



RUMAH IT INDONESIA

By EL ghifari 

DASAR PEMROGRAMAN

# PROGRAMMER

Fundamental Programmer





# Apa itu bahasa pemrograman

## Bahasa Pemrograman adalah

Bahasa pemrograman adalah sekumpulan **instruksi** yang dapat **dimengerti** dan **dijalankan** oleh **komputer** untuk **melakukan tugas tertentu**. Bahasa ini memungkinkan programmer untuk menulis kode yang dapat memanipulasi data dan mengendalikan perilaku mesin atau perangkat lunak.

```
if ( $this->rule_exists( $resource_details['id'], $role_details['id'] ) ) {
    if ( $access == false ) {
        // Remove the rule as there is currently no need for it
        $details['access'] = !$access;
        $this->_sql->delete( 'acl_rules', $details );
    } else {
        // Update the rule with the new access value
        $this->_sql->update( 'acl_rules', array( 'access' => $access ) );
    }
}
foreach( $this->rules as $key=>$rule ) {
    if ( $details['role_id'] == $rule['role_id'] && $details['id'] != $rule['id'] ) {
```

# Fundamental & Elemen kunci

## Sintaksis

**Sintaksis** adalah *Aturan* dan *struktur* yang mengatur cara penulisan kode.

Sintaksis menentukan bagaimana pernyataan harus ditulis agar dapat dipahami oleh kompiler atau interpreter.

javascript

 Copy code

```
console.log("Hello, World!");
```

python

 Copy code

```
print("Hello, World!")
```

ruby

 Copy code

```
puts "Hello, World!"
```

kotlin

 Copy code

```
fun main() {  
    println("Hello, World!")  
}
```

php

 Copy code

```
<?php  
echo "Hello, World!";  
?>
```

Contoh :

1. Semua kode PHP harus berada di dalam tag PHP `<?php (semua kode disini) ?>`.
2. Komentar satu baris menggunakan `//` dan komentar multi baris menggunakan `/* ... */`.
3. Variabel di PHP diawali dengan tanda `$`.
4. Fungsi dideklarasikan menggunakan kata kunci `function`.
5. Penulisan sintaksis dalam deklarasi semantik di tutup dengan titik koma `;`

# Fundamental & Elemen kunci

## Semantik

**Semantik** adalah Makna dari struktur sintaksis yang ditulis.

Semantik menentukan apa yang akan dilakukan oleh kode saat dijalankan.

```
php Salin kode
<?php
$x = 5;
$y = 10;
$z = $x + $y;
echo $z;
?>
```

Sintaksis di atas memiliki semantik bahwa sintak variabel **\$z** akan berisi hasil penjumlahan dari **\$x** dan **\$y**, yang kemudian akan dicetak ke layar dengan sintak **echo**.

Jadi **Semantiknya adalah operasi penjumlahan dan pencetakan hasil.**



# Variabel dan type data

## Variabel dan Tipe Data adalah

**Variabel** digunakan untuk menyimpan data, dan **tipe data** menentukan jenis data yang dapat disimpan dalam variabel (misalnya, integer, string, float, dll). nanti kita akan pelajari

### JavaScript

1. **Number** - Bilangan (baik bilangan bulat maupun desimal)
2. **String** - String (teks)
3. **Boolean** - Boolean (true atau false)
4. **Object** - Objek
5. **Array** - Larik (daftar)
6. **Function** - Fungsi
7. **Undefined** - Tidak terdefinisi
8. **Null** - Null (kosong)
9. **Symbol** - Simbol (tipe data baru di ES6)
10. **BigInt** - Bilangan besar (di atas  $2^{53} - 1$ )

### PHP

1. **int** - Bilangan bulat
2. **float** - Bilangan desimal
3. **string** - String (teks)
4. **bool** - Boolean (true atau false)
5. **array** - Larik (daftar)
6. **object** - Objek
7. **resource** - Sumber daya eksternal (mis. koneksi database)
8. **NULL** - Null (kosong)

### Python

1. **int** - Bilangan bulat
2. **float** - Bilangan desimal
3. **str** - String (teks)
4. **bool** - Boolean (True atau False)
5. **list** - Daftar
6. **tuple** - Tupel (tidak bisa diubah)
7. **dict** - Kamus (key-value pairs)
8. **set** - Himpunan (unik, tidak berurutan)

Setiap bahasa pemrograman memiliki tipe data dasar dan kompleks yang mungkin berbeda dalam cara penggunaannya dan fitur-fiturnya. Tipe data di atas mencakup sebagian besar tipe data dasar yang umum digunakan dalam berbagai bahasa pemrograman.

# Variabel dan type data

```
python Copy code

# Integer
a = 10

# Float
b = 20.5

# String
c = "Hello"

# Boolean
d = True

# List
e = [1, 2, 3, 4, 5]

# Tuple
f = (1, 2, 3, 4, 5)

# Dictionary
g = {"key1": "value1", "key2": "value2"}

# Set
h = {1, 2, 3, 4, 5}
```

```
csharp Copy code

// Integer
int a = 10;

// Float
float b = 20.5f;

// Double
double c = 30.5;

// String
string d = "Hello";

// Boolean
bool e = true;

// Array
int[] f = {1, 2, 3, 4, 5};

// List
List<int> g = new List<int> {1, 2, 3, 4, 5};

// Dictionary
Dictionary<string, string> h = new Dictionary<string, string>
{
    {"key1", "value1"},
    {"key2", "value2"}
};

// HashSet
HashSet<int> i = new HashSet<int> {1, 2, 3, 4, 5};
```

```
php Copy code

// Integer
$a = 10;

// Float
$b = 20.5;

// String
$c = "Hello";

// Boolean
$d = true;

// Array
$e = [1, 2, 3, 4, 5];

// Associative Array (similar to dictionary/map)
$f = ["key1" => "value1", "key2" => "value2"];
```



# Kontrol Alur

**Kontrol alur** adalah

Mekanisme sintak berisikan semantik untuk mengontrol aliran eksekusi semantik program, seperti pernyataan kondisi (if-else), perulangan (for, while), dan switch-case.

```
<?php
$age = 20;

if ($age < 18) {
    echo "You are a minor.";
} elseif ($age < 65) {
    echo "You are an adult.";
} else {
    echo "You are a senior.";
}
```

python

```
x = 10
if x > 0:
    print("x is positive")
elif x == 0:
    print("x is zero")
else:
    print("x is negative")
```

javascript

```
let x = 10;
if (x > 0) {
    console.log("x is positive");
} else if (x === 0) {
    console.log("x is zero");
} else {
    console.log("x is negative");
}
```

# Objek & Kelas/ (OOP)

## Objek dan kelas adalah

Konsep dari **pemrograman berorientasi objek** (OOP) yang memungkinkan programmer untuk membuat objek yang merupakan instansiasi dari kelas. Kelas dapat memiliki atribut (data) dan metode (fungsi).

## Kelas (Class)

**Definisi:** Kelas adalah template atau cetak biru yang mendefinisikan atribut (data) dan metode (fungsi) yang dimiliki oleh objek.

**Tujuan:** Untuk mendefinisikan struktur dan perilaku yang akan dimiliki oleh objek yang dibuat dari kelas tersebut.

### Kegunaan:

- Abstraksi: Menyederhanakan kompleksitas dengan hanya menunjukkan antarmuka yang diperlukan.
- Enkapsulasi: Mengelompokkan data dan metode yang bekerja pada data tersebut dalam satu unit.
- Reusabilitas: Memungkinkan penggunaan kembali kode dengan mendefinisikan kelas sekali dan menggunakannya berulang kali.

## Objek (Object)

- **Definisi:** Objek adalah instance dari kelas. Ini adalah entitas yang dibuat berdasarkan definisi kelas.
- **Tujuan:** Untuk merepresentasikan entitas dunia nyata dengan atribut dan metode yang ditentukan oleh kelas.
- **Kegunaan:**
  - Keadaan: Menyimpan data spesifik dalam atributnya.
  - Interaksi: Berinteraksi dengan objek lain melalui metode.
  - Reusabilitas: Digunakan di berbagai konteks dalam program.

Kode mendefinisikan sebuah kelas Person dengan atribut **\$name** dan **\$age**, serta metode introduce. Objek **\$person1** adalah instansiasi dari kelas **Person**.

php

```
<?php
class Person {
    public $name;
    public $age;

    public function __construct($name, $age) {
        $this->name = $name;
        $this->age = $age;
    }

    public function introduce() {
        return "Hi, my name is $this->name and
    }
}

$person1 = new Person("Charlie", 25);
echo $person1->introduce();
?>
```