

LAPORAN
WORKSHOP PEMOGRAMAN WEB
(Semester 2)



Disusun Oleh:
FULAY FILLAH
E32231532
GOLONGAN C

PROGRAM STUDI TEKNIK KOMPUTER
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI JEMBER
TAHUN 2023/2024

ACARA 7

DASAR TEORI

Object Oriented Programming atau dalam bahasa indonesia diartikan Pemrograman Berbasis Objek, adalah salah satu cara membuat program (programming paradigm) BKPM Workshop Pemrograman Web 2 dengan memecah alur program menjadi modul-modul sederhana yang disebut dengan objek. Setiap objek akan memiliki fungsi dan tugas tersendiri.

OOP berbeda dengan prosedural programming yang memecah program menjadi fungsi-fungsi/prosedural. Pengertian OOP (Object Oriented Programming) adalah suatu metode pemrograman yang berorientasi kepada objek. Tujuan dari OOP diciptakan adalah untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari. Jadi setiap bagian dari suatu permasalahan adalah objek, nah objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil lagi. Saat ini, Object Oriented Programming (OOP) telah menjadi standar dalam dunia pemrograman, termasuk PHP. Walaupun kita bisa membuat program PHP tanpa menggunakan OOP sama sekali, namun untuk membuat aplikasi 'real world' yang fleksibel, programmer PHP akan beralih menggunakan OOP.

Jika anda telah menguasai pemrograman PHP dasar seperti tipe data, array, dan fungsi, mempelajari pemrograman objek PHP adalah langkah berikutnya. Fitur dan desain kode yang ditawarkan dengan membuat program menggunakan objek akan sangat memudahkan kita dalam merancang aplikasi website modern dan memiliki fleksibilitas yang tinggi. Terlebih jika anda memang 'serius' menguasai PHP, memahami pengertian dan cara penggunaan OOP dalam PHP sangat penting. Aplikasi framework PHP seperti Code Igniter, Yii Framework, Symfony dan Laravel, semuanya menggunakan OOP. Untuk mempermudah ilustrasi dalam belajar konsep OOP, saya akan ambil contoh Pesawat, Pesawat adalah sebuah objek. Pesawat itu sendiri terbentuk dari beberapa objek yang lebih kecil lagi seperti mesin, roda, baling-baling, kursi, dll. Pesawat sebagai objek yang terbentuk dari objek-objek yang lebih kecil saling berhubungan, berinteraksi, berkomunikasi dan saling mengirim pesan kepada objek-objek yang lainnya. Begitu juga dengan program, sebuah objek yang besar dibentuk dari beberapa objek yang lebih kecil, objek-objek itu saling berkomunikasi, dan saling berkirim pesan kepada objek yang lain. Class adalah cetak biru atau blueprint dari object. Class digunakan hanya untuk membuat kerangka dasar. Yang akan kita pakai nantinya adalah hasil cetakan dari class, yakni

object. Class adalah sebuah konsep OOP yang digunakan untuk mengencapsulasi /membungkus data dan abstraksi prosedural yang diperlukan dalam menggambarkan isi dan tingkah laku berbagai entitas. Sebagai analogi, class bisa diibaratkan dengan laptop atau notebook. Kita tahu bahwa laptop memiliki ciri-ciri seperti merk, memiliki keyboard, memiliki processor, dan beberapa ciri khas lain yang menyatakan sebuah benda tersebut adalah laptop. Selain memiliki ciriciri, sebuah laptop juga bisa dikenakan tindakan, seperti: menghidupkan laptop atau mematikan laptop. Class dalam analogi ini adalah gambaran umum tentang sebuah benda. BKPM Workshop Pemrograman Web 3 Di dalam pemrograman nantinya, contoh class seperti: koneksi_database dan profile_user. Di dalam PHP, penulisan class diawali dengan keyword class, kemudian diikuti dengan nama dari class. Aturan penulisan nama class sama seperti aturan penulisan variabel dalam PHP, yakni diawali dengan huruf atau underscore untuk karakter pertama, kemudian boleh diikuti dengan huruf, underscore atau angka untuk karakter kedua dan selanjutnya. Isi dari class berada dalam tanda kurung kurawal. Kelas juga merupakan deskripsi tergeneralisir (misal templet, pola, cetak biru) yang menggambarkan kumpulan objek yang sama, untuk membuat class dalam pemograman PHP bisa digunakan dengan menambahkan keyword class lalu di ikuti oleh nama class.

Property (atau disebut juga dengan atribut) adalah data yang terdapat dalam sebuah class. Melanjutkan analogi tentang laptop, property dari laptop bisa berupa merk, warna, jenis processor, ukuran layar, dan lain-lain. Property dapat disebut juga variable dari Class sedangkan Method biasa dalam bentuk fungsi. Jika sudah terbiasa dengan program PHP, property ini sebenarnya hanyalah variabel yang terletak di dalam class. Seluruh aturan dan tipe data yang biasa diinput ke dalam variabel, bisa juga diinput kedalam property. Aturan tata cara penamaan property sama dengan aturan penamaan variabel. Pada Class person kita akan menambahkan sebuah properti baru dengan nama name, pada konsep OOP dalam bahasa pemograman PHP untuk menyatakan sebuah properti bisa dengan menggunakan keyword var lalu di ikuti dengan nama propertinya.

Method adalah tindakan yang bisa dilakukan di dalam class. Jika menggunakan analogi class laptop kita, maka contoh method adalah: menghidupkan laptop, mematikan laptop, mengganti cover laptop, dan berbagai tindakan lain. Method adalah suatu operasi berupa fungsi-fungsi yang dapat dikerjakan oleh suatu object. Method didefinisikan pada class akan tetapi dipanggil melalui object. Untuk membuat method bisa dengan dimulai dengan keyword function lalu di ikuti oleh nama functionnya. Method pada dasarnya adalah function yang berada di dalam class. Seluruh fungsi dan sifat function bisa

diterapkan ke dalam method, seperti argumen/parameter, mengembalikan nilai (dengan keyword return), dan lain-lain. Berikut ini adalah contoh method Setter dan Getter untuk memberikan value kepada properti dan mengakses.

Object atau Objek adalah hasil cetak dari class, atau hasil 'konkrit' dari class. Jika menggunakan analogi class laptop, maka objek dari class laptop bisa berupa: laptop_andi, laptop_anto, laptop_duniaikom, dan lain-lain. Objek dari class laptop akan memiliki seluruh ciri-ciri laptop, yaitu property dan method-nya.

Constructor adalah fungsi khusus / fungsi unik yang akan di panggil secara otomatis pada saat object dibuat. Secara umum, pada saat pembuatan object, class harus menginisialisasikan variabelnya untuk menghindari error yang tidak diinginkan sedangkan adalah method khusus yang dijalankan secara otomatis pada saat sebuah objek dihapus. Di dalam PHP, seluruh objek secara otomatis dihapus ketika halaman PHP dimana objek itu berada selesai diproses.

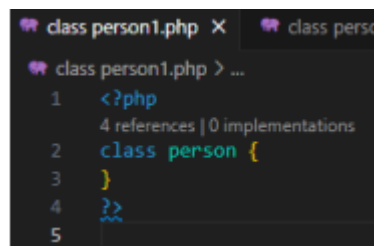
(tugas 7)

SOAL:

1. Cobalah seluruh potongan source-code di bagian dasar teori!

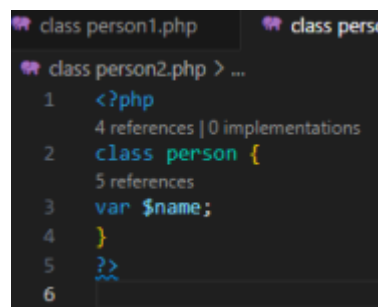
JAWAB:

1. Code pertama



```
class person1.php X class perso
class person1.php > ...
1 <?php
  4 references | 0 implementations
2 class person {
3 }
4 ?>
5
```

2. Code kedua



```
class person1.php class perso
class person2.php > ...
1 <?php
  4 references | 0 implementations
2 class person {
  5 references
3 var $name;
4 }
5 ?>
6
```

3. Ss an ke 3

```
class person1.php  class person2.php
class person3.php > ...
1  <?php
   4 references | 0 implementations
2  class person {
   5 references
3  var $name;
   4 references | 0 overrides
4  function set_name($new_name) {
5  $this->name = $new_name;
6  }
   2 references | 0 overrides
7  function get_name() {
8  return $this->name;
9  }
10 }
11 ?>
12
```

4. Ss an ke 4

```
class person1.php  class person2.php  class person3.php
class person4.php
1  <?php include("class person3.php"); ?>
2  <?php
3  $stefan = new person();
4  $jimmy = new person;
5  $stefan->set_name("Stefan Mischhook");
6  $jimmy->set_name("Nick Waddles");
7  echo "Stefan's full name: " . $stefan->get_name();
8  echo "Nick's full name: " . $jimmy->get_name();
9  ?>
```

5. Ss ke 5

```
class person1.php  class person2.php  class p
class person5.php
1  <?php include("class person3.php"); ?>
2  <?php
3  $stefan = new person();
4  $jimmy = new person;
5  $stefan->set_name("Stefan Mischhook");
6  $jimmy->set_name("Nick Waddles");
7  // directly accessing properties in a clas
8  echo "Stefan's full name: ".$stefan->name;
9  ?>
```

6. Screenshot 6

```
class person1.php class person2.php class person6.php > ...
1  <?php
   4 references | 0 implementations
2  class person {
   5 references
3  var $name;
   4 references | 0 overrides
4  function __construct($persons_name) {
5  echo "<p>initialize class</p>";
6  }
   4 references | 0 overrides
7  function set_name($new_name) {
8  $this->name = $new_name;
9  }
   2 references | 0 overrides
10 function get_name() {
11 return $this->name;
12 }
   0 references | 0 overrides
13 function __destructor(){
14 echo "<p>end class</p>";
15 }
16 }
17 ?>
18
```

7. Screenshot 7 hasil

Stefan's full name: Stefan MischookNick's full name: Nick Waddles

Stefan's full name: Stefan Mischook

ACARA 8

(tugas 8)

SOAL

1. Buatlah aplikasi perhitungan yang mengimplementasi OOP!

JAWAB

1. Ini adalah contoh codingan dari kalkulator sederhananya

```

1  <?php
2
3  class Kalkulator {
4      public function tambah($a, $b) {
5          return $a + $b;
6      }
7
8      public function kurang($a, $b) {
9          return $a - $b;
10     }
11
12     public function kali($a, $b) {
13         return $a * $b;
14     }
15
16     public function bagi($a, $b) {
17         if ($b != 0) {
18             return $a / $b;
19         } else {
20             return "Tidak dapat dibagi oleh nol";
21         }
22     }
23 }
24
25 // Contoh penggunaan:
26 $kalkulator_sederhana = new Kalkulator();
27
28 // Penjumlahan
29 $hasil_tambah = $kalkulator_sederhana->tambah(5, 3);
30 echo "Hasil Penjumlahan: " . $hasil_tambah . "\n";
31
32 // Pengurangan
33 $hasil_kurang = $kalkulator_sederhana->kurang(8, 2);
34 echo "Hasil Pengurangan: " . $hasil_kurang . "\n";
35
36 // Perkalian
37 $hasil_kali = $kalkulator_sederhana->kali(4, 6);
38 echo "Hasil Perkalian: " . $hasil_kali . "\n";
39
40 // Pembagian
41 $hasil_bagi = $kalkulator_sederhana->bagi(9, 3);
42 echo "Hasil Pembagian: " . $hasil_bagi . "\n";
43
44 ?>

```

PENJELASAN:

```

1  <?php
2
3  class Kalkulator {
4      public function tambah($a, $b) {
5          return $a + $b;
6      }
7
8      public function kurang($a, $b) {
9          return $a - $b;
10     }
11

```


1. Dalam contoh codingan di atas yang di maksud dengan class kalkulator adalah sebuah definisi kelas dalam bahasa pemrograman PHP. Sebuah kelas adalah suatu entitas yang membungkus properti (variabel) dan metode (fungsi) yang berhubungan satu sama lain. Dalam hal ini, kita memiliki kelas `Kalkulator` yang digunakan untuk menyusun operasi-operasi kalkulasi.

```
25 // Contoh penggunaan:
26 $kalkulator_sederhana = new Kalkulator();
27
```

2. Kelas `kalkulator` ini dapat dianggap sebagai suatu entitas atau "blueprint" untuk menciptakan objek-objek kalkulator. Saat kita membuat objek dari kelas ini, kita dapat menggunakan metode-metodenya untuk melakukan operasi kalkulasi. Misalnya, pada bagian penggunaan contoh di akhir kode, kita membuat objek `$kalkulator_sederhana` dan menggunakan metode-metodenya untuk melakukan perhitungan.

```
4     public function tambah($a, $b) {
5         return $a + $b;
6     }
7
8     public function kurang($a, $b) {
9         return $a - $b;
10    }
11
12    public function kali($a, $b) {
13        return $a * $b;
14    }
15
16    public function bagi($a, $b) {
17        if ($b != 0) {
18            return $a / $b;
19        } else {
20            return "Tidak dapat dibagi oleh nol";
21        }
22    }
23 }
```

3. Untuk membuat kalkulator kita pastinya membutuhkan pembuatan fungsi tambah, kurang, kali, bagi.
4. Dalam contoh codingan di atas fungsi dari return pada kalkulator sederhana tersebut yaitu digunakan untuk mengembalikan nilai dari suatu metode (fungsi) di dalam kelas `Kalkulator`.
5. Terakhir mari kita coba jalankan kalkulator tersebut.



Hasilnya benar dan tidak ada yang salah.

Link github: <https://github.com/fulayyy/projectfulay>