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

OOP adalah class dan object. Class adalah generalisasi definisi sesuatu. Class dapat dipanding sebagai cetak biru objek 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 diberikan modul yaitu class. Kode dapat dikembang secara mandiri dengan memisahkan yang tidak terkait (walaupun berinteraksi) elemen. Disamping itu juga mempermudah membuat perawatan dan update kode. dan menyederhanakan debungging

⇒ 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 diawah dengan kata function yang diikuti dengan nama method serta argumen. Atribut dalam class didefinisikan sebata 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 objek digunakan kata new. Contoh
$object = new classname ();
Untuk memanggil methods menggunakan sintak object → contoh
$object → methodName (); untuk mengakses property object digunakan
sintaks $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
        $kel = 2 * $this -> $PI * $r;
        return $kel;
    }
    function luas_lingkaran ($r) {
        $luas = $this -> PI * $this -> kuadrat ($r);
        return $luas;
    }
}
```

⇒ Constructor

constructor adalah method yang dieksekusi secara otomatis pertamakali 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;
    }
}
```

Percobaan 1

```php
<?php
class matematika {
    Private $Pi = 3.19;
    Function tambah ($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

⇒ Class_matematika.php
```php
<?php
Class matematika {
    Private $Pi = 3.19
    function tambah ($a, $b) {
        $c = $a + $b;
        return $c;
    }
    Function kuadrat ($x) {
        return $x * $x;
```

```php
                }
        function keliling_lingkaran ($r) {
                $kel = 2 * $this -> Pi * $r;
                return $kel;
        }
        function luas_lingkaran ($r) {
                $luas = $this -> Pi * $this -> kuadrat ($r);
                return $luas;
        }
}
?>
```

=> lingkaran.php

```php
<?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 Lingkungan <br>";
        echo "Jari-Jari :".$jari."<br>";
        echo "Keliling =".$kel_lingkaran."<br>";
        echo "Luas =".$luas_lingkaran;
?>
```

Latihan 1

```php
<?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 <sup>2</sup>";
    }

    Function hitung_balok ($P, $l, $t) {
        $this -> P = $P; $this -> p. $P; $this -> l = $l;
        echo "Volume =".$P."x".$l."x".$t" =". $this -> p * $this
        -> l * $this -> t." cm <sup>3</sup>";
    }

    Function form_hitungbalok () {
        echo '<Form method = "GET">';
        echo "<h1> luas dan volume balok </h1>";
        echo '<h3> isi data </h3>';
        echo '<br> Panjang :';
        echo '<br> <input type = "text' name = "P" Placeholder =
        "masukkan nilai"> cm <br>';
        echo '<br> lebar :';
```

```php
        echo '<br><input type = "text" name = "l" Placeholder = "
        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" >';
        }
    }

    $hitung_balok = new balok ;
    $_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
<?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 ;
        $voume = 4/3 * $this -> phi * $P3 ;
        echo 'volume = 4/3 x ', $this -> phi .'x'.$this->r. '<sup>3
        </sup> = ', ($voume). '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'])) {
```

```php
        $r = $_GET ['r'];
        echo "Jari-Jari = ".$r." cm <br>";
        echo $bola -> was_bola ($r);
        echo $bola -> vbola ($r);
    }
?>
```

Latihan 3

```php
<? 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 >= 75) { $grade = 'B';
        }elseif ($this -> nilai < 75 && $this -> nilai >= 65) {$grade = 'C';
        }elseif ($this -> nilai < 65 && $this -> nilai >= 55) {$grade = 'D';
        } else {$grade = 'E';
        } return $grade;
    }
    function form_nilai () {
        echo '<Form method = "GET">';
        echo "<h1> konversi Nilai </h1>";
        echo '<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 :
[masukkan nilai] cm

lebar :
[masukkan nilai] cm

tinggi :
[masukkan nilai] cm

[hitung]

luas = 8 x 6 = 48 cm^2

volume = 8 x 6 x 10 = 480 cm^3

Latihan 2

hitung luas dan volume bola

jari-jari = [masukkan jari-jari]

[hitung]

jari-jari = 14cm

luas = 4 x 3.14 x 14^2 = 2461.76 Cm^2

volume = 4.3 x 3.14 x 14^3 = 11488.213333333 Cm^3

Latihan 3

Konversi Nilai

masukkan nilai

Submit

nilai : 70
nilai dikonversi : C
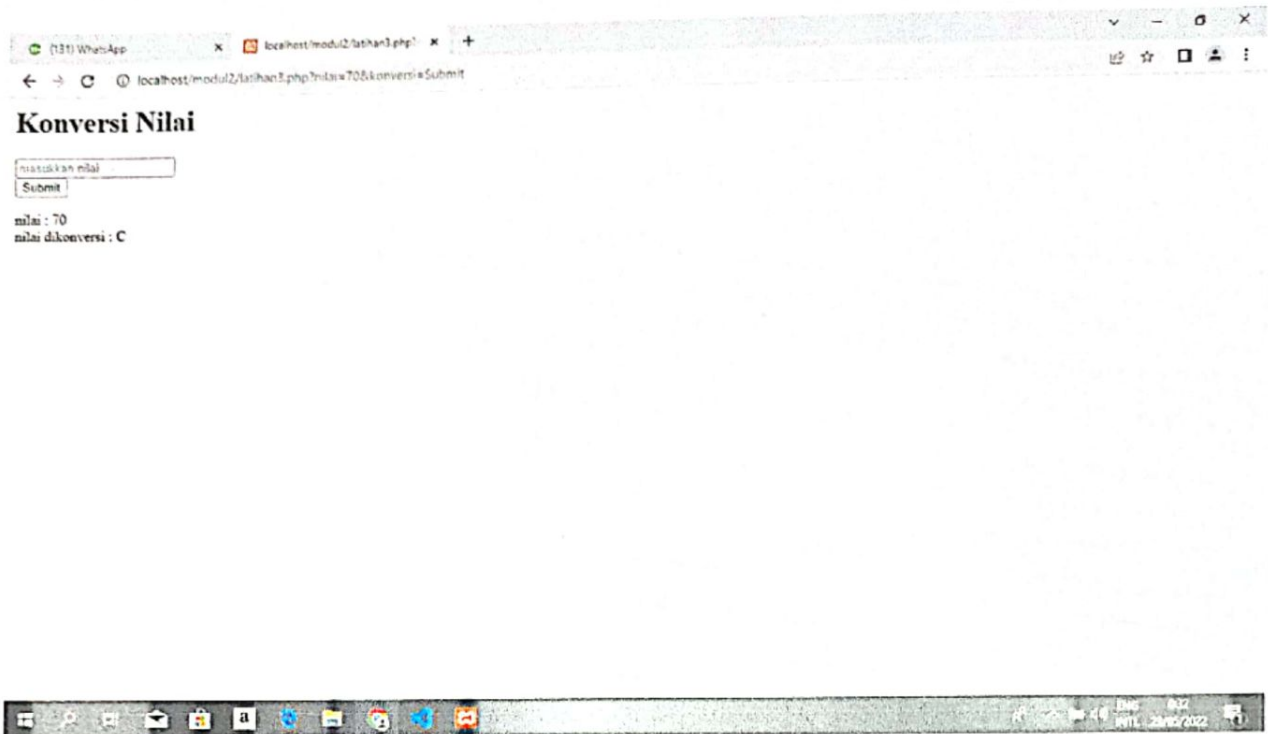
BAB IV

KESIMPULAN

Kesimpulan dan OOP dapat beberapa Sricpt yaitu: Class, Object, This, dan Constructor. Jadi 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.