

# Laporan Tugas

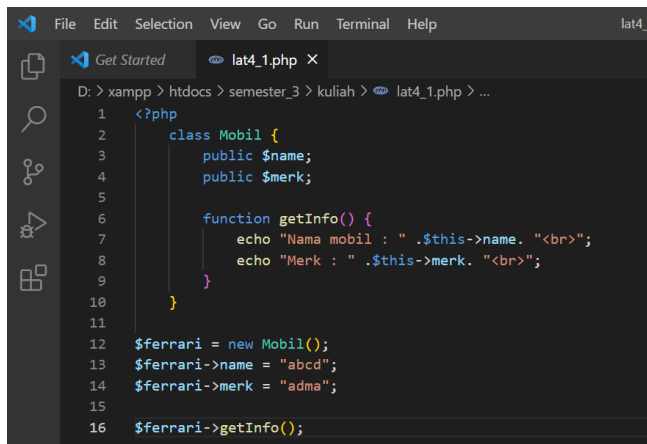
## Pemrograman Framework

Nama : Bagus Kuncoro Aji

NIM : G.211.21.0080

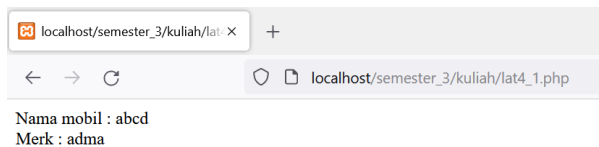
Link GitHub : <https://github.com/Aji-Use/P3-Project1-OOP>

### Latihan 1



```
1 <?php
2 class Mobil {
3     public $name;
4     public $merk;
5
6     function getInfo() {
7         echo "Nama mobil : " . $this->name. "<br>";
8         echo "Merk : " . $this->merk. "<br>";
9     }
10 }
11
12 $ferrari = new Mobil();
13 $ferrari->name = "abcd";
14 $ferrari->merk = "adma";
15
16 $ferrari->getInfo();
```

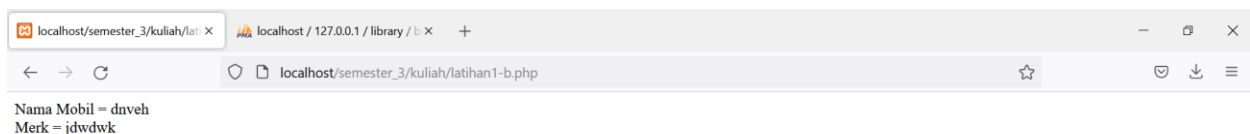
#### a. Tampilan



b. Buatlah sebuah method overload getInfo dengan parameter \$a, jalankan dan amati perubahan yang terjadi!

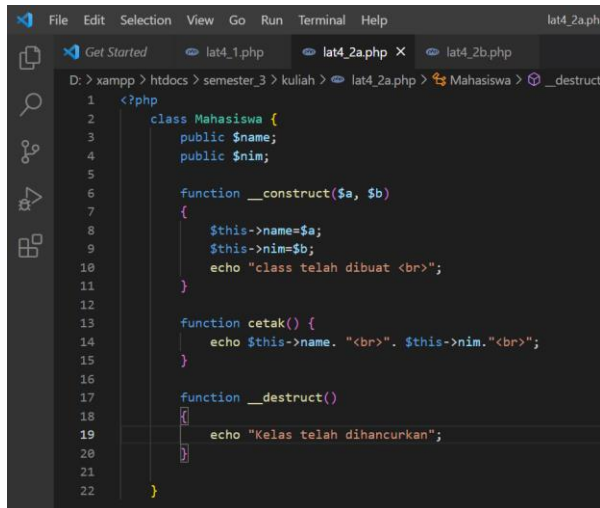


```
1 <?php
2 class Mobil {
3     public $nama;
4     public $merk;
5
6     function getInfo($a) {
7         echo "Nama Mobil = " . $a->nama . "<br>";
8         echo "Merk = " . $a->merk . "<br>";
9     }
10 }
11
12 $mobil = new Mobil();
13 $mobil->nama = "dnveh";
14 $mobil->merk = "jdwddk";
15 $mobil->getInfo($mobil);
```

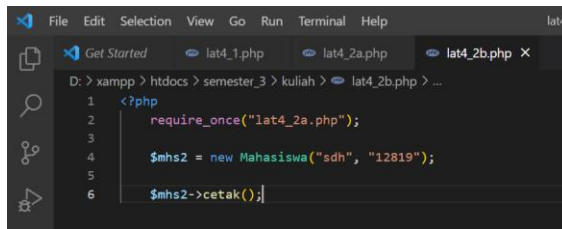


## Latihan 2

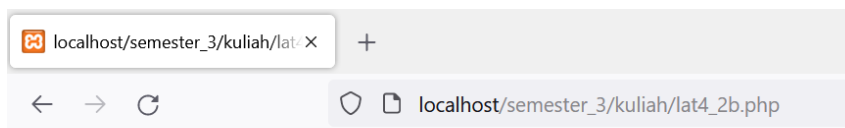
- Sebelum dirubah



```
1 <?php
2 class Mahasiswa {
3     public $name;
4     public $nim;
5
6     function __construct($a, $b)
7     {
8         $this->name=$a;
9         $this->nim=$b;
10        echo "class telah dibuat <br>";
11    }
12
13    function cetak() {
14        echo $this->name. "<br>". $this->nim."<br>";
15    }
16
17    function __destruct()
18    {
19        echo "Kelas telah dihancurkan";
20    }
21
22 }
```

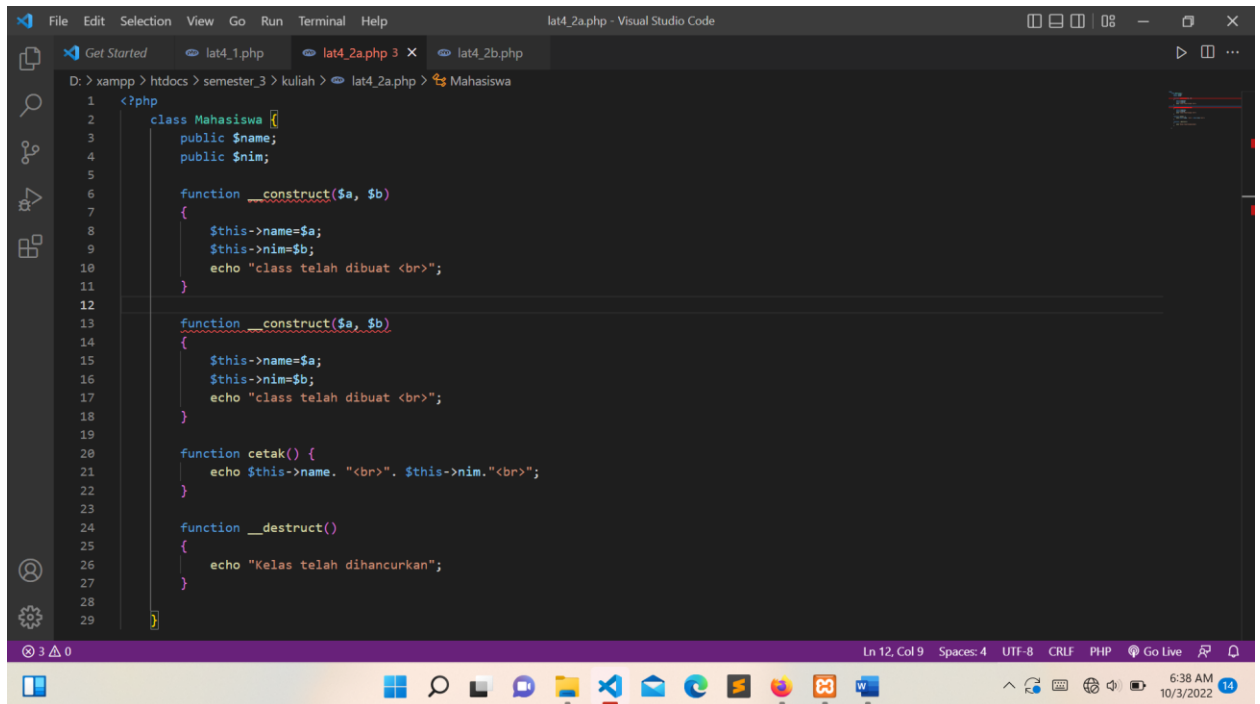


```
1 <?php
2 require_once("lat4_2a.php");
3
4 $mhs2 = new Mahasiswa("sdh", "12819");
5
6 $mhs2->cetak();
```



class telah dibuat  
sdh  
12819  
Kelas telah dihancurkan

## Setelah dirubah



```
1 <?php
2 class Mahasiswa {
3     public $name;
4     public $nim;
5
6     function __construct($a, $b)
7     {
8         $this->name=$a;
9         $this->nim=$b;
10        echo "class telah dibuat <br>";
11    }
12
13    function __construct($a, $b)
14    {
15        $this->name=$a;
16        $this->nim=$b;
17        echo "class telah dibuat <br>";
18    }
19
20    function cetak() {
21        echo $this->name. "<br>". $this->nim."<br>";
22    }
23
24    function __destruct()
25    {
26        echo "Kelas telah dihancurkan";
27    }
28
29 }
```

Fatal error: Cannot redeclare Mahasiswa::\_\_construct() in D:\xampp\htdocs\semester\_3\kuliah\lat4\_2a.php on line 12

**Kesimpulan:** Dalam satu kelas tidak bisa diberi Construct lebih dari satu.

## Latihan 3

Sebelum dirubah

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



**Fatal error:** Uncaught Error: Cannot access protected property Mahasiswa::\$name in D:\xampp\htdocs\semester\_3\kuliah\lat4\_3b.php:5 Stack trace: #0 {main} thrown in D:\xampp\htdocs\semester\_3\kuliah\lat4\_3b.php on line 5

```
2 class Mahasiswa {
3     private $name;
4     private $nim;
5
6     function __construct($a = "Name",
7                         $b = "Nim")
8     {
9         $this->name = $a;
10        $this->nim = $b;
11    }
12
13    function setName($a) {
14        $this->name=$a;
15    }
16
17    function setNim($b) {
18        $this->nim=$b;
19    }
20
21    function getName() {
22        return $this->name;
23    }
24
25    function getNim() {
26        return $this->nim;
27    }
28
29    function destruct() {}
30 }
```

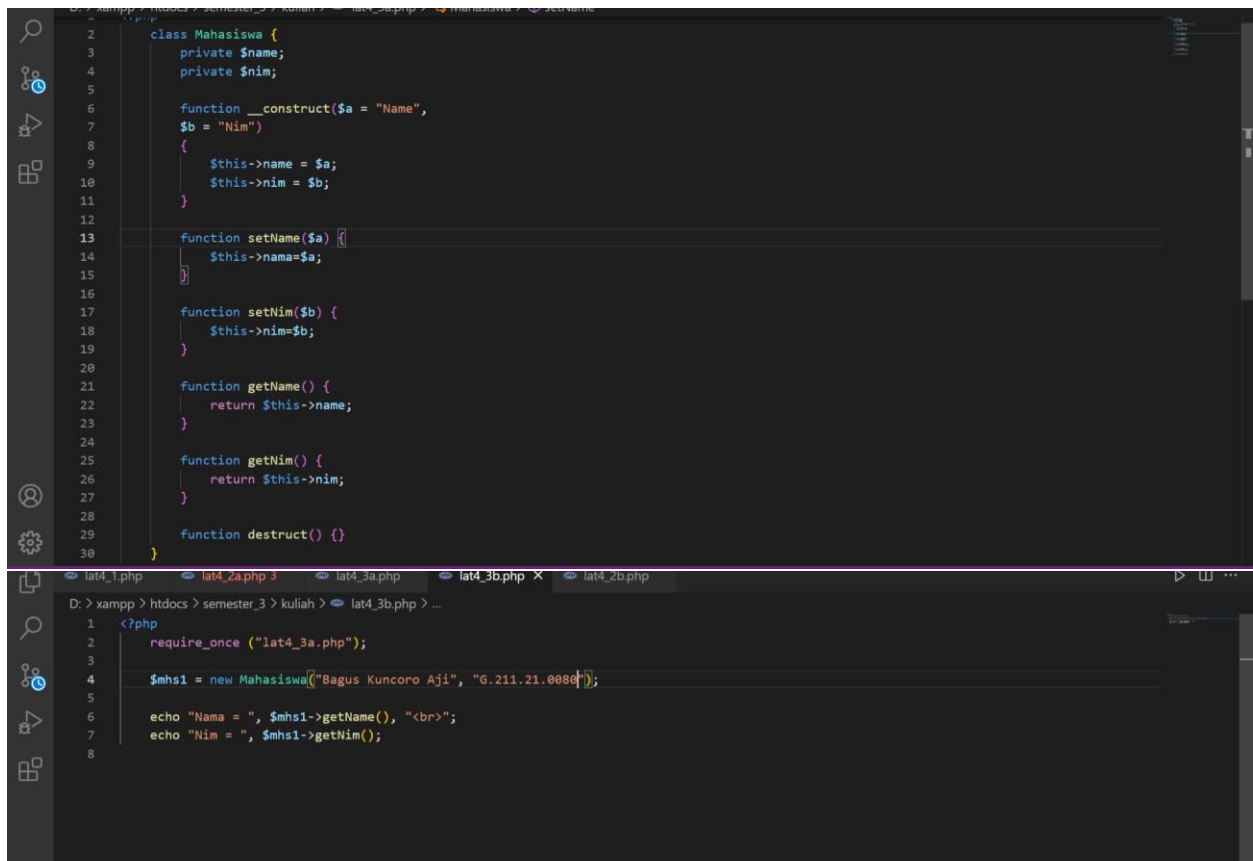
a. Apakah program error? Jika error mengapa hal itu dapat terjadi?

*Program error karena modifier menggunakan private yang hanya bisa diakses didalam class*

b. Rubahlah modifier dari variable nama dan nim menjadi protected dan public. Lalu amati perubahan yang terjadi?

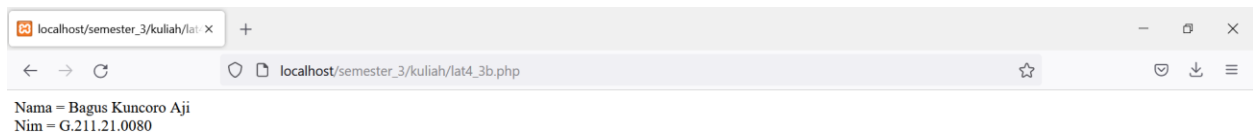
*Jika modifier diganti menggunakan public maka script akan berjalan sesuai perintah, karena modifier public dapat diakses dimana saja.*

c. Modifikasilah Lat3\_b sehingga dapat memberi dan mencetak isi dari nim dan nama dengan modifier private.



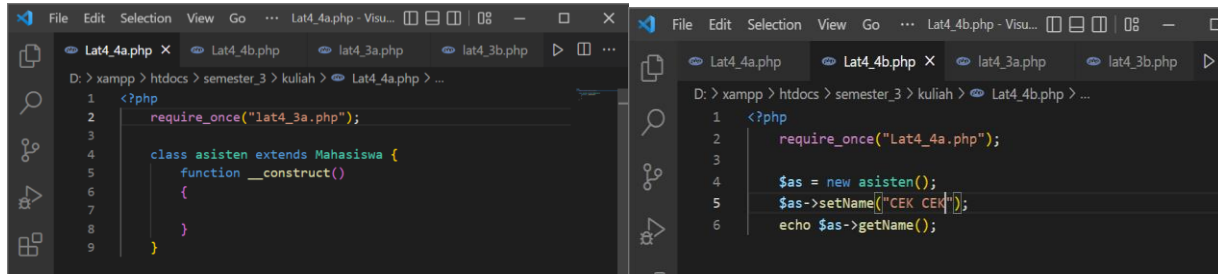
```
1 class Mahasiswa {
2     private $nama;
3     private $nim;
4
5     function __construct($a = "Name",
6         $b = "Nim")
7     {
8         $this->nama = $a;
9         $this->nim = $b;
10    }
11
12    function setName($a) {
13        $this->nama=$a;
14    }
15
16    function setNim($b) {
17        $this->nim=$b;
18    }
19
20    function getName() {
21        return $this->nama;
22    }
23
24    function getNim() {
25        return $this->nim;
26    }
27
28    function destruct() {}
29 }
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

```
1 <?php
2 require_once ("lat4_3a.php");
3
4 $mhs1 = new Mahasiswa("Bagus Kuncoro Aji", "G.211.21.0080");
5
6 echo "Nama = ", $mhs1->getName(), "<br>";
7 echo "Nim = ", $mhs1->getNim();
8
```



d. Modifer private hanya bisa digunakan dengan cara memanggil method yang berada didalam kelasnya

## Latihan 4

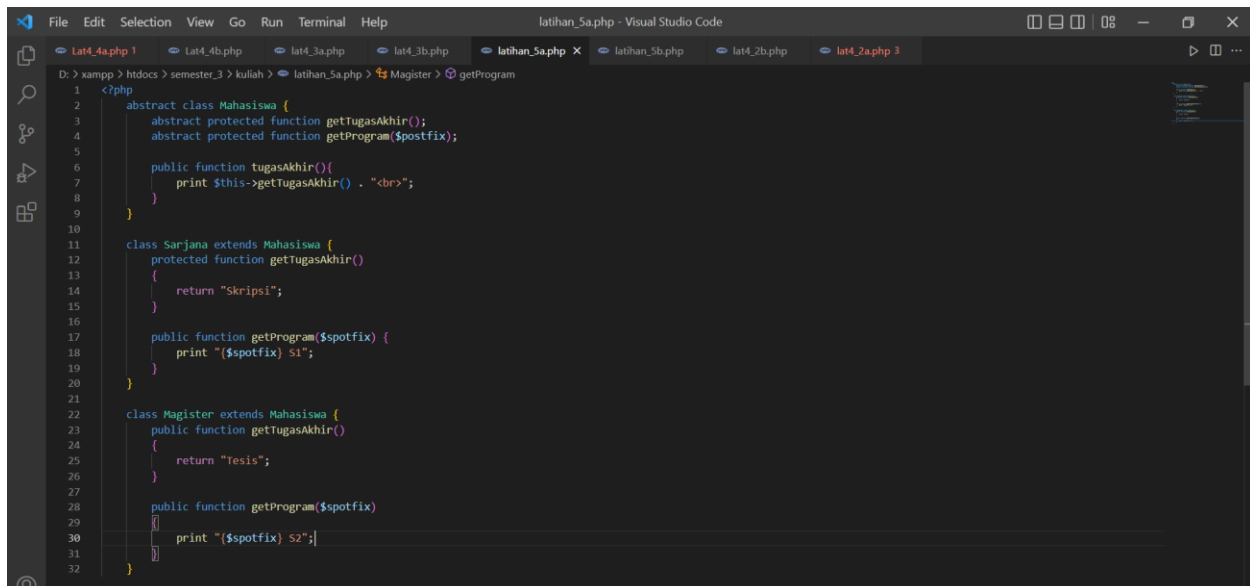


```
Lat4_4a.php
1 <?php
2 require_once("lat4_3a.php");
3
4 class asisten extends Mahasiswa {
5     function __construct() {
6     }
7 }
8
9

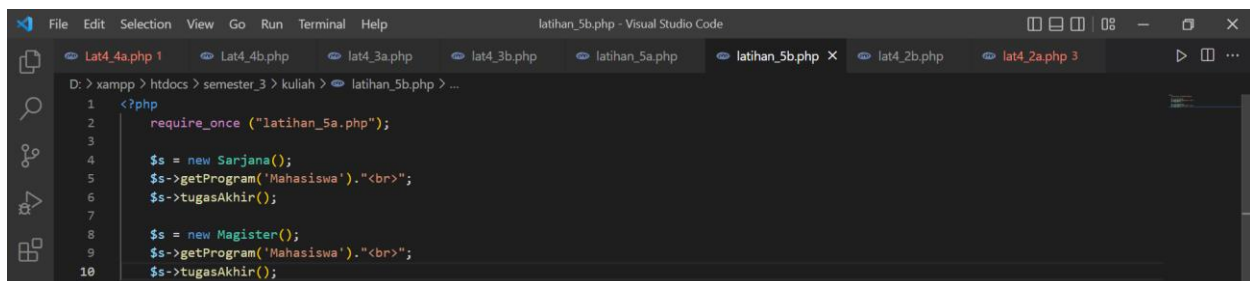
Lat4_4b.php
1 <?php
2 require_once("Lat4_4a.php");
3
4 $as = new asisten();
5 $as->setName("CEK CEK");
6 echo $as->getName();
```

*Sub class/ child class dapat mewarisi method dari class atau parentnya. Atau bisa disebut inheritance*

## Latihan 5

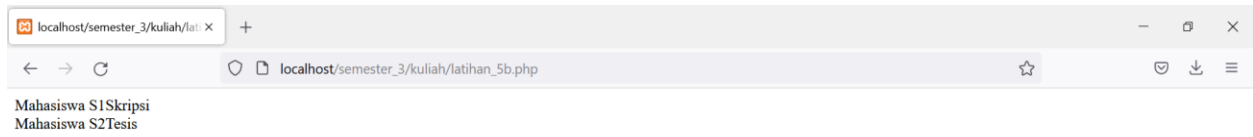


```
latihan_5a.php - Visual Studio Code
1 <?php
2 abstract class Mahasiswa {
3     abstract protected function getTugasAkhir();
4     abstract protected function getProgram($postfix);
5
6     public function tugasAkhir(){
7         print $this->getTugasAkhir() . "<br>";
8     }
9
10
11 class Sarjana extends Mahasiswa {
12     protected function getTugasAkhir()
13     {
14         return "Skripsi";
15     }
16
17     public function getProgram($spotfix) {
18         print "({$spotfix} s1";
19     }
20 }
21
22 class Magister extends Mahasiswa {
23     public function getTugasAkhir()
24     {
25         return "Tesis";
26     }
27
28     public function getProgram($spotfix)
29     {
30         print "({$spotfix} s2";
31     }
32 }
```

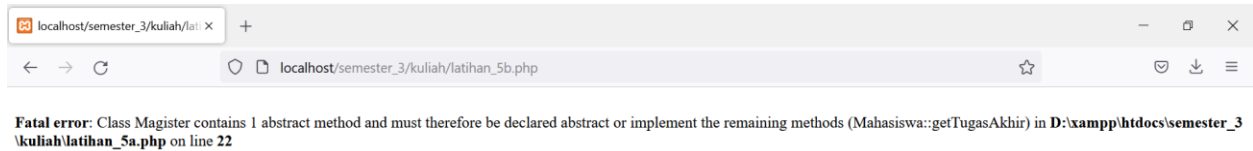
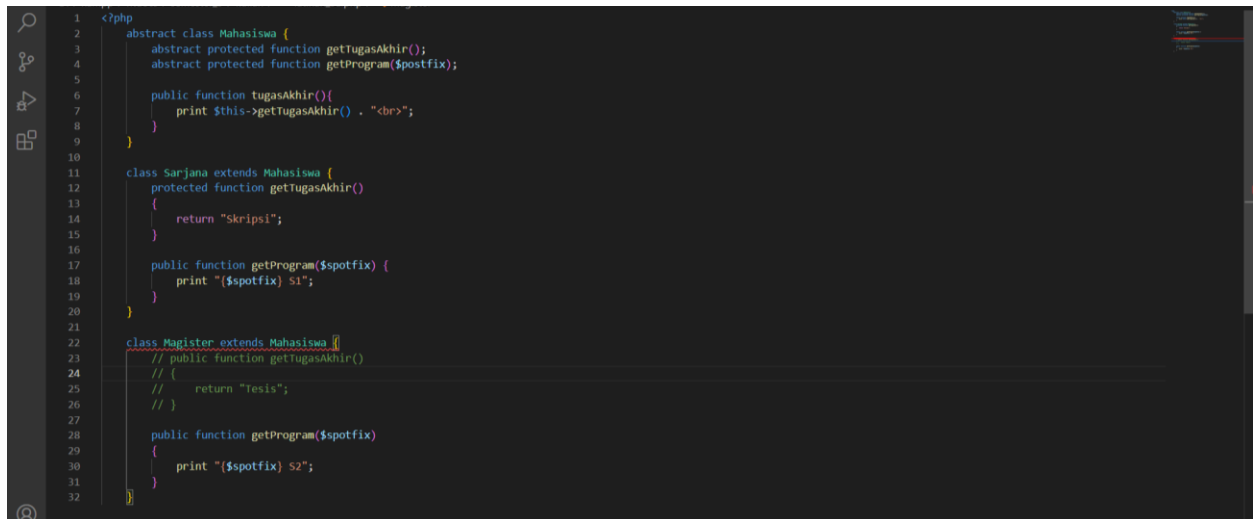


```
latihan_5b.php - Visual Studio Code
1 <?php
2 require_once ("latihan_5a.php");
3
4 $s = new Sarjana();
5 $s->getProgram('Mahasiswa')."<br>";
6 $s->tugasAkhir();
7
8 $m = new Magister();
9 $m->getProgram('Mahasiswa')."<br>";
10 $m->tugasAkhir();
```

a. Bagaimana hasil tampilan dari program di atas?



b. Hapuslah kode baris 23 – 26 pada Latihan\_5a.php, lalu jalankan Latihan\_5b.php. Bagaimana hasil tampilan program di atas? Jelaskan mengapa hal tersebut terjadi?



Ouput error, karena method `getTugasAkhir()` tidak ada di turunannya sedangkan method dengan keyword `abstract` hanya bisa dipanggil/diakses apabila diturunannya/subclass terdapat method `getTugasAkhir()`

c. Kesimpulan

Class `abstract` hanya bisa diakses di class turunannya

## Latihan 6

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

a. Jelaskan maksud dari program di atas?

Interface a b c merupakan sebuah class yang didalamnya terdapat sebuah method berupa method foo(), bar(), dan baz() tetapi tidak boleh memiliki properti didalamnya dan method tersebut harus dalam bentuk abstract. Sedangkan class d merupakan implements dari class c dan dimana class c juga merupakan turunan dari interface a dan b. Di class d ini boleh menempatkan properti di dalamnya seperti modifier private maupun protected dan mengisi semua method yang digunakan diparentnya.

b. Hapuslah kode baris 27-29, lalu jalankan lat6.php. Bagaimana tampilan program di atas? Jelaskan mengapa hal tersebut dapat terjadi?

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

localhost/semester\_3/kuliah/lati x +  
localhost/semester\_3/kuliah/latihan\_6.php

**Fatal error:** Class d contains 1 abstract method and must therefore be declared abstract or implement the remaining methods (c::baz) in D:\xampp\htdocs\semester\_3\kuliah\latihan\_6.php on line 14

*Terjadi error karena Kelas d berisi 1 metode abstrak*



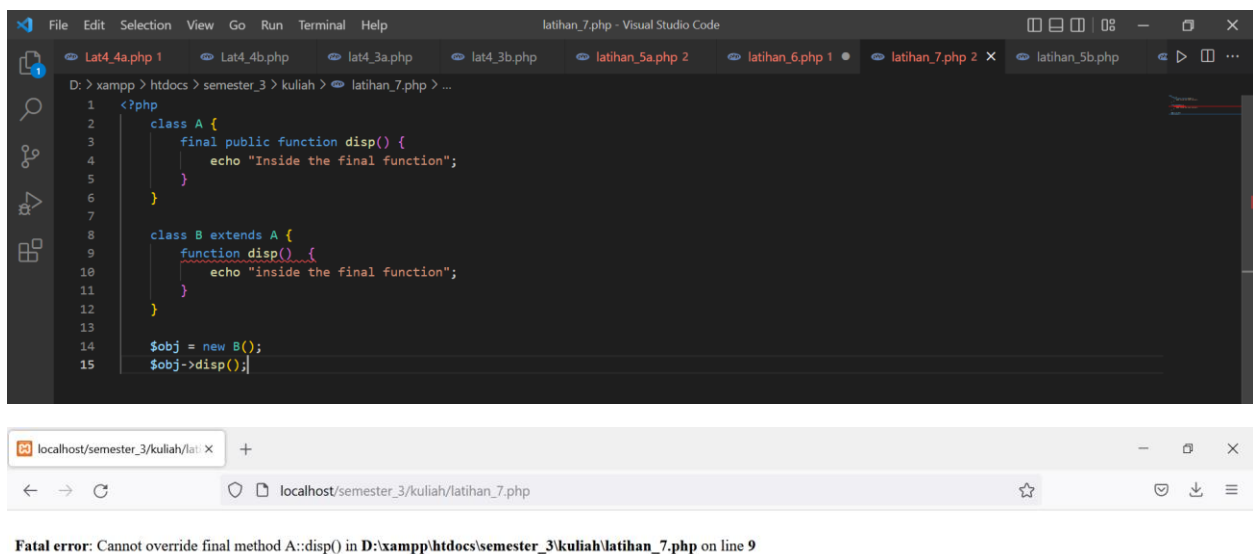
c. Dari contoh kode diatas, buatlah class baru dengan nama “e” yang mempunyai method foo dan bar

```
14 class d implements c {
15     public function foo() {
16     }
17
18     public function bar() {
19     }
20
21     // public function baz() {
22     // }
23 }
24
25 class e implements a, b {
26     public function foo() {
27     }
28     public function bar() {
29     }
30 }
31
32 }
```

d. Simpulkan apa yang anda peroleh dari Latihan 6! Jawab:

*Sebuah interface dengan method abstract harus mempunyai class dengan implements yang sama dengan parentnya.*

## Latihan 7



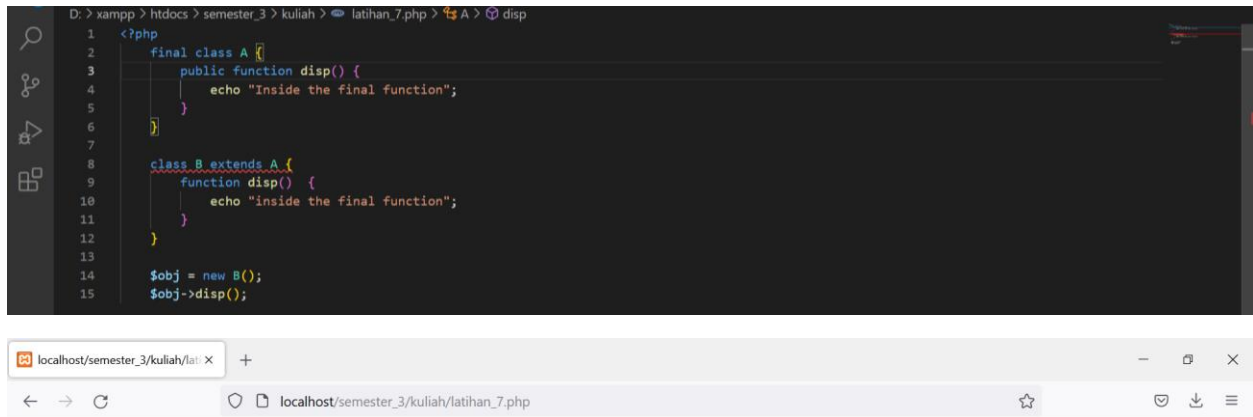
```
1 <?php
2 class A {
3     final public function disp() {
4         echo "Inside the final function";
5     }
6 }
7
8 class B extends A {
9     function disp() {
10         echo "inside the final function";
11     }
12 }
13
14 $obj = new B();
15 $obj->disp();
```

Fatal error: Cannot override final method A::disp() in D:\xampp\htdocs\semester\_3\kuliah\latihan\_7.php on line 9

a. Bagaimana tampilan program di atas? Jelaskan mengapa hal tersebut terjadi?

Output error karena kelas A tidak dapat di override

b. Modifikasi program di atas dengan menghapus kata final pada kode baris 5 dan menambahkan kata final pada baris 2. Bagaimana tampilan program di atas? Jelaskan mengapa hal tersebut terjadi?



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

localhost/semester\_3/kuliah/lati X +

localhost/semester\_3/kuliah/latihan\_7.php

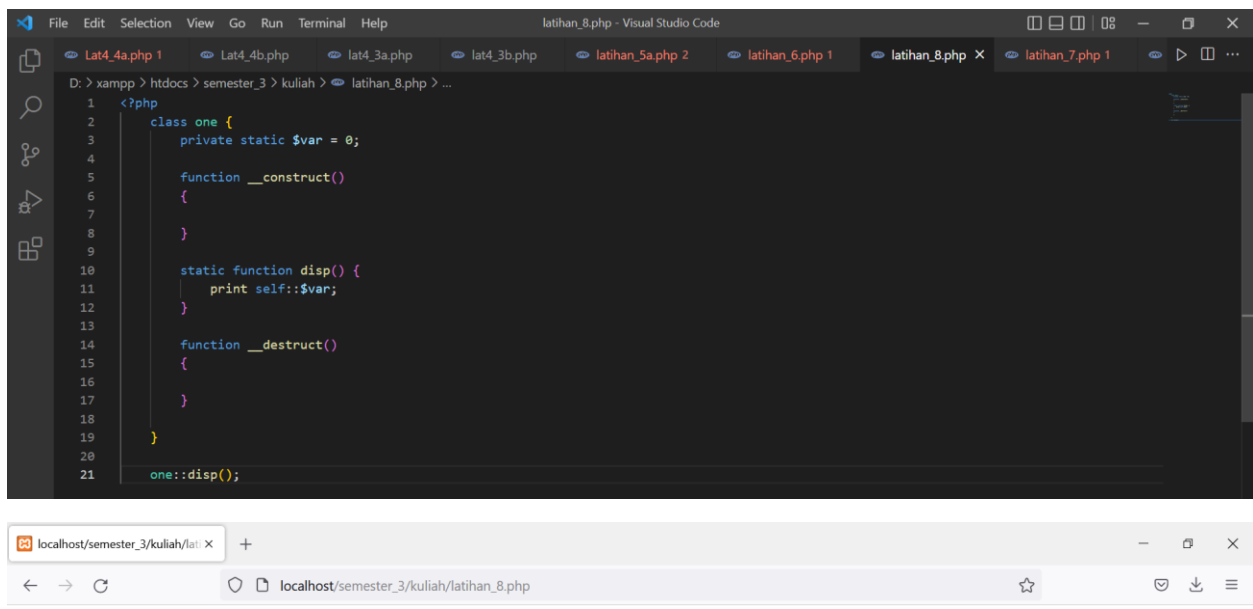
**Fatal error:** Class B cannot extend final class A in D:\xampp\htdocs\semester\_3\kuliah\latihan\_7.php on line 8

Keyword final memang digunakan agar kelas tidak bisa dioverride

c. Simpulkan apa yang anda peroleh dari latihan 7!

Keyword final, akan mencegah proses overriding method pada class anak (sub-class).

## Latihan 8



```
1 <?php
2 class one {
3     private static $var = 0;
4
5     function __construct()
6     {
7
8     }
9
10    static function disp() {
11        print self::$var;
12    }
13
14    function __destruct()
15    {
16
17    }
18
19 }
20
21 one::disp();
```

localhost/semester\_3/kuliah/lati X +

localhost/semester\_3/kuliah/latihan\_8.php

keyword static digunakan untuk mencetak method tanpa melakukan instansiasi, yaitu dengan memanggil class nya kemudian titik dua, dua kali lalu method nya atau properti yang ingin dipanggil

## Latihan 9

### Controllers

```
1 <?php
2
3 class Library {
4     private $model;
5
6     function __construct()
7     {
8         $this->model = new LibraryModel();
9     }
10
11     public function index() {
12         $data['listBuku'] = $this->model->getLibrary();
13         return $this->render('daftar-buku', $data);
14     }
15
16     private function render($view, $data=[]) {
17         if ($data) {
18             foreach ($data as $key=>$value) {
19                 $$key = $value;
20             }
21         }
22         require_once __DIR__ . '/../views/' . $view . '.php'
23     }
24 }
```

### Models.php

```
1 <?php
2
3 class LibraryModel {
4     private $connection;
5
6     function __construct()
7     {
8         $this->connection = new mysqli(
9             'localhost', 'root', 'library'
10        );
11    }
12
13
14    public function getLibrary($id = 1) {
15        if($id) {
16            $query = "SELECT * FROM book WHERE id = $id";
17            $result = $this->connection->query($query);
18            if($result->num_rows == 0) {
19                return false;
20            } else{
21                return $result->fetch_assoc();
22            }
23        } else{
24            $query = "SELECT * FROM book";
25            $result = $this->connection->query($query);
26            if ($result->num_rows == 0) {
27                return false;
28            } else{
29                while($row = $result->fetch_assoc()){
30                    $result[] = $row;
31                }
32                return $result;
33            }
34        }
35    }
36 }
```

