

## **JOB SHEET 14 PEMROGRAMAN WEB DINAMIS**

### **CLASS DAN OBJEK PADA PHP (LANJUTAN)**

#### **I. Tujuan Instruksional Khusus**

1. Mahasiswa dapat memahami konsep pewarisan pada object oriented programming
2. Mahasiswa dapat memahami konsep enkapsulasi dalam pewarisan
3. Mahasiswa dapat memahami polimorfisme dalam objek oriented programming

#### **II. Landasan Teori**

##### **2. 1. Pewarisan (Inheritance)**

Inheritance atau pewarisan dalam OOP dalam PHP adalah mewariskan property dan method baik public maupun protected kepada class lain. Konsep ini sangat berguna jika kita ingin membuat class lagi yang memiliki fungsi mirip dengan class yang sudah ada, sehingga tidak usah membuat lagi method atau property yang memiliki fungsi mirip atau sama. Dengan inheritance kita bisa menghindari duplikasi kode program, atau disebut juga code reuse.

Konsep inheritance membuat sebuah struktur atau hierarki class dalam code program. Class yang akan ‘diturunkan’ bisa disebut sebagai class induk (parent class), super class atau base class. Sedangkan class yang ‘menerima penurunan’ bisa disebut sebagai class anak (child class), sub class, derived class atau heir class.

Tidak semua property dan method dari class induk akan diturunkan. Property dan method dengan hak akses private, tidak akan diturunkan kepada class anak. Hanya property dan method dengan hak akses protected dan public saja yang bisa diakses dari class anak.

Di dalam PHP, inheritance / penurunan dari sebuah class kepada class lain menggunakan kata kunci : “extends”.

```

C: > xampp > htdocs > jobsheet1 > OOP > extends.php
1  <?php
2      class induk
3      {
4          // ... class induk
5      }
6      class anak extends induk
7      {
8          // ..class anak bisa mengakses
9          // ...property dan method class induk
10     }
11  ?>

```

## 2. 2. Polimorfisme

Dari segi bahasa, polimorfisme berasal dari dua kata bahasa latin yakni poly dan morph. Poly berarti banyak, dan morph berarti bentuk. Polimorfisme berarti banyak bentuk. Di dalam pemrograman berbasis objek, polimorfisme adalah konsep dimana terdapat banyak class yang memiliki signature method yang sama. Implementasi dari method-method tersebut diserahkan kepada tiap class, akan tetapi cara pemanggilan method harus sama. Agar kita dapat “memaksakan” signature method yang sama pada banyak class, class tersebut harus diturunkan dari sebuah abstract class atau object interface.

### III. Peralatan yang Digunakan

1. PC
2. Peramban web : Internet Explorer, Firefox, Google Chrome, Opera atau Safari
3. Aplikasi server local : XAMPP
4. Aplikasi text editor : Visual Studio Code

### IV. Langkah Kerja

#### 4. 1. Latihan Inheritance 1

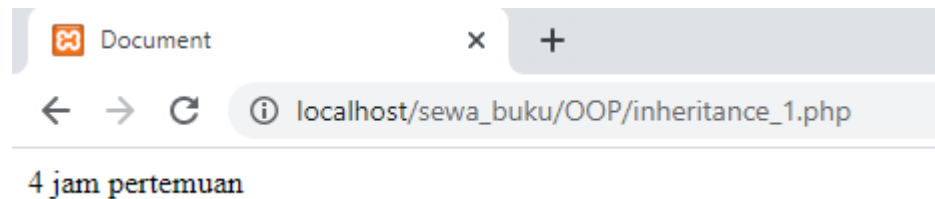
1. Buatlah file bernama inheritance\_1.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```

C: > xampp > htdocs > sewa_buku > OOP > inheritance_1.php
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <meta http-equiv="X-UA-Compatible" content="ie=edge">
7 <title>Document</title>
8 </head>
9 <body>
10 <?php
11 // buat class kuliah
12 class kuliah{
13     protected $jam_perkuliah = "4 jam pertemuan"; // property dengan hak akses protected
14 }
15 // buat class praktikum
16 class praktikum extends kuliah{
17     public function tampilkan_pertemuan() {
18         return $this->jam_perkuliah;
19     }
20 }
21 // buat objek dari class praktikum (instansiasi)
22 $kuliah_baru = new praktikum();
23 // jalankan method
24 echo $kuliah_baru->tampilkan_pertemuan(); // "Intel Core i7-4790 3.6Ghz"
25 ?>
26 </body>
27 </html>
28

```

2. Buka browser, kemudian ketikkan alamat :  
[http://localhost/sewa\\_buku/OOP/inheritance\\_1.php](http://localhost/sewa_buku/OOP/inheritance_1.php)



#### 4. 2. Latihan Inheritance 2

1. Buatlah file bernama file bernama inheritance\_2.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```

C: > xampp > htdocs > sewa_buku > OOP > inheritance_2.php
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Document</title>
7  </head>
8  <body>
9      <?php
10         // buat class induk: mahasiswa
11         class mahasiswa {
12             public $nama;
13             public $nim;
14             public $alamat;
15             public function kuliah() {
16                 return "Mengikuti perkuliahan";
17             }
18         }
19         // turunkan class mahasiswa ke mhs_informatika
20         class mhs_informatika extends mahasiswa{
21             public function belajar_pemrograman() {
22                 return "nama: $this->nama, nim: $this->nim,
23                     alamat: $this->alamat";
24             }
25         }
26         // buat objek dari class mhs_informatika (instansiasi)
27         $mhs_informatika_baru = new mhs_informatika();
28         // isi property objek
29         $mhs_informatika_baru->nama = "Sofia";
30         $mhs_informatika_baru->nim = "12345678";
31         $mhs_informatika_baru->alamat = "Semarang";
32         //panggil method objek
33         //panggil method objek
34         echo $mhs_informatika_baru->kuliah();echo "<br />";
35         echo $mhs_informatika_baru->belajar_pemrograman();
36     ?>
37 </body>
38 </html>

```

2. Buka browser, kemudian ketikkan alamat :
- [http://localhost/sewa\\_buku/OOP/inheritance\\_2.php](http://localhost/sewa_buku/OOP/inheritance_2.php)

← → ↻ ⓘ localhost/sewa\_buku/OOP/inheritance\_2.php

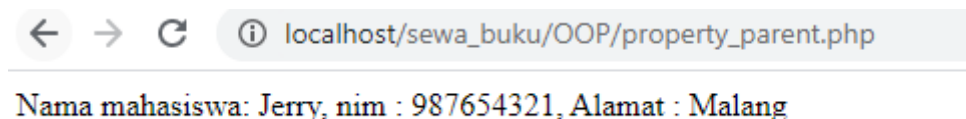
Mengikuti perkuliahan  
 nama: Sofia, nim: 12345678, alamat: Semarang

#### 4. 3. Latihan Cara Mengakses Property Parent Class

1. Buatlah file bernama file bernama property\_parent.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```
C: > xampp > htdocs > sewa_buku > OOP > property_parent.php
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Document</title>
7  </head>
8  <body>
9      <?php
10         // buat class mahasiswa
11         class mahasiswa {
12             public function data_mahasiswa() {
13                 return "Nama mahasiswa: Mariska,
14                     nim : 12345678, Alamat : Yogyakarta";
15             }
16         }
17         // turunkan class mahasiswa ke mhs_informatika
18         class mhs_informatika extends mahasiswa {
19             public function data_mahasiswa() {
20                 return "Nama mahasiswa: Jerry,
21                     nim : 987654321, Alamat : Malang";
22             }
23         }
24         // buat objek dari class laptop (instansiasi)
25         $mahasiswa_baru = new mhs_informatika();
26         //panggil method lihat_spec()
27         echo $mahasiswa_baru->data_mahasiswa();
28     ?>
29 </body>
30 </html>
31
```

2. Buka browser, kemudian ketikkan alamat :  
[http://localhost/sewa\\_buku/OOP/property\\_parent.php](http://localhost/sewa_buku/OOP/property_parent.php)



Nama mahasiswa: Jerry, nim : 987654321, Alamat : Malang

#### 4. 4. Latihan Cara Memanggil Method Parent Class dari Child Class

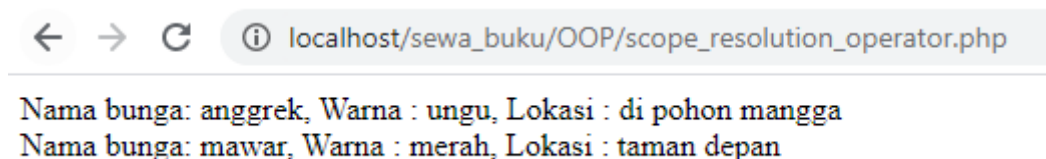
1. Buatlah file bernama file bernama scope\_resolution\_operator.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```

C: > xampp > htdocs > sewa_buku > OOP > scope_resolution_operator.php
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Document</title>
7  </head>
8  <body>
9      <?php
10         // buat class bunga
11         class bunga {
12             public function lihat_bunga() {
13                 return "Nama bunga: mawar,
14                     Warna : merah, Lokasi : taman depan";
15             }
16         }
17         // turunkan class bunga ke anggrek
18         class anggrek extends bunga {
19             public function lihat_bunga() {
20                 return "Nama bunga: anggrek,
21                     Warna : ungu, Lokasi : di pohon mangga";
22             }
23             public function lihat_jenis_bunga() {
24                 return parent::lihat_bunga();
25             }
26         }
27         $anggrek_baru = new anggrek(); // buat objek dari class anggrek (instansiasi)
28         echo $anggrek_baru->lihat_bunga();echo "<br />"; //panggil method lihat_bunga()
29         echo $anggrek_baru->lihat_jenis_bunga(); //panggil method lihat_jenis_bunga()
30     ?>
31 </body>
32 </html>

```

2. Buka browser, kemudian ketikkan alamat :
- [http://localhost/sewa\\_buku/OOP/scope\\_resolution\\_operator.php](http://localhost/sewa_buku/OOP/scope_resolution_operator.php)



Nama bunga: anggrek, Warna : ungu, Lokasi : di pohon mangga  
 Nama bunga: mawar, Warna : merah, Lokasi : taman depan

#### 4. 5. Latihan Cara Memanggil Constructor dan Destructor Parent

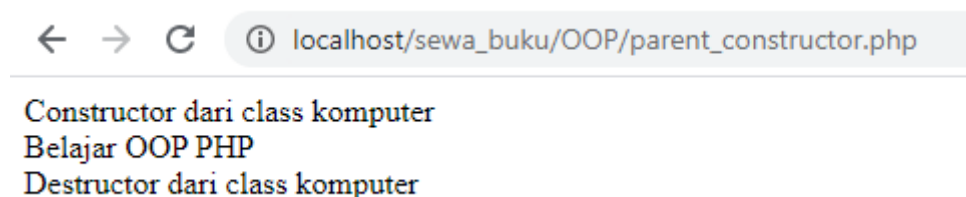
1. Buatlah file bernama file bernama parent\_constructor.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```

C: > xampp > htdocs > sewa_buku > OOP > parent_constructor.php
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Document</title>
7  </head>
8  <body>
9      <?php
10         // buat class komputer
11         class komputer {
12             // buat constructor class komputer
13             public function __construct() {
14                 echo "Constructor dari class komputer <br />";
15             }
16             // buat destructor class komputer
17             public function __destruct() {
18                 echo "Destructor dari class komputer <br />";
19             }
20         }
21         // turunkan class komputer ke laptop
22         class laptop extends komputer {
23         }
24         // turunkan class laptop ke chromebook
25         class chromebook extends laptop {
26         }
27         // buat objek dari class chromebook (instansiasi)
28         $gadget_baru = new chromebook();
29         echo "Belajar OOP PHP <br />";
30     ?>
31 </body>
32 </html>

```

2. Buka browser, kemudian ketikkan alamat :  
[http://localhost/sewa\\_buku/OOP/parent\\_constructor.php](http://localhost/sewa_buku/OOP/parent_constructor.php)



Constructor dari class komputer  
 Belajar OOP PHP  
 Destructor dari class komputer

#### 4. 6. Latihan Overridden Constructor dan Destructor Parent

1. Buatlah file bernama overridden\_constructor\_destructor.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

C: > xampp >htdocs > sewa\_buku > OOP > overridden\_constructor\_destructor.php

```
1  <?php
2  // buat class komputer
3  class mahasiswa {
4      // buat constructor class komputer
5      public function __construct() {
6          echo "Constructor dari class mahasiswa <br />";
7      }
8      // buat destructor class komputer
9      public function __destruct() {
10         echo "Destructor dari class mahasiswa <br />";
11     }
12 }
13 // turunkan class mahasiswa ke mhs_informatika
14 class mhs_informatika extends mahasiswa {
15     // buat constructor class mhs_informatika
16     public function __construct() {
17         echo "Constructor dari class mhs_informatika <br />";
18     }
19     // buat destructor class mhs_informatika
20     public function __destruct() {
21         echo "Destructor dari class mhs_informatika";
22     }
23 }
24 // turunkan class mhs_informatika ke kelas_reguler
25 class kelas_reguler extends mhs_informatika {
26     // buat constructor class kelas_reguler
27     public function __construct() {
28         echo "Constructor dari class kelas_reguler <br />";
29     }
30     // buat destructor class kelas_reguler
31     public function __destruct() {
32         echo "Destructor dari class kelas_reguler <br />";
33     }
34 }
35 // buat objek dari class kelas_reguler (instansiasi)
36 $mhs_baru = new kelas_reguler();
37 echo "Belajar OOP PHP <br />";
38 ?>
39
```

2. Buka browser, kemudian ketikkan alamat :
- [http://localhost/sewa\\_buku/OOP/overridden\\_constructor\\_destructor.php](http://localhost/sewa_buku/OOP/overridden_constructor_destructor.php)



Constructor dari class kelas\_reguler  
Belajar OOP PHP  
Destructor dari class kelas\_reguler

#### 4. 7. Latihan Enkapsulasi pada Pewarisan

1. Buatlah file bernama file bernama enkapsulasi\_pewarisan.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```
C: > xampp > htdocs > sewa_buku > OOP > ✎ enkapsulasi_pewarisan.php
1  <?php
2      // The parent class
3      class Mahasiswa {
4          private $nim;
5          public function setNim($nim)
6          {
7              $this -> nim = $nim;
8          }
9      }
10     class mhs_informatika extends Mahasiswa{
11         public function hello()
12         {
13             return "Halo NIM saya <b>" . $this -> nim . "</b><br />";
14         }
15     }
16     $mhs_informatika1 = new mhs_informatika();
17     $mhs_informatika1 -> setNim('123456789');
18     echo $mhs_informatika1 -> hello();
19     ?>
20
```

2. Buka browser, kemudian ketikkan alamat :  
[http://localhost/sewa\\_buku/OOP/enkapsulasi\\_pewarisan.php](http://localhost/sewa_buku/OOP/enkapsulasi_pewarisan.php)

Notice: Undefined property: mhs\_informatika::\$nim in C:\xampp\htdocs\sewa\_buku\OOP\enkapsulasi\_pewarisan.php on line 13  
Halo NIM saya

#### 4. 8. Latihan Enkapsulasi pada Pewarisan 2

1. Buatlah file bernama file bernama enkapsulasi\_pewarisan2.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```

C: > xampp > htdocs > sewa_buku > OOP > enkapsulasi_pewarisan2.php
1  <?php
2      class Mahasiswa {
3          protected $nim;
4          public function setNim($nim)
5          {
6              $this -> nim = $nim;
7          }
8      }
9      class mhs_informatika extends Mahasiswa{
10         public function hello()
11         {
12             return "Halo NIM saya <b>" . $this -> nim . "</b><br />";
13         }
14     }
15     $mhs_informatika1 = new mhs_informatika();
16     $mhs_informatika1 -> setNim('123456789');
17     echo $mhs_informatika1 -> hello();
18 ?>
19

```

2. Buka browser, kemudian ketikkan alamat :
- [http://localhost/sewa\\_buku/OOP/enkapsulasi\\_pewarisan2.php](http://localhost/sewa_buku/OOP/enkapsulasi_pewarisan2.php)

← → ↻ ⓘ localhost/sewa\_buku/OOP/enkapsulasi\_pewarisan2.php

---

Halo NIM saya **123456789**

#### 4. 9. Latihan Getter dan Setter

1. Buatlah file bernama file bernama `getter_setter.php` pada folder `C:\xampp\htdocs\sewa_buku\OOP\`, kemudian ketikkan barisan kode berikut :

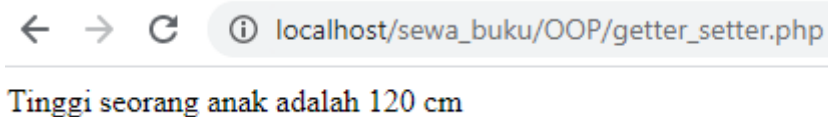
```

C: > xampp > htdocs > sewa_buku > OOP > getter_setter.php
1  <?php
2      class Anak_Anak
3      {
4          private $tinggi;
5          public function setTinggi($tinggi)
6          {
7              if ($tinggi > 30)
8              {
9                  $this->tinggi= $tinggi;
10             }
11         }
12         public function getTinggi()
13         {
14             echo "Tinggi seorang anak adalah " . $this->tinggi . " cm";
15         }
16     }
17     $anak1= new Anak_Anak;
18     $anak1->setTinggi(120);
19     $anak1->getTinggi();
20     ?>
21

```

2. Buka browser, kemudian ketikkan alamat :

[http://localhost/sewa\\_buku/OOP/getter\\_setter.php](http://localhost/sewa_buku/OOP/getter_setter.php)



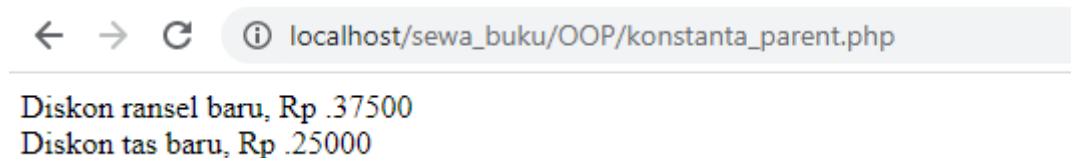
Tinggi seorang anak adalah 120 cm

#### 4. 10. Latihan Mengakses Konstanta Parent

1. Buatlah file bernama file bernama konstanta\_parent.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```
C: > xampp > htdocs > sewa_buku > OOP > konstanta_parent.php
1  <?php
2      class tas {
3          // buat konstanta class tas
4          const DISKON = 0.1;
5      }
6      // turunkan class tas ke class ransel
7      class ransel extends tas {
8          // buat konstanta class ransel
9          const DISKON = 0.15;
10         public function beli_tas($harga){
11             return "Diskon tas baru, Rp .". $harga*parent::DISKON;
12         }
13         public function beli_ransel($harga){
14             return "Diskon ransel baru, Rp .". $harga*self::DISKON;
15         }
16     }
17     // buat objek dari class ransel (instansiasi)
18     $ransel_baru=new ransel();
19     echo $ransel_baru->beli_ransel(250000);
20     echo "<br />";
21     echo $ransel_baru->beli_tas(250000);
22 ?>
23
```

2. Buka browser, kemudian ketikkan alamat :
- [http://localhost/sewa\\_buku/OOP/konstanta\\_parent.php](http://localhost/sewa_buku/OOP/konstanta_parent.php)



#### 4. 11. Latihan Objek Interface

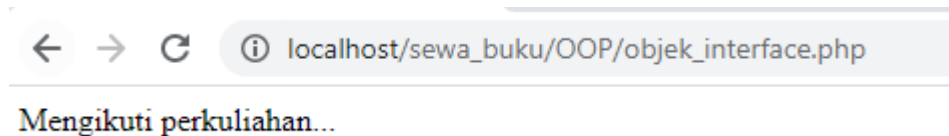
1. Buatlah file bernama objek\_interface.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```

C: > xampp > htdocs > sewa_buku > OOP > objek_interface.php
1  <?php
2      interface mahasiswa{
3          public function masuk_kuliah();
4          public function praktikum();
5      }
6      class mhs_informatika implements mahasiswa{
7          public function masuk_kuliah(){
8              return "Mengikuti perkuliahan...";
9          }
10         public function praktikum(){
11             return "Melakukan prkatikum...";
12         }
13     }
14     $mhs_baru = new mhs_informatika();
15     echo $mhs_baru->masuk_kuliah();
16     ?>
17

```

2. Buka browser, kemudian ketikkan alamat :  
http://localhost/sewa\_buku/OOP/objek\_interface.php



Mengikuti perkuliahan...

4. 12. Latihan Objek Interface dengan Method Protected
  1. Buatlah file bernama file bernama objek\_interface\_protected.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```

C: > xampp > htdocs > sewa_buku > OOP > objek_interface_protected.php
1  <?php
2      interface mahasiswa{
3          public function masuk_kuliah();
4          protected function praktikum();
5      }
6      class mhs_informatika implements mahasiswa{
7          public function masuk_kuliah(){
8              return "Mengikuti perkuliahan...";
9          }
10         public function praktikum(){
11             return "Melakukan prkatikum...";
12         }
13     }
14     $mhs_baru = new mhs_informatika();
15     echo $mhs_baru->masuk_kuliah();
16     ?>
17

```

2. Buka browser, kemudian ketikkan alamat :
- [http://localhost/sewa\\_buku/OOP/objek\\_interface\\_protected.php](http://localhost/sewa_buku/OOP/objek_interface_protected.php)



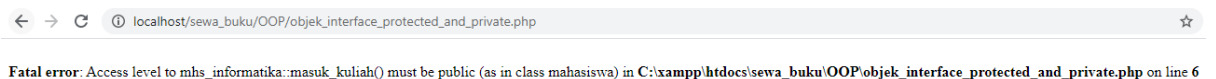
4. 13. Latihan Objek Interface dengan Method Protected dan Private
1. Buatlah file bernama file bernama objek\_interface\_protected\_and\_private.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```

C: > xampp > htdocs > sewa_buku > OOP > objek_interface_protected_and_private.php
1  <?php
2      interface mahasiswa{
3          public function masuk_kuliah();
4          public function praktikum();
5      }
6      class mhs_informatika implements mahasiswa{
7          protected function masuk_kuliah(){
8              return "Mengikuti perkuliahan...";
9          }
10         private function praktikum(){
11             return "Melakukan prkatikum...";
12         }
13     }
14     $mhs_baru = new mhs_informatika();
15     echo $mhs_baru->masuk_kuliah();
16     ?>
17

```

2. Buka browser, kemudian ketikkan alamat :
- [http://localhost/sewa\\_buku/OOP/objek\\_interface\\_protected\\_and\\_private.php](http://localhost/sewa_buku/OOP/objek_interface_protected_and_private.php)



#### 4. 14. Latihan Objek Interface yang diturunkan

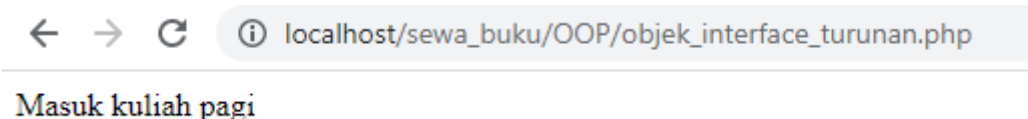
1. Buatlah file bernama file bernama objek\_interface\_turunan.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

```

C: > xampp > htdocs > sewa_buku > OOP > objek_interface_turunan.php
1  <?php
2      interface mahasiswa{
3          public function masuk_kuliah();
4          public function praktikum();
5      }
6      interface mahasiswa_reguler extends mahasiswa{
7          public function kelas_pagi();
8      }
9      class mhs_informatika implements mahasiswa_reguler{
10         public function masuk_kuliah(){
11             return "Mengikuti kuliah...";
12         }
13         public function praktikum(){
14             return "Melakukan praktikum...";
15         }
16         public function kelas_pagi(){
17             return "Masuk kuliah pagi";
18         }
19     }
20     $mhs_baru = new mhs_informatika();
21     echo $mhs_baru->kelas_pagi();
22     ?>
23

```

2. Buka browser, kemudian ketikkan alamat :
- [http://localhost/sewa\\_buku/OOP/objek\\_interface\\_turunan.php](http://localhost/sewa_buku/OOP/objek_interface_turunan.php)



Masuk kuliah pagi

#### 4. 15. Latihan Polimorfisme

1. Buatlah file bernama file bernama polimorfisme.php pada folder C:\xampp\htdocs\sewa\_buku\OOP\, kemudian ketikkan barisan kode berikut :

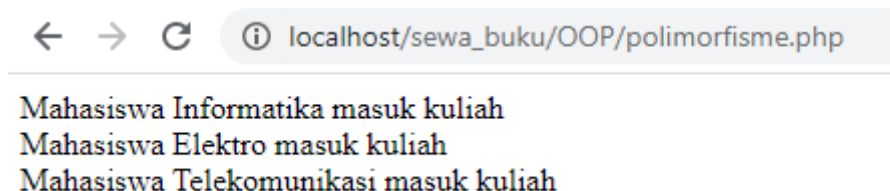


```

C: > xampp >htdocs > sewa_buku > OOP > polimorfisme.php
1  <?php
2      abstract class mahasiswa{ // buat abstract class
3          abstract public function masuk_kuliah();// buat abstract method
4      }
5      class mhs_informatika extends mahasiswa{
6          public function masuk_kuliah(){
7              return "Mahasiswa Informatika masuk kuliah";
8          }
9      }
10     class mhs_elektro extends mahasiswa{
11         public function masuk_kuliah(){
12             return "Mahasiswa Elektro masuk kuliah";
13         }
14     }
15     class mhs_telekomunikasi extends mahasiswa{
16         public function masuk_kuliah(){
17             return "Mahasiswa Telekomunikasi masuk kuliah";
18         }
19     }
20     // buat objek dari class diatas
21     $mhs_informatika_baru = new mhs_informatika();
22     $mhs_elektro_baru = new mhs_elektro();
23     $mhs_telekomunikasi_baru = new mhs_telekomunikasi();
24     // buat fungsi untuk memproses objek
25     function mhs_masuk_kuliah($objek_komputer){
26         return $objek_komputer->masuk_kuliah();
27     }
28     // jalankan fungsi
29     echo mhs_masuk_kuliah($mhs_informatika_baru);echo "<br />";
30     echo mhs_masuk_kuliah($mhs_elektro_baru);echo "<br />";
31     echo mhs_masuk_kuliah($mhs_telekomunikasi_baru);
32     ?>

```

2. Buka browser, kemudian ketikkan alamat :
- [http://localhost/sewa\\_buku/OOP/polimorfisme.php](http://localhost/sewa_buku/OOP/polimorfisme.php)



[←](#) [→](#) [↻](#) [ⓘ](#) localhost/sewa\_buku/OOP/polimorfisme.php

Mahasiswa Informatika masuk kuliah  
 Mahasiswa Elektro masuk kuliah  
 Mahasiswa Telekomunikasi masuk kuliah

## V. Lembar Kerja

1. Lakukan praktikum sesuai langkah kerja yang diberikan.

## VI. Latihan Soal

(Tidak ada)

## **VII. Kesimpulan**

Buatlah kesimpulan dari praktek yang sudah anda lakukan

