

**Tugas Pemrograman Berorientasi Objek**  
**Praktikum 5 (Polimorfisme)**



Disusun oleh :  
Muhammad Yuda Pratama (21091397025)

**Program Studi D4 Manajemen Informatika**  
**Fakultas Vokasi**  
**Universitas Negeri Surabaya**  
**2022**

## 1. Source Code :

```
1  <!-- Muhammad Yuda Pratama - 21091397025 --->
2
3
4  <!DOCTYPE html>
5  <html lang="id">
6  <head>
7      <meta charset="UTF-8">
8      <meta http-equiv="X-UA-Compatible" content="IE=edge">
9      <meta name="viewport" content="width=device-width, initial-scale=1.0">
10     <title>Praktikum 5</title>
11 </head>
12 <body style="align-items: center;">
13     <div class="container">
14         <h2>Soal Nomor 1</h2>
15
16         <?php
17 class Pegawai
18 {
19     public $nama;
20     public function __construct($nama)
21     {
22         $this->nama = $nama;
23     }
24     public function getName()
25     {
26         return $this->nama;
27     }
28 }
```

```
29 class Manajer extends Pegawai
30 {
31     public $tunjangan;
32     public function __construct($nama, $tunjangan)
33     {
34         parent::__construct($nama);
35         $this->tunjangan = $tunjangan;
36     }
37     public function getTunjangan()
38     {
39         return $this->tunjangan;
40     }
41 }
42 class Kurir extends Pegawai
43 {
44     public $gaji;
45
46     public function __construct($nama, $gaji)
47     {
48         parent::__construct($nama);
49
50         $this->gaji = $gaji;
51     }
52     public function getGaji()
53     {
54         return $this->gaji;
55     }
56 }
57 }
```

```

58 class Soal1{
59     public static
60     function Proses($peg){
61         if ($peg instanceof Manajer){
62             $man = $peg;
63             echo "<br>Nama manajer : ".$man->nama, "\n";
64             echo "<br>Tunjangan : Rp. ".strval($man->tunjangan), "\n";
65         }
66         else if ($peg instanceof Kurir){
67             $kur = $peg;
68             echo "<br>Nama kurir : ".$kur->nama, "\n";
69             echo "<br>Gaji : Rp. ".strval($kur->gaji), "\n";
70         }
71     }
72     public static
73     function main($args){
74         echo "Muhammad Yuda Pratama - 21091397025", "\n";
75         echo "<br>";
76         $peg1 = new Manajer("Iwan", 5000000);
77         Soal1::Proses($peg1);
78         echo "<br>";
79         $peg2 = new Kurir("Dimas", 2000000);
80         Soal1::Proses($peg2);
81     }
82 }
83 Soal1::main(array());
84 ?>
85 </div>
86 </body>
87 </html>

```

**Output :**

## Soal 1

Muhammad Yuda Pratama - 21091397025

Nama manajer : Iwan  
Tunjangan : Rp. 5000000

Nama kurir : Dimas  
Gaji : Rp. 2000000

**Analisis :**

Dalam kasus tersebut implementasi dari polymorphic argument berada pada class info dimana method info dibuat static agar pemanggilannya tidak perlu diinisiasi, sehingga bisa langsung dimasukkan menjadi parameter pada method info di class info.

## 2. Source Code :

```
1  <!-- Muhammad Yuda Pratama - 21091397025 --->
2
3  <!DOCTYPE html>
4  <html lang="id">
5  <head>
6      <link rel="stylesheet" type="text/css" href="No2css.css">
7      <meta charset="UTF-8">
8      <meta http-equiv="X-UA-Compatible" content="IE=edge">
9      <meta name="viewport" content="width=device-width, initial-scale=1.0">
10     <title>Praktikum 5</title>
11 </head>
12 <body>
13     <div class="container">
14         <h2>Soal Nomor 2</h2>
15         <?php
16 class Pegawai
17 {
18     public $nama;
19     public $gaji;
20     public function __construct($nama, $gaji)
21     {
22         $this->nama = $nama;
23         $this->gaji = $gaji;
24     }
25     public function infoGaji()
26     {
27         return $this->gaji;
28     }
29 }
```

```
30 class Manajer extends Pegawai
31 {
32     private $tunjangan;
33     public function __construct($nama, $gaji, $tunjangan)
34     {
35         parent::__construct($nama, $gaji);
36         $this->tunjangan = $tunjangan;
37     }
38     public function infoGaji()
39     {
40         return $this->gaji;
41     }
42     public function infoTunjangan()
43     {
44         return $this->tunjangan;
45     }
46 }
```

```

47 class Programmer extends Pegawai
48 {
49     private $bonus;
50     public function __construct($nama, $gaji, $bonus)
51     {
52         parent::__construct($nama, $gaji);
53         $this->bonus = $bonus;
54     }
55     public function infoGaji()
56     {
57         return $this->gaji;
58     }
59     public function infoBonus()
60     {
61         return $this->bonus;
62     }
63 }

```

```

64 class Bayaran
65 {
66     public function hitungBayaran($peg)
67     {
68         $uang = $peg->infoGaji();
69
70         return $uang;
71     }
72     public static function main($args)
73     {
74         echo "Muhammad Yuda Pratama - 21091397025";
75         $man = new Manajer("Iwan", 5000000, 500000);
76         echo "<br>";
77         $prog = new Programmer("Tama", 4000000, 300000);
78         echo "<br>";
79         $hr = new Bayaran();
80         echo "<br> Gaji untuk Manajer Bernama ". $man->nama. " : Rp. ".strval($hr->hitungBayaran($man)) , "\n";
81         echo "<br>", "<br>";
82         echo "<br>Gaji untuk Programmer Bernama ".$prog->nama. " : Rp. ".strval($hr->hitungBayaran($prog)) , "\n";
83     }
84 }
85
86 Bayaran::main(array());
87 ?>
88 <!-- END PHP -->
89 </div>
90 </body>
91 </html>

```

**Output :**

## Soal Nomor 2

Muhammad Yuda Pratama - 21091397025

Gaji untuk Manajer Bernama Iwan : Rp. 5000000

Gaji untuk Programmer Bernama Tama : Rp. 4000000

**Analisis :**

Program ini merupakan penerapan inheritance dengan konsep overriding, dimana pemanggilan construct pada masing-masing class turunan hanya akan menginisiasi property yang dimiliki dengan visibilitas private dan property lain yang diturunkan akan langsung diinisiasi dengan construct dari parentnya.