

PRAKTIKUM
PEMROGRAMAN WEB
SEMESTER GENAP TAHUN AKADEMIK 2021/2022

Tanggal

.....

Materi

OBJECT ORIENTED PROGRAMMING (OOP)

.....

PRODI SISTEM INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS NUSANTARA PGRI KEDIRI

BAB I

DASAR TEORI

Object Oriented Programming (OOP)

Dua istilah yang paling penting untuk OOP adalah Class dan Object. Class adalah generalisasi definisi sesuatu. Class dapat di pandang sebagai cetak biru object yaitu implementasi yang spesifik dari sesuatu hal. Salah satu prinsip OOP adalah modularitas yaitu memecah aplikasi menjadi submodul tertentu. Website banyak melakukan hal tersebut misalnya: berinteraksi dengan database, menangani formulir, mengirim email, menghasilkan HTML, dll. Masing-masing hal tersebut dapat dijadikan modul, yaitu Class. Kode dapat dikembangkan secara mandiri dengan memisahkan yang tidak terkait (walaupun berinteraksi) elemen. Disamping itu juga mempermudah membuat perawatan dan update kode, dan menyederhanakan debugging.

* Class

Secara sintak, definisi class dimulai dengan kata class, yang diikuti dengan nama class tersebut. Nama class tidak dapat menggunakan reserved word. Selanjutnya nama class diikuti dengan kurung kurawal buka dan tutup.

Method dalam class diawali dengan kata function, yang diikuti dengan nama method serta argumen. Atribut dalam class didefinisikan sebagai variabel yang terdiri 3 jenis yaitu: public, private dan protected.

Luaran dari sebuah method dituliskan menggunakan kata return.

Contoh:

```
class matematika{
    function tambah($a,$b){
        $c = $a + $b;
        return $c;
    }
}
```

* Object

OOP digunakan melalui 2 tahap. Pertama, mendefinisikan class. Kedua, menggunakan class tersebut dengan membuat sebuah object (class instance). Untuk membuat object digunakan kata new. Contoh = \$object = new ClassName();

Untuk memanggil methods menggunakan sintak object operator.

Contoh = \$object → methodName();. Untuk mengakses properti object digunakan sintak \$object → propertyName;

Contoh:

```
class matematika{
    function tambah($a,$b){
        $c = $a + $b;
        return $c;
    }
}

$math = new matematika();
$bil1 = 7;
$bil2 = 5;
$hasil = $math → tambah($bil1, $bil2);
echo $hasil;
```

* This

Sebuah method dimungkinkan untuk mengakses properti atau method dalam class yang sama menggunakan kata this.

Contoh :

```
class matematika {  
    public $pi = 3.14;  
    function tambah ($a, $b) {  
        $c = $a + $b;  
        return $c;  
    }  
    function kuadrat ($x) {  
        return $x * $x;  
    }  
    function keliling_lingkaran ($r) {  
        $kel = 2 * $this->$pi * $r;  
        return $kel;  
    }  
    function luas_lingkaran ($r) {  
        $luas = $this->$pi * $this->kuadrat($r);  
        return $luas;  
    }  
}
```

* Konstruktor

Konstruktor merupakan method yang dieksekusi secara otomatis pertama kali saat sebuah class diturunkan menjadi sebuah objek.

Contoh :

```
class matematika () {  
    public $pi;  
    function __construction () {  
        $this->$pi = 3.14;  
    }  
    function keliling_lingkaran ($r) {  
        $kel = 2 * $this->$pi * $r;  
        return $kel;  
    }  
}
```


BAB II

PERCOBAAN DAN LATIHAN

*PERCOBAAN 1

```
<?php
class matematika{
    public $pi = 3.14;
    function tambah ($a, $b){
        $c = $a + $b;
        return $c;
    }
    function kuadrat ($x){
        return $x * $x;
    }
    function keliling_lingkaran ($r){
        $kel = 2 * $this->$pi * $r;
        return $kel;
    }
    function luas_lingkaran ($r){
        $luas = $this->$pi * $this->kuadrat ($r);
        return $luas;
    }
}

$math = new matematika();
$jari = 10;
$kel_lingkaran = $math->keliling_lingkaran ($jari);
$luas_lingkaran = $math->luas_lingkaran ($jari);
echo "Menghitung keliling dan Luas lingkaran <br>";
echo "Jari-Jari : " . $jari . "<br>";
echo "Keliling = " . $kel_lingkaran . "<br>";
echo "Luas = " . $luas_lingkaran;
```

*PERCOBAAN 2

```
<?php
class matematika{
    public $pi = 3.14;
    function tambah ($a, $b){
        $c = $a + $b;
        return $c;
    }
    function kuadrat ($x){
        return $x * $x;
    }
    function keliling_lingkaran ($r){
        $kel = 2 * $this->$pi * $r;
        return $kel;
    }
    function luas_lingkaran ($r){
        $luas = $this->$pi * $this->kuadrat ($r);
        return $luas;
    }
}

?>
```


* Percobaan 3

< ? php

```
include 'class-matematika.php';
```

```
$math = new matematika();
```

```
$jari = 10;
```

```
$kel_lingkaran = $math->keliling_lingkaran($jari);
```

```
$luas_lingkaran = $math->luas_lingkaran($jari);
```

```
echo "Menghitung keliling dan Luas Lingkaran <br>"
```

```
echo "Jari-Jari : ". $jari . " <br>"
```

```
echo "Keliling = ". $kel_lingkaran . " <br>"
```

```
echo "Luas = ". $luas_lingkaran;
```

?>

* Latihan 1.

< ? PHP

class balok {

private \$p; private \$l; private \$t;

function l-balok (\$p, \$l) {

\$this->p = \$p; \$this->l = \$l;

echo "luas = " . \$p . " x " . \$l . " = " . \$this->p * \$this->l . " cm ²";

}

function hitung-balok (\$p, \$l, \$t) {

\$this->t = \$t; \$this->p = \$p; \$this->l = \$l;

echo "volume = " . \$p . " x " . \$l . " x " . \$t . " = " . \$this->p * \$this->l * \$this->t . " cm ³";

}

function form-hitungbalok () {

echo '< form method = "GET">';

echo '< h1> luas dan volume balok </ h1>';

echo '< h2> isi data </ h2>';

echo '< br> panjang : ';

echo '< br>< input type = "text" name = "p" placeholder = "masukkan nilai"> cm < br>';

echo '< br> lebar : ';

echo '< br>< input type = "text" name = "l" place holder = "masukkan nilai"> cm < br>';

echo '< br> tinggi : ';

echo '< br>< input type = "text" name = "t" placeholder = "masukkan nilai"> cm < br>';

echo '< p style = "margin-left : 15 px"> < input type = "submit" name = "submit" value = "hitung" >';

echo '< / form>';

}

}

\$hitung-balok = new balok;

\$l-balok = new balok;

\$hitung-balok->form-hitungbalok();

if (isset(\$_GET['submit'])) {

\$p = \$_GET['p']; \$l = \$_GET['l']; \$t = \$_GET['t'];

echo \$l-balok->l-balok (\$p, \$l);

echo "< br>";

echo \$hitung-balok->hitung-balok (\$p, \$l, \$t);

}

?>

* Latihan 2

< ? php

```
class bola {
    private $r;
    public $phi = 3.14;
    function luas_bola ($r) {
        $this->r = $r; $p2 = $this->r * $this->r;
        $luas = 4 * $this->phi * $p2;
        echo 'luas = 4 x ' . $this->phi . ' x ' . $this->r . '<sup>2</sup> = ' .
            ($luas) . ' cm <sup>2</sup> <br>';
    }
    function vbola ($r) {
        $this->r = $r; $p3 = $this->r * $this->r * $this->r;
        $volume = 4/3 * $this->phi * $p3;
        echo 'volume = 4/3 x ' . $this->phi . ' x ' . $this->r . '
            <sup>3</sup> = ' . ($volume) . ' cm <sup>3</sup> <br>';
    }
    function form_hitungbola() {
        echo '<form method = "GET" >';
        echo "<h1>hitung luas dan volume bola </h1>";
        echo 'jari-jari = <input type = "text" name = "r"
            placeholder = "masukkan jari-jari"> <br>';
        echo '<br><input type = "submit" name = "submit" value = "hitung">';
        echo '</form>';
    }
}
```

```
}
$bola = new bola;
$bola->form_hitungbola();
if (isset($_GET['r'])) {
    $r = $_GET['r'];
    echo "jari-jari = " . $r . " cm <br>";
    echo $bola->luas_bola($r);
    echo $bola->vbola($r);
}
?>
```


* Latihan 3

<? php

```
class konversiNilai {
```

```
    public $nilai;
```

```
    function nilai ($Z) {
```

```
        $this->nilai = $Z;
```

```
    }
```

```
    function konversi () {
```

```
        if ( $this->nilai <= 100 && $this->nilai >= 85 ) { $grade = 'A';
```

```
        } elseif ( $this->nilai < 85 && $this->nilai >= 77 ) { $grade = 'B';
```

```
        } elseif ( $this->nilai < 77 && $this->nilai >= 61 ) { $grade = 'C';
```

```
        } else if ( $this->nilai < 61 && $this->nilai >= 50 ) { $grade = 'D';
```

```
        } else { $grade = 'E';
```

```
        } return $grade;
```

```
    }
```

```
    function form-nilai () {
```

```
        echo '<form method = "GET">';
```

```
        echo "< h1> konversi nilai </h1>";
```

```
        echo '<input type = "text" name = "nilai" placeholder = "masukkan nilai">';
```

```
        echo '<br><input type = "submit" name = "konversi">';
```

```
        echo '</form>';
```

```
    }
```

```
}
```

```
$score = new konversiNilai;
```

```
$score->form-nilai ();
```

```
if (isset($_GET['konversi'])) {
```

```
    echo 'nilai: ' . $_GET['nilai'];
```

```
    echo "<br> nilai dikonversi = ";
```

```
    $x = $score->nilai ($_GET['nilai']);
```

```
    echo $score->konversi ();
```

```
}
```

```
?>
```


BAB III

TAMPILAN PROGRAM

Percobaan 1



Percobaan 2



Latihan 1

luas dan volume balok

isi data

panjang : cm
lebar : cm
tinggi : cm

hitung

luas = $25 \times 10 = 250 \text{ cm}^2$
volume = $25 \times 10 \times 15 = 3750 \text{ cm}^3$

Latihan 2

hitung luas dan volume bola

jari-jari : cm

hitung

luas = $4 \times 3.14 \times 21^2 = 11316.96 \text{ cm}^2$
volume = $\frac{4}{3} \times 3.14 \times 21^3 = 38772.72 \text{ cm}^3$

Latihan 3

latihan3.html

Konversi Nilai

masukkan nilai

Substitusi

nilai : 90
nilai dikonversi : A

latihan3.html

Konversi Nilai

masukkan nilai

Substitusi

nilai : 40
nilai dikonversi : E

BAB IV

KESIMPULAN

Jadi kesimpulanya di oop terdapat beberapa script yaitu :

1. class

2. Object

contoh: `object = new className();`

3. this

this disini untuk mengakses properti atau method.

4. Constructor

merupakan method yang dieksekusi secara otomatis.

Dari kesimpulan dapat dijelaskan bahwa :

- mahasiswa dapat memahami konsep object oriented programming (oop)
- mahasiswa dapat memahami oop menggunakan PHP
- mahasiswa dapat membuat program sederhana menggunakan oop PHP.