

Coding for Teens

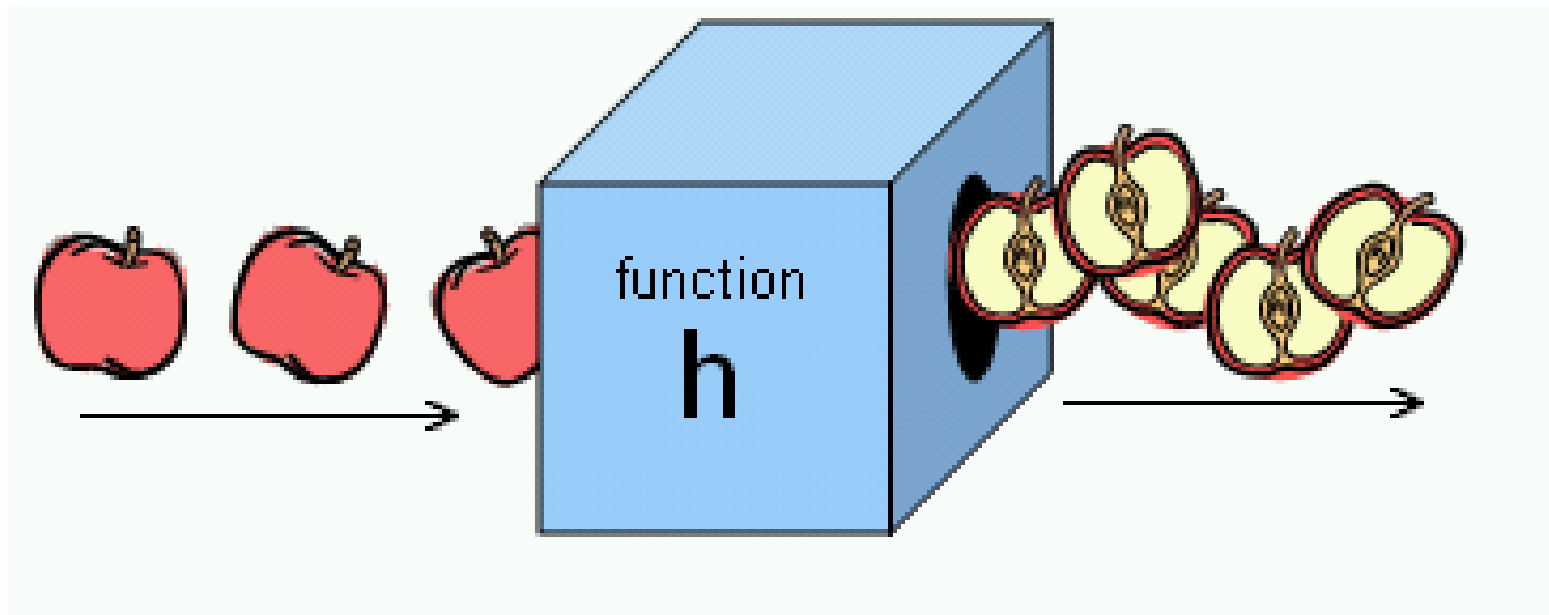
JS

Exploring

#5 Function & Array

Function

Functions are blocks of code that can be named and reused.



Function Statement

```
function contoh() {  
    console.log('Halo Dunia!');  
}  
contoh();
```

```
/*  
function namafunc(param) {prog}  
*/
```

Function Expression

```
var contoh = function () {  
    console.log('Halo Dunia!');  
}  
contoh();
```

```
/*  
var namafunc = function (param) {prog}  
*/
```

Function

```
let x = 10
```

```
let y = 50
```

```
function contoh() {  
    console.log(x+y)  
}
```

```
contoh();
```

Function with a Parameter

```
function namaku(nama) {  
    console.log (nama+'  
Susilo');  
}
```

```
namaku('Adi');  
namaku('Budi');  
namaku('Caca');  
namaku('Dedi');
```

Function with 2 Parameters

```
function data(x,y) {  
    console.log (x+' Lahir th '+y);  
}
```

```
data('Adi', '1990');  
data('Budi', '1991');  
data('Caca', '1992');  
data('Dedi', '1993');
```

Return Function

```
function total(x,y) {  
    z = x + y  
    return z  
}  
console.log(total(4,5))  
console.log(z)
```

/*

- z adalah local variabel dalam func total, tidak dapat dipanggil di luar func tsb.
- jika z tidak di-return maka total(4,5) = undefined

*/

Return Function

```
function total(x,y) {  
    z = x + y  
}
```

```
console.log(total(4,5))
```

/*

- z adalah local variabel dalam func total, tidak dapat dipanggil di luar func tsb.

- jika z tidak di-return maka total(4,5) = undefined

*/

Return Function

```
function total(x,y) {  
    z = x + y  
    console.log(z)  
}
```

```
console.log(total(4,5))
```

/*

- z adalah local variabel dalam func total, tidak dapat dipanggil di luar func tsb.

- jika z tidak di-return maka total(4,5) = undefined

*/

Recursive Function

```
function Pangkat(x,y){  
    if (y == 1) {  
        return x;  
    }  
    else {  
        return x*x*Pangkat(x,y-1);  
    }  
}  
console.log(Pangkat(7,2))
```

Fn inside Fn

```
function kali(x) {  
    if (x < 2) {return 1;}  
    else {return (x * tiga());}  
}  
  
function tiga(){  
    return 3;  
}  
  
console.log(kali(5))
```

setTimeout

```
setTimeout(waktu, 3000);
```

```
function waktu() {  
    console.log('Halo');  
}
```

```
console.log('Yuk');
```

3000 ms (3 detik) setelah program running, output 'Halo' muncul. Output 'Yuk' muncul lebih dahulu, tanpa menunggu baris kode di atasnya ('Halo') selesai.

clearTimeout

```
var x = setTimeout(waktu, 3000);  
  
function waktu() {  
    console.log('Halo');  
}  
clearTimeout(x)  
console.log('Yuk');
```

Baris setTimeout tidak diproses, lantaran dibatalkan seketika oleh clearTimeout.

setInterval

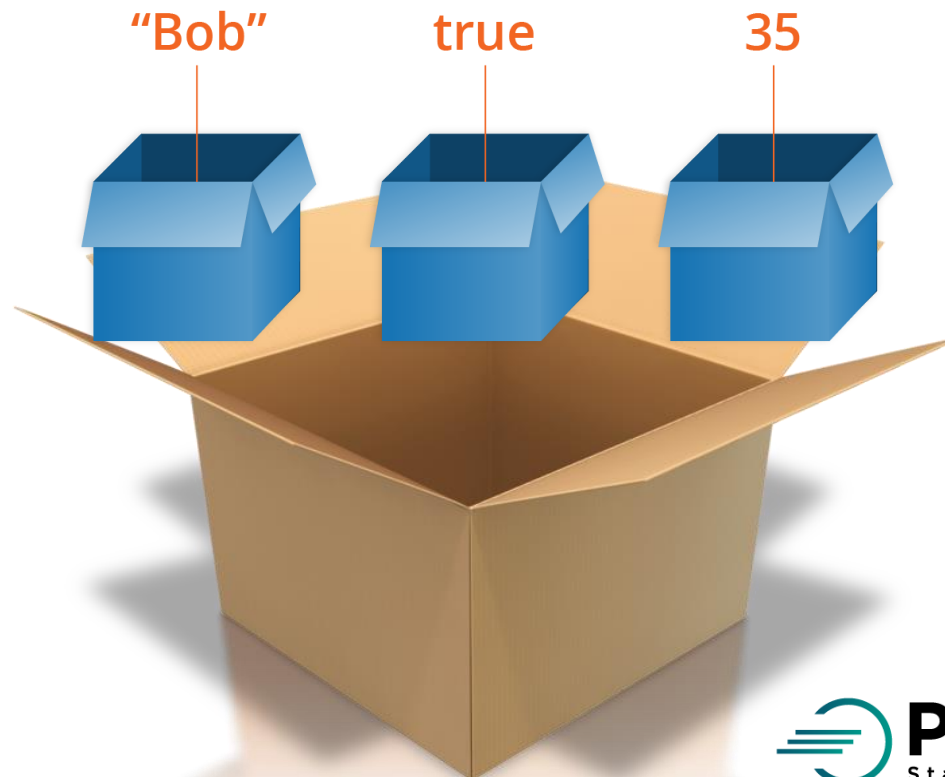
```
setInterval(waktu, 1500);
```

```
function waktu() {  
    console.log('Halo');  
}
```

Output 'Halo' akan muncul tiap 1500 ms (1.5 detik).
Untuk stop proses di cmd, tekan CTRL + C.

Array

Arrays are container-like values that can hold other values. The values inside an array are called elements.




```
let mobil1 = 'Alya';  
let mobil2 = 'Xenia';  
let mobil3 = 'Avanza';
```

Array

=====

```
let mobil =  
['Alya', 'Xenia', 'Avanza'];
```

```
let mobil = [  
  'Alya',  
  'Xenia',  
  'Avanza'  
];
```

Array

```
let mobil = ['Alya', 'Xenia', 'Avanza'];  
console.log(mobil)  
console.log(mobil.toString())  
console.log(mobil.join(' * '))
```

Array Elements

```
let mobil = ['Alya', 'Xenia', 'Avanza'];
```

```
console.log(mobil[0])
```

```
console.log(mobil[1])
```

```
console.log(mobil[2])
```

```
console.log(mobil[3])
```

Array Elements

```
let buah, bPjg, i;
```

```
buah = ['Jeruk', 'Nanas', 'Apel'];
```

```
bPjg = buah.length;
```

```
for (i = 0; i < bPjg; i++) {  
    console.log(buah[i]);  
}
```

Array Properties & Methods

< length, sort, reverse & indexOf >

```
let mobil = ['Alya', 'Xenia', 'Avanza'];
```

```
let w = mobil.length;
```

```
let x = mobil.sort();
```

```
//let y = mobil.reverse();
```

```
let z = mobil.indexOf('Avanza');
```

```
console.log(w)
```

```
console.log(x)
```

```
// console.log(y)
```

```
console.log(z)
```

pop & push

```
let buah = ['Jeruk', 'Nanas', 'Apel'];
```

```
buah.pop();  
console.log(buah)
```

```
buah.push('Kiwi');  
console.log(buah)
```

shift & unshift

```
let buah = ['Jeruk', 'Nanas', 'Apel'];
```

```
buah.shift();  
console.log(buah)
```

```
buah.unshift('Lemon');  
console.log(buah)
```

delete & splice

```
let buah = ['Jeruk', 'Nanas', 'Apel'];
```

```
buah.splice(2, 0, 'Lemon', 'Kiwi');  
console.log(buah)
```

```
buah.splice(0, 1);  
console.log(buah)
```

```
delete buah[0];  
console.log(buah)
```


slice

```
let buah = ['Banana', 'Orange',  
            'Lemon', 'Apple', 'Mango'];
```

```
let buah2 = buah.slice(1);  
console.log(buah2)
```

```
let buah4 = buah.slice(1,4);  
console.log(buah4)
```

Adding Array Elements

```
let buah = ['Jeruk', 'Nanas', 'Apel'];
```

```
buah.push('Duku');  
console.log(buah)
```

```
buah[buah.length] = 'Pisang';  
console.log(buah)
```

```
buah[6] = 'Mangga';  
console.log(buah)
```

Merging (Concatenating) 2 Arrays

```
let nama1 = ['Andi', 'Budi'];  
let nama2 = ['Caca', 'Dede', 'Euis'];
```

```
let nama3 = nama1.concat(nama2);  
let nama4 = nama2.concat(nama1);
```

```
console.log(nama3)  
console.log(nama4)
```

Merging (Concatenating) 3 Arrays

```
let nama1 = ['Andi', 'Budi'];  
let nama2 = ['Caca', 'Dede', 'Euis'];  
let nama3 = ['Faza', 'Gilang'];
```

```
let x = nama1.concat(nama2, nama3);
```

```
console.log(nama1)  
console.log(nama2)  
console.log(nama3)  
console.log(x)
```

Numeric Sorting Ascending

Function comparison

```
let x = [40, 100, 1, 5, 25, 10];
```

```
console.log(x)
```

```
console.log(x.sort())
```

```
x.sort(function(a,b){  
    return a-b  
});
```

```
console.log(x)
```

Numeric Sorting Descending

Function comparison

```
let x = [40, 100, 1, 5, 25, 10];
```

```
console.log(x)
```

```
console.log(x.sort())
```

```
x.sort(function(a,b){  
    return b-a  
});
```

```
console.log(x)
```

Lowest & Highest Element #1

```
let x = [40, 100, 1, 5, 25, 10];
```

```
x.sort(function(a,b){  
    return a-b  
});
```

```
console.log(x[0])  
console.log(x[x.length-1])
```

Lowest & Highest Element #2

```
let x = [40, 100, 1, 5, 25, 10];
```

```
function nilaiMin(a) {  
    return Math.min.apply(null, a);  
}
```

```
function nilaiMax(a) {  
    return Math.max.apply(null, a);  
}
```

```
console.log(nilaiMin(x))  
console.log(nilaiMax(x))
```


Array of Arrays

```
var arrayKu = [  
  ['Andi', 24, 'PNS'],  
  ['Budi', 28, 'Pengacara'],  
  ['Caca', 21, 'Siswa'],  
]
```

```
console.log(arrayKu[0])  
console.log(arrayKu[0][0])  
console.log(arrayKu[1][1])  
console.log(arrayKu[2][2])
```

Sorting Array of Arrays

```
var arrayKu = [  
  ['Andi', 24, 'PNS'],  
  ['Budi', 28, 'Pengacara'],  
  ['Caca', 21, 'Siswa'],  
]
```

```
arrayKu[0].sort()  
arrayKu[1].reverse()
```

```
console.log(arrayKu[0])  
console.log(arrayKu[1])
```

Coding for Teens

JS

Exploring

#5 Function & Array