

**PRAKTIKUM 4**  
**OVERRIDING DAN OVERLOADING**

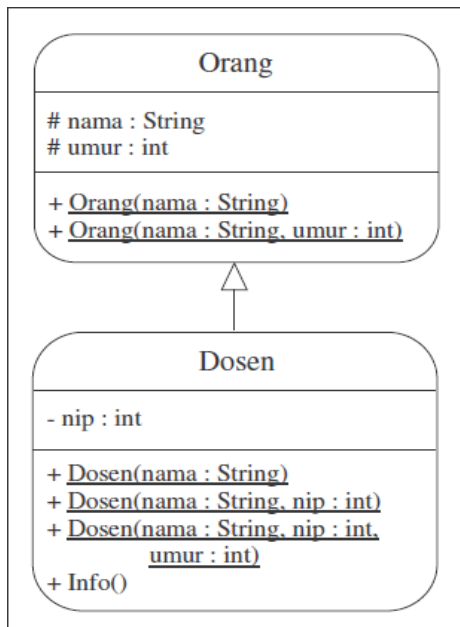


Oleh:  
Handika Dio Pradana  
21091397021  
2021A

**PROGRAM STUDI D4 MANAJEMEN INFORMATIKA**  
**FAKULTAS VOKASI**  
**UNIVERSITAS NEGERI SURABAYA 2021**

## TUGAS PRAKTIKUM

1. Buat program berdasarkan UML berikut



Source Code:

```
body {
    align-items: center;
}
h2{
    text-align: center;
}

.container{
    margin: auto;
    text-align: center;
    width: 350px;
    border-style: groove;
}
```

```

<!DOCTYPE html>
<html lang="id">
<head>
    <link rel="stylesheet" type="text/css" href="prak4 no1.css">
    <title>Praktikum 4</title>
</head>
<body>
<div class="container">
<h2>Soal No.1</h2>
<?php
    class Orang {
        protected $nama,$umur;
        public function __construct($nama,$umur){
            $this->nama = $nama;
            $this->umur = $umur;
        }
    }

    class Dosen extends Orang {
        private $nip;
        public function __construct($nama, $umur, $nip) {
            $this->nip = $nip;
            parent::__construct($nama, $umur);
        }

        public function hasil() {
            return $this->nama . ' , umur ' . $this->umur . ' , NIP ' . $this->nip;
        }
    }

    $dosen1 = new Dosen('Rahadian Bisma', 35, 2015041003);
    $dosen2 = new Dosen('Dini Kinati Fardah', 35,2015042006 );
    $dosen3 = new Dosen('Ferly Isnomo Abdi', 30, 2019031016);

    echo $dosen1->hasil();
    echo ' <br> <br>';
    echo $dosen2->hasil();
    echo ' <br> <br>';
    echo $dosen3->hasil();
    echo ' <br> <br>';
?>
</div>
</body>
</html>

```

Output:

## Soal No.1

Rahadian Bisma , umur 35 , NIP 2015041003

Dini Kinati Fardah , umur 35 , NIP 2015042006

Ferly Isnomo Abdi , umur 30 , NIP 2019031016

Analisa :

Program berdasarkan studi kasus UML di atas merupakan sebuah praktikum yang menggunakan pewarisan sifat dari orang tua ke anak (Orang ke Dosen) yang mana semua dosen adalah orang dan setiap orang pasti memiliki identitas seperti Nama, Umur dan NIP. Ketika dosen dipanggil maka dosen akan mengambil data yang diwariskan dari orang tuanya yaitu orang.

2. Buat program berdasarkan UML berikut

RerataNilai
+ int average(int, int)
+ double average(double, double)
+ int average(int, int, int)

Source Code :

```
<?php
    require_once 'prak4 no2.php';
?>

<html lang="id">
<head>
    <link rel="stylesheet" href="prak4 no2.css">
    <title>Praktikum 4</title>
</head>
<body>
<div class="container">
<h2>Soal No.2</h2>
<div class="form">
    <form action="" method="post">
        <input type="number" name="input1">
        <input type="number" name="input2">
        <input type="number" name="input3">
        <button type="submit" name="submit">Submit</button>
    </form>
</div>

    <?php
        if (isset($_POST['submit'])) {
            if (!$_POST['input3']) {
                $value1 = new RerataNilai2($_POST['input1'], $_POST['input2']);
                echo $value1->average(0);
            } else {
                $value1 = new RerataNilai2($_POST['input1'], $_POST['input2']);
                echo $value1->average($_POST['input3']);
            }
        }
    ?>
</div>
</body>
</html>
```

```
.container {
    margin: auto;
    text-align: center;
    width: 650px;
    height: 150px;
    border-style: groove;
}
```

```
<?php
class RerataNilai {
    protected $value, $value2;
    public function __construct($value, $value2){
        $this->value = $value;
        $this->value2 = $value2;
    }

    public function average($value3 = 0){
        if ($value3 == 0) {
            $str = $this->value . ' + ' . $this->value2 . ' / 2 = ';
            return $str . ($this->value += $this->value2 ) / 2;
        } else {
            $str = '(' . $this->value . ' + ' . $this->value2 . ' + ' . $this->value2 . ')' / 3 = ';
            return $str . ($this->value += $this->value2 += $value3 ) / 3;
        }
    }
}

class RerataNilai2 extends RerataNilai{
    public function __construct($value, $value2){
        parent::__construct($value, $value2);
    }

    public function average($value3 = 0){
        return parent::average($value3);
    }
}

?>
```

### Soal No.2

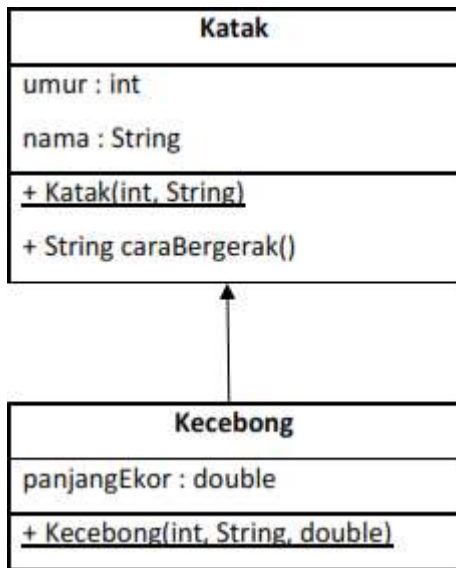
$(2 + 2 + 2) / 3 = 2$

Analisa:

Program berdasarkan studi kasus UML di atas merupakan pewarisan sifat dari orang tua ke anak. Tetapi dalam hal ini orang tua hanya mewariskan cara menghitung rata-ratanya saja adapun yang mengeksekusi bilangannya tetap anak dengan panutan warisan orang tua tersebut. Pada program ini menghitung sebuah

rata-rata dari 3 nilai yang diinputkan oleh user kemudian dibagi 3 untuk mendapatkan hasil rata-rata nilai tersebut.

3. Buat program berdasarkan UML berikut



Dan buat objek dengan karakteristik seperti tabel dibawah ini

Obyek	umur	nama	panjangEkor	caraBergerak
O1	5	Froggy		melompat
O2	2	Junior Frog	10	berenang

```
1 h2{
2   margin: auto;
3   text-align: center;
4   padding-bottom: 10px;
5 }
6
7 .table{
8   margin: auto;
9   font-family: sans-serif;
10  border-collapse: collapse;
11 }
12
13 .table, td{
14   border: 1px solid;
15   padding: 8px 10px;
16 }
17
18 thead{
19   font-weight: bold;
20 }
```

```

<?php
    require_once 'prak4 no3.php';
?>

<html lang="id">
<head>
    <link rel="stylesheet" type="text/css" href="prak4 no3.css">
    <title>Praktikum 4</title>
</head>
<body>
<h2><center>Soal No.3</center></h2>
<table align="center">

<tr align="center">
    <thead>
        <td>Objek</td>
        <td>Umur</td>
        <td>Nama</td>
        <td>Panjang Ekor</td>
        <td>Cara Bergerak</td>
    </thead>
</tr>

<tbody>
<tr>
    <td>01</td>
    <td><?=$katak->getUmur() ?> </td>
    <td><?=$katak->getnama() ?> </td>
    <td>-</td>
    <td><?=$katak->carab Bergerak() ?> </td>
</tr>

<tr>
    <td>02</td>
    <td><?=$kecebong->getUmur() ?></td>
    <td><?=$kecebong->getNama() ?></td>
    <td><?=$kecebong->getpanjangekor() ?></td>
    <td><?=$kecebong->carab Bergerak() ?></td>
</tr>
</tbody>
</table>
</body>
</html>

```

### Soal No.3

Objek	Umur	Nama	Panjang Ekor	Cara Bergerak
01	5	Froggy	-	Melompat
02	5	Junior Froggy	10	Berenang

Analisa :

Program berdasarkan studi kasus UML di atas merupakan pewarisan sifat antara orang tua dan anak. Tetapi ada beberapa hal yang dirubah karena anak tidak puas dengan warisan orang tua tersebut seperti contoh program diatas adalah Katak dan kecebong. Katak adalah orang tua dari kecebong, tetapi ada beberapa hal dari mereka yang tidak sama seperti katak melompat dan kecebong berenang, hal tersebut menunjukkan bahwa terdapat beberapa informasi yang dipakai oleh kecebong dan ada beberapa informasi yang diganti oleh kecebong kemudian ditampilkan oleh class yang lain dengan memanggil objek-objek dari katak maupun kecebong.