

Nama : Ryan Andrianto

NIM : G.211.21.0105

MATKUL : Pemrograman Framework

LINK :

1) Latihan 1: class dan object

Code:

```
1 <!DOCTYPE html>
2 <html Lang="en">
3
4 <head>
5     <meta charset="UTF-8">
6     <meta http-equiv="X-UA-Compatible" content="IE=edge">
7     <meta name="viewport" content="width=device-width, initial-scale=1.0">
8     <title>Document</title>
9     <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css">
10    <script defer src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"></script>
11 </head>
12
13 <body>
14     <?php
15         //class mobil
16         class Mobil
17         {
18             public $nama;
19             public $merk;
20             function getInfo($a)
21             {
22                 echo "Nama mobil: " . $a->nama . "<br/>";
23                 echo "Merk: " . $a->merk . "<br/>";
24             }
25         }
26
27     ?>
28     <div class="container mt-5 bg-gray-300">
29         <div class="card pt-3 px-3 shadow mx-auto" style="width: 400px">
30             <div class="card-header text-center text-bg-danger border-0 rounded">
31                 <h4>Tugas 2 || Lat4_1.php </h4>
32             </div>
33             <div class="card-body">
34                 <p>
35                     <?php
36                         $a = new Mobil();
37                         $a->nama = "Ferrari F8 Tributo";
38                         $a->merk = "Ferrari";
39                         $a->getInfo($a);
40                     ?>
41                 </p>
42             </div>
43             <div class="btn-group btn-group-sm" style="width: 40px;">
44                 <a href="Lat4_2a.php" class="btn btn-danger">
45                     < </a>
46             </div>
47             <p style="font-size: 15px;" class="text-center mt-2">
48                 Alden Christian T.W || G.211.21.0087
49             </p>
50         </div>
51     </div>
52 </body>
53
54 </html>
```

Hasil

Tugas 2 || Lat4_1.php

Nama mobil: Suzuki carry

Merk: Suzuki



Ryan Andrianto || G.211.21.0105

Kesimpulan:

1. Cara membuat class pada PHP

```
<?php
```

```
Class nama_class {
```

```
}
```

```
?>
```

2. Cara penulisan property

```
Modifier $nama_properti;
```

3. Penulisan method

```
Modifier function nama_method(){
```

```
Isi_method;}
```

4. Cara inisiasi object

```
$nama_object = new nama_class();
```

5. Cara mengisi property atau mendefinisikan property

```
$nama_object->properties="aaa";
```

6. Cara memanggil/menjalankan method pada suatu Class

```
$nama_object->nama_methode();
```

2) Latihan 2

Code :

```
1  <?php
2  class mahasiswa
3  {
4      public $nama;
5      public $nim;
6      public $prodi;
7      public $semester;
8      function __construct($a, $b, $c, $d)
9      {
10         $this->nama = $a;
11         $this->nim = $b;
12         $this->prodi = $c;
13         $this->semester = $d;
14
15         echo "Kelas telah dibuat<br/><br/>";
16     }
17     function cetak()
18     {
19         echo $this->nama . "<br/>" . $this->nim . "<br/>";
20         echo $this->prodi . "<br/>" . $this->semester . "<br/></br>";
21     }
22     function __destruct()
23     {
24         echo "Kelas telah dihancurkan";
25     }
26 }
27 class mahasiswi
28 {
29     public $nama;
30     public $nim;
31     public $prodi;
32     public $semester;
33     function __construct($a, $b, $c, $d)
34     {
35         $this->nama = $a;
36         $this->nim = $b;
37         $this->prodi = $c;
38         $this->semester = $d;
39
40         echo "Kelas telah dibuat<br/><br/>";
41     }
42     function cetak()
43     {
44         echo $this->nama . "<br/>" . $this->nim . "<br/>";
45         echo $this->prodi . "<br/>" . $this->semester . "<br/></br>";
46     }
47     function __destruct()
48     {
49         echo "Kelas telah dihancurkan";
50     }
51 }
52
```

```
1 <?php
2 require_once("lat4_2a.php");
3 $mhs2 = new mahasiswa("Ryan Andrianto", "G.211.21.0105", "FTIK", "Semester 3");
4 $mhs2->cetak();
5
```

Hasil :

← → ↻ ⓘ http://localhost/tut/Lat4_2b.php

Kelas telah dibuat

Ryan Andrianto
G.211.21.0105
FTIK
Semester 3

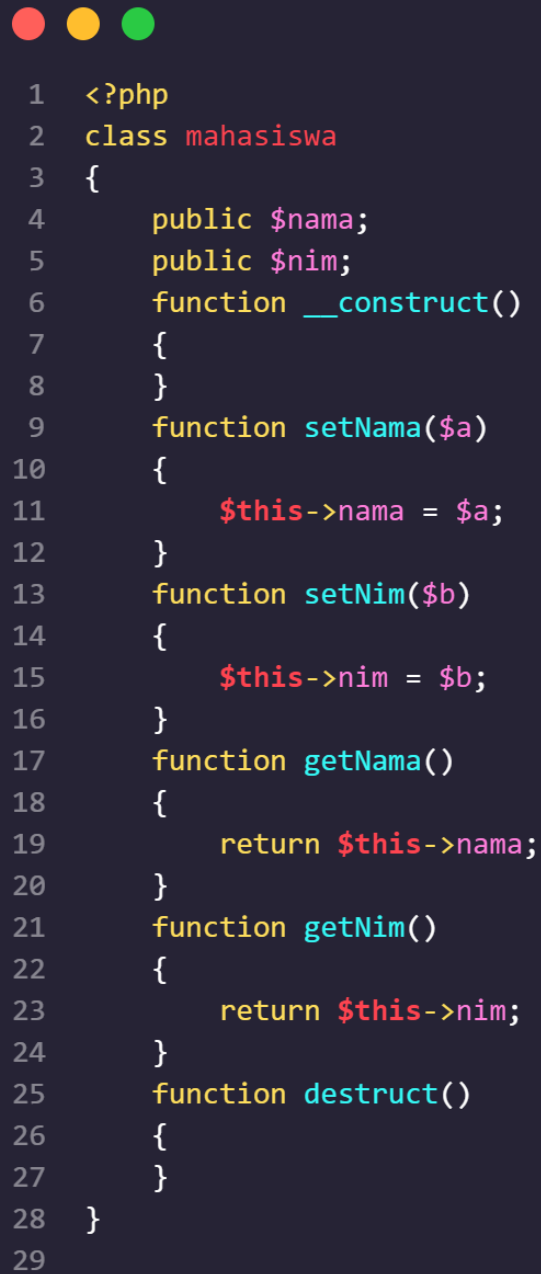
Kelas telah dihancurkan

Kesimpulan:


Constructor dalam OOP PHP tidak dapat di Override

3) Latihan 3

Code :



```
1  <?php
2  class mahasiswa
3  {
4      public $nama;
5      public $nim;
6      function __construct()
7      {
8      }
9      function setNama($a)
10     {
11         $this->nama = $a;
12     }
13     function setNim($b)
14     {
15         $this->nim = $b;
16     }
17     function getNama()
18     {
19         return $this->nama;
20     }
21     function getNim()
22     {
23         return $this->nim;
24     }
25     function destruct()
26     {
27     }
28 }
29
```



```
1  <?php
2  require_once("lat4_3a.php");
3  $mhs1 = new mahasiswa();
4  $mhs1->nama = "Ryan Andrianto </br></br>";
5  $mhs1->nim = "G.211.21.0105";
6  echo $mhs1->nama;
7  echo $mhs1->nim;
8
```

Hasil :

Ryan Andrianto

G.211.21.0105

Kesimpulan:

- A. Program tersebut mengalami beberapa error, Hal itu terjadi dikarenakan Property yang bermodifikasi private hanya bisa digunakan pada class mahasiswa sendiri.
- B. Modifier protected dan Public
 - b.1 Protected
 - b.2 Public

4) Latihan 4

Code :

```
1  <?php
2  require_once("Lat4_3a.php");
3  class asisten extends mahasiswa
4  {
5      function_construct (){}
6  }
```

```
1  <?php
2  require_once("lat4_4a.php");
3  $as = new asisten();
4  $as->setNama("tes");
5  echo $as->getNama();
6
```

Hasil :

Parse error: syntax error, unexpected 'function_construct' (T_STRING), expecting function (T_FUNCTION) or const (T_CONST) in C:\xampp\htdocs\tut\Lat4_4a.php on line 5

- A. **Kesimpulan:** : Pada PHP OOP, Class asisten (child) bisa memanggil metode dari mahasiswa (Parent).

5) Latihan 5

Code :

```
1  <?php
2  abstract class mahasiswa
3  {
4      abstract protected function getTugasAkhir();
5      abstract protected function getProgram($postfix);
6      public function tugasAkhir()
7      {
8          print $this->getTugasAkhir() . "<br>";
9      }
10 }
11 class sarjana extends mahasiswa
12 {
13     protected function getTugasAkhir()
14     {
15         return "Skripsi";
16     }
17     public function getProgram($postfix)
18     {
19         print "{$postfix} S1";
20     }
21 }
22 class magister extends mahasiswa
23 {
24     public function getTugasAkhir()
25     {
26         return "Tesis";
27     }
28     public function getProgram($postfix)
29     {
30         print "{$postfix} S2";
31     }
32 }
33
```

```
1  <?php
2  require_once("lat4_5a.php");
3  $s = new sarjana;
4  $s->getProgram('Mahasiswa') . "<br>";
5  $s->tugasAkhir();
6  $m = new magister;
7  $m->getProgram('Mahasiswa') . "<br>";
8  $m->tugasAkhir();
9
```


Hasil :

Mahasiswa S1Skripsi
Mahasiswa S2Tesis

- A. **Kesimpulan:** Kesimpulan class anak yang mewarisi super class harus menuliskan semua method abstrak dari super classnya.

6) Latihan 6

Code :

```
1  <?php
2  interface a
3  {
4      public function foo();
5  }
6  interface b
7  {
8      public function bar();
9  }
10 interface c extends a, b
11 {
12     public function baz();
13 }
14 class d implements c
15 {
16     public function foo()
17     {
18     }
19     public function bar()
20     {
21     }
22     public function baz()
23     {
24     }
25 }
26 class e
27 {
28     public function foo()
29     {
30     }
31     public function bar()
32     {
33     }
34 }
35
```

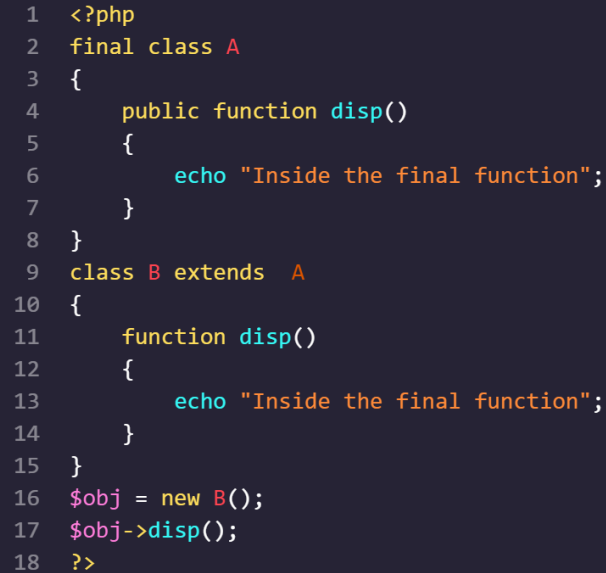
Kesimpulan:

- 1) Interface didefinisikan dengan “Interface” keyword, mirip dengan deklarasi class biasa, hanya saja definisi atau detail method tidak dituliskan.
- 2) Seluruh method yang dideklarasikan pada interface harus memiliki modifier “ public ”.
- 3) Untuk mengimplementasikan sebuah interface, kita dapat menggunakan “implement” keyword.
- 4) Seluruh method yang ada pada interface harus diimplementasikan seluruhnya. Sebuah class bisa mengimplementasikan lebih dari satu interface.

- 5) Class tidak bisa mengimplementasikan dua interface yang mempunyai nama method yang sama.
- 6) Interface bisa diwariskan seperti class menggunakan “extends”.
- 7) Class yang mengimplementasikan interface harus menggunakan method-method yang ada pada interface tersebut dengan nama dan spesifikasi yang sama persis.

7) Latihan 7

Code :



```
1  <?php
2  final class A
3  {
4      public function disp()
5      {
6          echo "Inside the final function";
7      }
8  }
9  class B extends A
10 {
11     function disp()
12     {
13         echo "Inside the final function";
14     }
15 }
16 $obj = new B();
17 $obj->disp();
18 ?>
```

Hasil :

Fatal error: Class B may not inherit from final class (A) in C:\xampp\htdocs\tut\Lat4_7.php on line 15

Kesimpulan:

1. final, akan mencegah proses overriding method pada class anak (sub-class)
2. Apabila metode kita berikan status final, maka metode tersebut tidak akan bisa dioverride, begitu juga pada class, apabila kita berikan status “final” pada deklarasi class maka class tersebut tidak bisa diperpanjang (diwariskan).

8) Latihan 8

Code :

```
1  <?php
2  class One
3  {
4      private static $var = 10;
5      function __construct()
6      {
7      }
8      static function disp()
9      {
10         print self::$var;
11     }
12     function __destruct()
13     {
14     }
15 }
16 One::disp();
17
```

Hasil :

10

Kesimpulan:

mengimplementasikan penggunaan property static dengan modifier private. Properti statis dideklarasikan dengan menggunakan kata kunci statis sebelum modifier- Sintaks:

modifier static \$nama_property = nilai;

2. Sifat statis dapat diakses tanpa perlu sebuah contoh objek dari kelas, menggunakan nama kelas bersama dengan ::- Sintaks:

ClassName :: \$nama_property method_name();

Perhatikan bahwa properti statis menggunakan tanda dollar (\$). Properti statis tidak dapat diakses melalui obyek menggunakan operator panah "->"

9) Latihan 9

Dari contoh kode MVC pada penjelasan diatas buatlah :

1. Model berhubungan langsung dengan database mysql dengan spesifikasi sebagai berikut :

- a. Database : Library
- b. Table : book

Username dan password menyesuaikan.

2. Buatlah view menjadi sebuah object, susun class view dimana fungsinya adalah mengambil file-file template yang sudah disediakan!

3. Tuliskan masing-masing kodenya, buatlah simulasinya!

Code:

a. Database dan Table

Menampilkan bars 0 - 4 (total 5, Pencarian dilakukan dalam 0,0003 detik.)

SELECT * FROM `buku`

Profil [Edit kotak] [Ubah] [Jelaskan SQL] [Buat kode PHP] [Segarkan]

☐ Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini | Sort by key: Tidak ada

<input type="checkbox"/>	Ubah	Salin	Hapus	13	One Piece	Eiichiro Oda	Shueisha	1997	8997399100017 30 2010-03-10 03102022193043
<input type="checkbox"/>	Ubah	Salin	Hapus	14	Naruto	Masashi Kishimoto	Shueisha	1997	8997399100017 30 2003-02-10 03102022193039
<input type="checkbox"/>	Ubah	Salin	Hapus	15	Full Metal Alchemist	Hiromu Arakawa	Elex media komputindo	1997	8997399100017 30 2007-04-10 03102022193032
<input type="checkbox"/>	Ubah	Salin	Hapus	16	Dragon Ball	Akira Toriyama	Elex media komputindo	1997	8997399100017 30 2004-02-10 03102022193026
<input type="checkbox"/>	Ubah	Salin	Hapus	17	Sword Art Online	Kawahara Reki	A-1 Pictures	1997	8997399100017 30 2019-12-01 03102022193020

☐ Pilih Semua | Dengan pilihan: ☐ Ubah ☐ Salin ☐ Hapus ☐ Ekspor

☐ Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini | Sort by key: Tidak ada

b. Model

```
1 <?php
2
3 class Koneksi {
4     var $host = 'localhost';
5     var $user = 'root';
6     var $pass = '';
7     var $db_name = 'perpustakaan';
8
9     public function __construct() {
10         $this->konek = mysqli_connect($this->host, $this->user, $this->pass, $this->db_name);
11     }
12 }
13
14 ?>
```

c. Controller




```
1 <?php
2
3 require_once '../..../config/perpustakaan';
4
5 class buku extends perpustakaan {
6     public function getDataPetugas($id) {
7         $stmt = mysqli_query($this->konek, "SELECT * FROM tb_petugas WHERE id_petugas = '" . $id . "'");
8
9         return $stmt;
10    }
11 }
```


d. View & admin


[illegible]


e. Hasil

Toko Buku Oase



 Data Buku

 Data Peminjam

 Transaksi Peminjaman

Logout

Tambah Buku

Table Buku

10 records per page

Search:

No	Judul	Pengarang	Penerbit	ISBN	Jumlah Buku	Aksi
1	One Piece	Eiichiro Oda	Shueisha	8997399100017	30	Ubah Hapus
2	Naruto	Masashi Kishimoto	Shueisha	8997399100017	30	Ubah Hapus
3	Full Metal Alchemist	Hiromu Arakawa	Elex media komputindo	8997399100017	30	Ubah Hapus
4	Dragon Ball	Akira Toriyama	Elex media komputindo	8997399100017	30	Ubah Hapus
5	Sword Art Online	Kawahara Reki	A-1 Pictures	8997399100017	30	Ubah Hapus

Showing 1 to 5 of 5 entries

[Previous](#) [1](#) [Next](#)