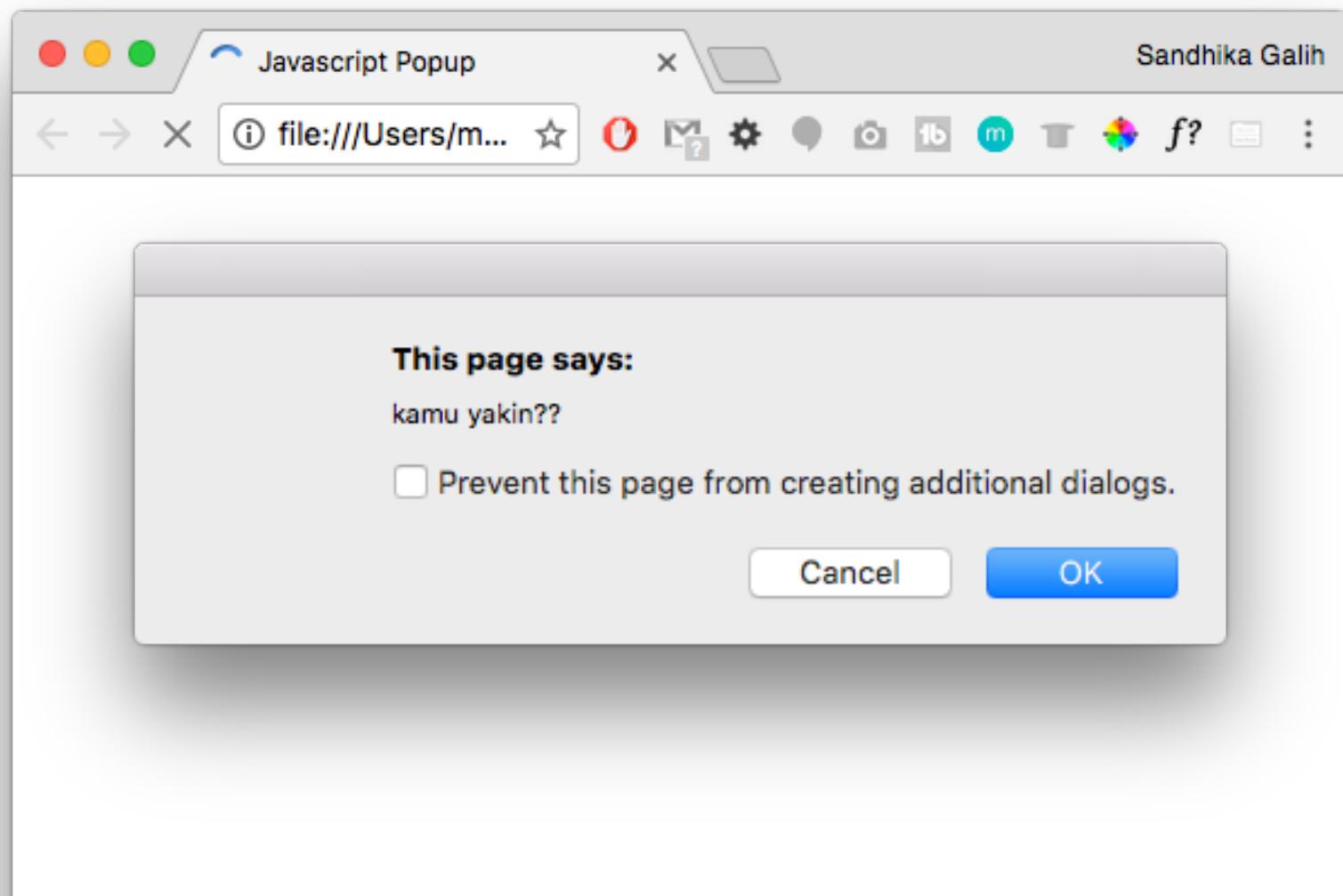
prompt

```
prompt( 'masukkan nama:' );
```



confirm

confirm('kamu yakin??');



Control Flow

alur kendali program

Normal Flow

“ apabila program kita memiliki **lebih dari satu statement**, maka statement tersebut akan **dieksekusi** dari **atas ke bawah** atau dari **kiri ke kanan**. ”

```
alert('Halo');  
alert('nama');  
alert('saya');  
alert('Sandhika');  
alert('Galih');
```

```
var angka = prompt('Masukkan angka : ');
alert('Angka yang anda masukkan adalah : ' + angka);
```

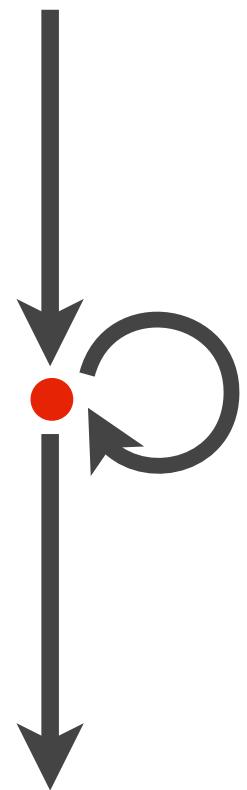


Pengulangan
Control Flow
Pengkondisian

Pengulangan

loop / iteration





Pengulangan

for

while

do while

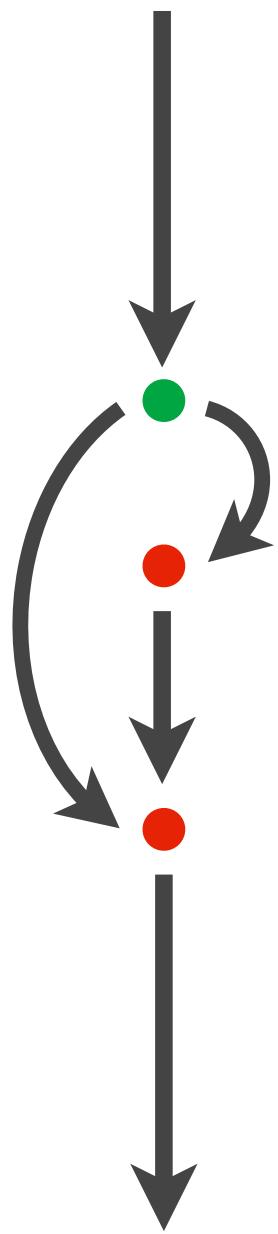
```
alert('Mulai');
alert('Hello World!');
alert('Hello World!');
alert('Hello World!');
alert('Hello World!');
alert('Hello World!');
alert('Selesai');
```

```
alert('Mulai');

for(var i = 0; i < 5; i++) {
    alert('Hello World!');
}

alert('Selesai');
```

Pengkondisian Percabangan



Pengkondisian

if

if... else

if... else if... else

switch

```
var angka = prompt('Masukkan angka :');

if( angka % 2 === 0 ) {
    alert(angka + ' adalah bilangan GENAP');

} else {
    alert(angka + ' adalah bilangan GANJIL');
}
```

While

Developer Tools - file:///Users/mac/Documents/_webprogrammingun...

Elements Console » :

Preserve log

Hello World! script.js:1
Hello World! script.js:2
Hello World! script.js:3
Hello World! script.js:4
Hello World! script.js:5
Hello World! script.js:6
Hello World! script.js:7
Hello World! script.js:8
Hello World! script.js:9
Hello World! script.js:10

>

⋮ Console X

A screenshot of a code editor window titled "script.js — code". The window has three tabs: "while.html", "script.js", and an unregistered tab. The "script.js" tab is active, displaying the following code:

```
1 console.log('Hello World!');  
2 console.log('Hello World!');  
3 console.log('Hello World!');  
4 console.log('Hello World!');  
5 console.log('Hello World!');  
6 console.log('Hello World!');  
7 console.log('Hello World!');  
8 console.log('Hello World!');  
9 console.log('Hello World!');  
10 console.log('Hello World!');
```

The code consists of a series of ten identical statements, each using the `console.log` function to output the string "Hello World!". The line numbers 1 through 10 are displayed before each statement. The last line, "10 console.log('Hello World!');", is highlighted with a light gray background.

At the bottom of the editor, the status bar shows "Line 10, Column 29", "Tab Size: 4", and "JavaScript".

while

selama kondisi
bernilai **true**



while(kondisi) {

aksi ↪

lakukan **aksi**
dalam blok {}

}

```
while(true) {  
    console.log('hello world');  
}
```

Infinite Loop /
looping forever

dihentikan oleh **user**

```
var ulang = true;  
while(ulang) {  
    console.log('hello world');  
    ulang = confirm('lagi?');  
}
```

dihentikan oleh **program**

nilai awal

while(kondisi terminasi) {

aksi

increment / decrement

}

```
var nilaiAwal = 1;  
while(nilaiAwal <= 5) {  
    console.log('hello world');  
    nilaiAwal++;  
}
```

Latihan

JURAGAN ANGKOT



vector by: @aditverdee

1



2



3



4



5



6



7



8



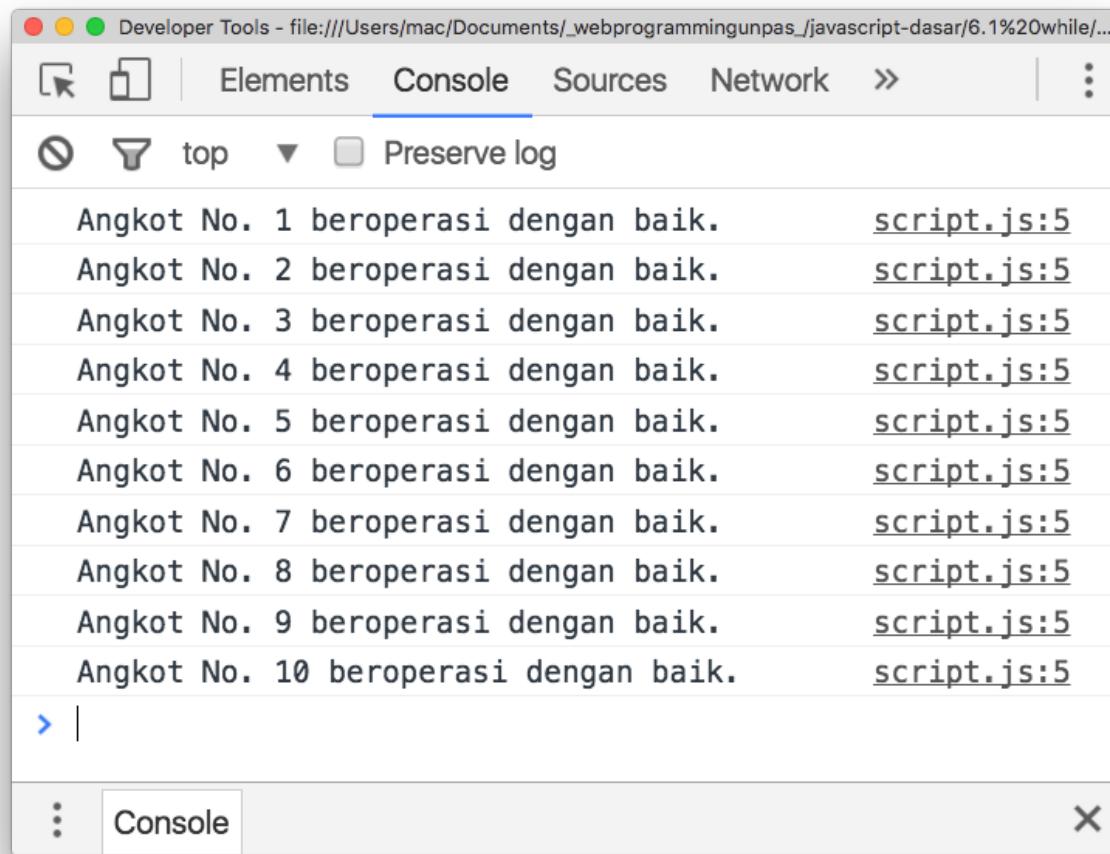
9



10



angkot1.js



The screenshot shows the Developer Tools Console tab with the following log output:

```
Angkot No. 1 beroperasi dengan baik.          script.js:5
Angkot No. 2 beroperasi dengan baik.          script.js:5
Angkot No. 3 beroperasi dengan baik.          script.js:5
Angkot No. 4 beroperasi dengan baik.          script.js:5
Angkot No. 5 beroperasi dengan baik.          script.js:5
Angkot No. 6 beroperasi dengan baik.          script.js:5
Angkot No. 7 beroperasi dengan baik.          script.js:5
Angkot No. 8 beroperasi dengan baik.          script.js:5
Angkot No. 9 beroperasi dengan baik.          script.js:5
Angkot No. 10 beroperasi dengan baik.         script.js:5
```

Buat program javascript untuk menampilkan hasil diatas!

Mari kita telusuri..

```
var nilaiAwal = 1;
while(nilaiAwal <= 10) {
    console.log('...
        nilaiAwal++);
}
```

sebelum mulai loop,
tentukan nilai awal

var nilaiAwal = 1;

ulangi terus sampai
kondisi false

nilaiAwal <= 10 ?

cek kondisi apakah
bernilai true atau false

console.log(' ...

lakukan hal ini jika
kondisi bernilai true

nilaiAwal++;

increment nilai awal
di akhir tiap
pengulangan

Perbaiki program

```
var jmlAngkot = 10;  
var noAngkot = 1;  
while(noAngkot <= jmlAngkot) {  
    console.log('...  
        noAngkot++;  
}
```

```

var jmlAngkot = 10;
var noAngkot = 1;

while( noAngkot <= jmlAngkot ) {
    console.log('Angkot No. ' + noAngkot + ' beroperasi dengan baik.');
    noAngkot++;
}

```

noAngkot	noAngkot <= jmlAngkot ?	Output
1	TRUE	Angkot No. 1 beroperasi dengan baik
2	TRUE	Angkot No. 2 beroperasi dengan baik
3	TRUE	Angkot No. 3 beroperasi dengan baik
...
10	TRUE	Angkot No. 10 beroperasi dengan baik
11	FALSE	KELUAR DARI LOOP!

for

```
var nilaiAwal = 1;
while(nilaiAwal <= 10) {
    console.log('hello world');
    nilaiAwal++;
}
```

```
for( var nilaiAwal = 1; nilaiAwal <= 10; nilaiAwal++ ) {  
    console.log('hello world');  
}
```

```
for( var nilaiAwal = 1; nilaiAwal <= 10; nilaiAwal++ ) {  
    console.log('hello world');  
}
```

latihan for

1



2



3



4



5



6



7



8



9



10



angkot1.js

1



2



3



4



5



6



7



8



9



10



angkot2.js

while

Developer Tools - file:///Users/mac/Documents/_webprogrammingunpas_.javascript-dasar/6.1%20while/...

Elements Console Sources Network ::

top Preserve log

Angkot No. 1 beroperasi dengan baik.	angkot2.js:6
Angkot No. 2 beroperasi dengan baik.	angkot2.js:6
Angkot No. 3 beroperasi dengan baik.	angkot2.js:6
Angkot No. 4 beroperasi dengan baik.	angkot2.js:6
Angkot No. 5 beroperasi dengan baik.	angkot2.js:6
Angkot No. 6 beroperasi dengan baik.	angkot2.js:6
Angkot No. 7 sedang tidak beroperasi.	angkot2.js:11
Angkot No. 8 sedang tidak beroperasi.	angkot2.js:11
Angkot No. 9 sedang tidak beroperasi.	angkot2.js:11
Angkot No. 10 sedang tidak beroperasi.	angkot2.js:11

:

Console

for

variabel baru

```
var jmlAngkot = 10;  
var angkotBeroperasi = 6;  
var noAngkot = 1;
```

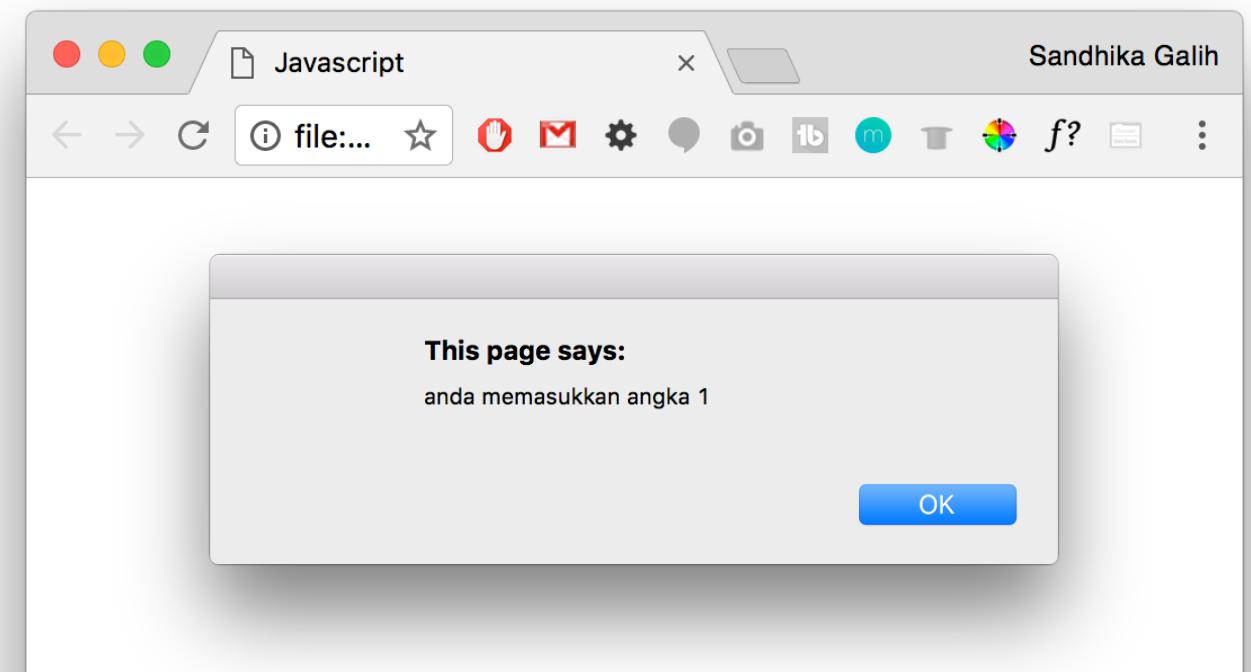


Pengkondisian / Percabangan

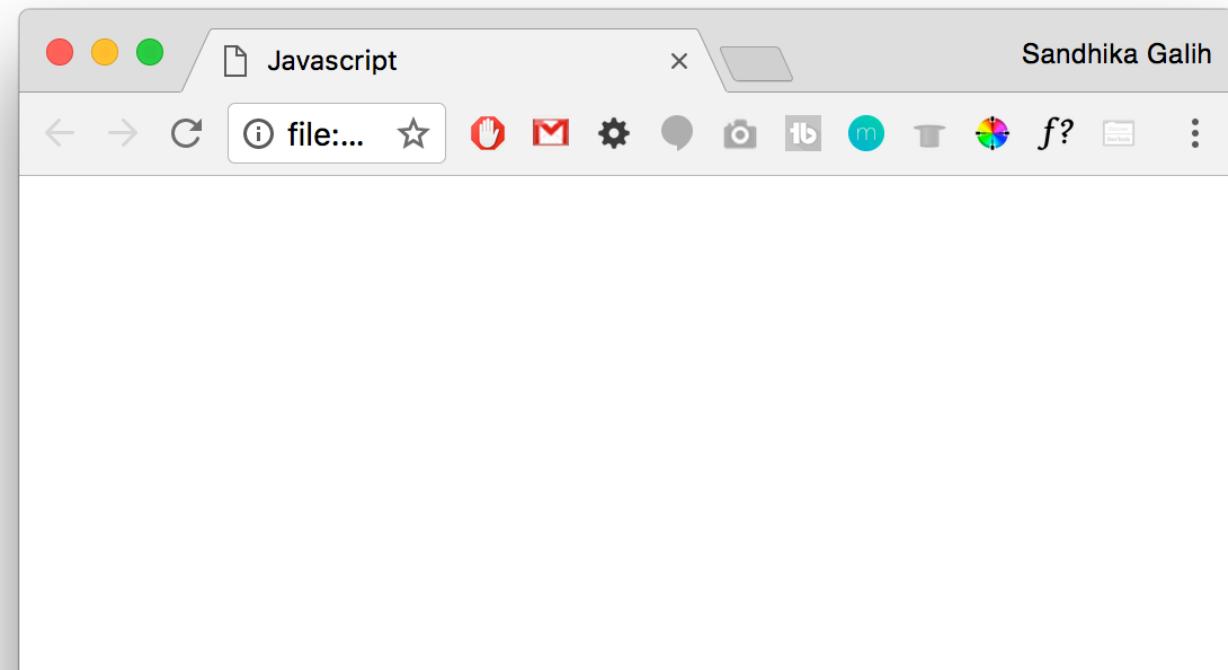
i f

```
if (kondisi) {  
    aksi  
}  
...  
lakukan aksi jika kondisi  
bernilai true  
keluar dari blok jika kondisi  
bernilai false
```

```
var angka = 1;  
  
if(angka === 1) {  
    alert('anda memasukkan angka 1');  
}  
}
```



```
var angka = 2;  
  
if(angka === 1) {  
    alert('anda memasukkan angka 1');  
}  
}
```



```
var angka = 2;  
  
if(angka === 1) {  
    alert('anda memasukkan angka 1');  
}  
}
```

 keluar dari blok jika kondisi
bernilai **false**

```
var angka = 2;  
  
if(angka <= 5) {  
    alert(angka + " nembak ke arah datar");  
}  
else {  
    alert("angka " + angka + " menembak ke atas");  
}
```

```
var angka = 2;  
  
if(angka% 2<= 5) {  
    alert((angka++' adæbiih bliðsíðan dageinAP' ));  
}
```

```
var angka = 3;

if(angka % 2 == 0) {
    alert(angka + ' adalah bilangan GENAP');

}

if(angka % 2 == 1) {
    alert(angka + ' adalah bilangan GANJIL');

}
```

else

```
if(kondisi) {
```

aksi 1

lakukan aksi 1 jika kondisi
bernilai **true**

```
} else {
```

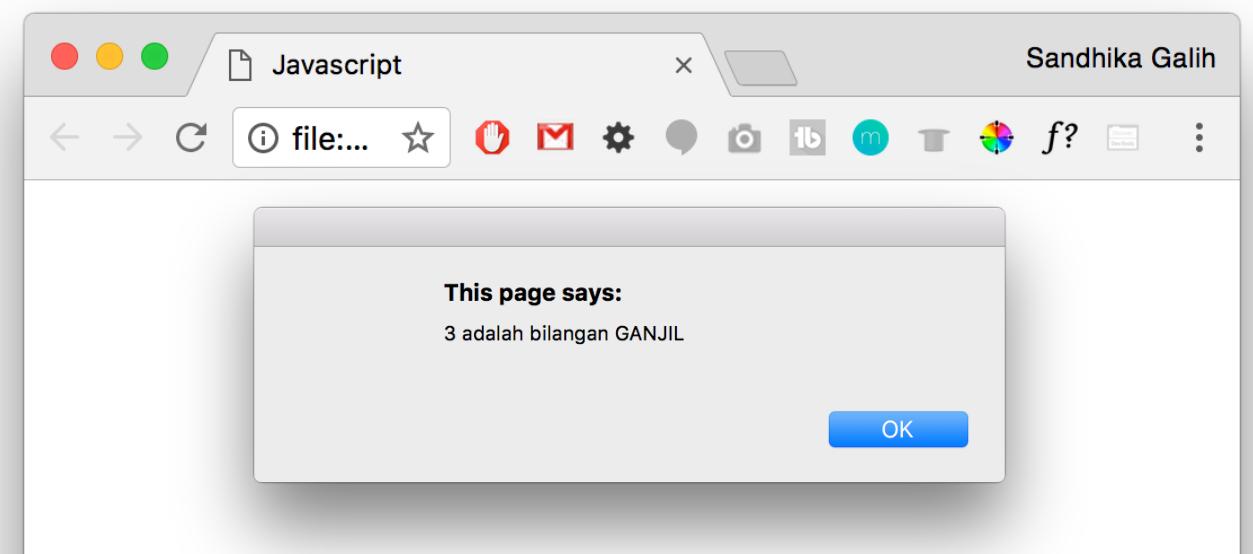
aksi 2

lakukan aksi 2 jika kondisi
bernilai **false**

```
}
```

```
var angka = 3;  
  
if(angka % 2 == 0) {  
    alert(angka + ' adalah bilangan GENAP');  
}  
  
if(angka % 2 == 1) {  
    alert(angka + ' adalah bilangan GANJIL');  
}
```

```
var angka = 3;  
  
if(angka % 2 == 0) {  
    alert(angka + ' adalah bilangan GENAP');  
} else {  
    alert(angka + ' adalah bilangan GANJIL');  
}
```



```
var angka= prompt ('masukkan angka:');

if(angka % 2 == 0) {
    alert(angka + ' adalah bilangan GENAP');
} else {
    alert(angka + ' adalah bilangan GANJIL');
}
```

latihan if else

1



2



3



4



5



6



7



8



9



10



angkot3.js

```
for (let i = 1; i <= 10; i++) {
    console.log(`Angkot No. ${i} beroperasi dengan baik.`);
    if (i === 7) {
        console.log(`Angkot No. ${i} sedang tidak beroperasi.`);
    }
}
```

Developer Tools - file:///Users/mac/Documents/_webprogrammingunpas_/javascript-dasar/6.1%20while/...

Elements Console Sources Network Preset log

top Preserve log

Angkot No. 1 beroperasi dengan baik.
Angkot No. 2 beroperasi dengan baik.
Angkot No. 3 beroperasi dengan baik.
Angkot No. 4 beroperasi dengan baik.
Angkot No. 5 beroperasi dengan baik.
Angkot No. 6 beroperasi dengan baik.
Angkot No. 7 sedang tidak beroperasi.
Angkot No. 8 sedang tidak beroperasi.
Angkot No. 9 sedang tidak beroperasi.
Angkot No. 10 sedang tidak beroperasi.

angkot2.js:6
angkot2.js:6
angkot2.js:6
angkot2.js:6
angkot2.js:6
angkot2.js:6
angkot2.js:6
angkot2.js:11
angkot2.js:11
angkot2.js:11
angkot2.js:11

gunakan hanya 1 for saja!

A screenshot of a browser's developer tools Console tab. The tab title is "Developer Tools - file:///Users/mac/Documents/_webprogrammingunpas_ja". The Console tab is active, showing the following log entries:

```
Angkot No. 1 beroperasi dengan baik.  
Angkot No. 2 beroperasi dengan baik.  
Angkot No. 3 beroperasi dengan baik.  
Angkot No. 4 beroperasi dengan baik.  
Angkot No. 5 beroperasi dengan baik.  
Angkot No. 6 beroperasi dengan baik.  
Angkot No. 7 sedang tidak beroperasi.  
Angkot No. 8 sedang tidak beroperasi.  
Angkot No. 9 sedang tidak beroperasi.  
Angkot No. 10 sedang tidak beroperasi.
```

The first six entries are highlighted with a red rectangular box and labeled "noAngkot 1-6" with a blue arrow pointing to the box. The last four entries are highlighted with a blue rectangular box and labeled "noAngkot 7-10" with a blue arrow pointing to the box. A red curved arrow points from the word "for" on the left towards the red box. The bottom of the screenshot shows a "Console" button.



else if

```
if(kondisi) {  
    aksi 1  
} else {  
    aksi 2  
}
```

```
if(kondisi) {  
    aksi 1  
}
```

```
else {  
    aksi 2  
}
```

```
if(kondisi 1) {  
    aksi 1  
}
```

```
else {  
    aksi 2  
}
```

```
if(kondisi 1) {  
    aksi 1  
} else if(kondisi 2) {  
    aksi 2  
} else {  
    aksi 2  
}
```

```
if(kondisi 1) {  
    aksi 1  
} else if(kondisi 2) {  
    aksi 2  
} else {  
    aksi 3  
}
```

```
var angka = prompt('masukkan angka:');

if(angka % 2 == 0) {
    alert(angka + ' adalah bilangan GENAP');
} else if (angka % 2 == 1) {
    alert(angka + ' adalah bilangan GANJIL');
} else {
    alert('yang anda masukkan bukan angka!');
}
```

latihan

1



2



3



4



5



6



7



8



9



10



angkot4.js

1



2



3



4



5



6



7



8



9



10



angkot4.js

The screenshot shows a developer tools window with the title "Developer Tools - file:///Users/sandhikagalih/Documents/_w". The "Console" tab is selected. The log area displays the following messages:

- Angkot No. 1 beroperasi dengan baik.
- Angkot No. 2 beroperasi dengan baik.
- Angkot No. 3 beroperasi dengan baik.
- Angkot No. 4 beroperasi dengan baik.
- Angkot No. 5 beroperasi dengan baik.
- Angkot No. 6 beroperasi dengan baik.
- Angkot No. 7 sedang tidak beroperasi.
- Angkot No. 8 sedang lembur.** (This line is highlighted with a red rectangle.)
- Angkot No. 9 sedang tidak beroperasi.
- Angkot No. 10 sedang tidak beroperasi.

A blue arrow icon is located at the bottom left of the console area.



1



2



3



4



5



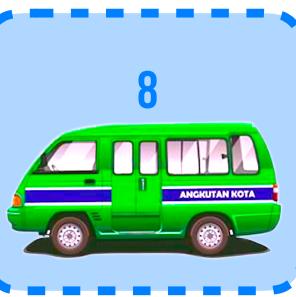
6



7



8



9



10



angkot5.js

- Angkot No. 1 beroperasi dengan baik.
- Angkot No. 2 beroperasi dengan baik.
- Angkot No. 3 beroperasi dengan baik.
- Angkot No. 4 beroperasi dengan baik.
- Angkot No. 5 beroperasi dengan baik.
- Angkot No. 6 beroperasi dengan baik.
- Angkot No. 7 sedang tidak dapat beroperasi.
- Angkot No. 8 sedang lembur.
- Angkot No. 9 sedang tidak dapat beroperasi.
- Angkot No. 10 sedang lembur.





angkot6.js

Angkot No. 1 beroperasi dengan baik.

Angkot No. 2 beroperasi dengan baik.

Angkot No. 3 beroperasi dengan baik.

Angkot No. 4 beroperasi dengan baik.

Angkot No. 5 sedang lembur.

Angkot No. 6 beroperasi dengan baik.

Angkot No. 7 sedang tidak dapat beroperasi.

Angkot No. 8 sedang lembur.

Angkot No. 9 sedang tidak dapat beroperasi.

Angkot No. 10 sedang lembur.



Switch

```
if(kondisi 1) {  
    aksi 1  
} else if(kondisi 2) {  
    aksi 2  
} else if(kondisi 3) {  
    aksi 3  
} ...  
else if(kondisi n) {  
    aksi n  
} else {  
    aksi default  
}
```

switch

```
switch(ekspresi) {  
    case "nilai 1":  
        aksi 1  
        [break;]  
    case "nilai 2":  
        aksi 2  
        [break;]  
    case "nilai n":  
        aksi n  
        [break;]  
    default:  
        aksi default  
        [break;]  
}
```



terimakasih.
sandhikagalih@unpas.ac.id