

LAPORAN UTS
PRAKTIKUM 5



Oleh :
Akhmad Ilham Muhrram (21091397009)
2021A

MANAJEMEN INFORMATIKA UNIVERSITAS NEGERI
SURABAYA
PROGRAM VOKASI
2021

1. Source code

```
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4 | <title>Praktikum 5</title>
5 </head>
6 <body>
7 | <div class="container">
8 | | <style>
9 | | | body {
10 | | | | align-items: center;
11 | | | }
12 | | | h2{
13 | | | | text-align: center;
14 | | | }
15 | | | .container{
16 | | | | margin: auto;
17 | | | | padding: 15px;
18 | | | | width: 350px;
19 | | | | border-style: groove;
20 | | | }
21 | | </style>
22 | | <h2><u>Soal No.1</u></h2>
23 |
24 | <?php
25 | | class Pegawai
26 | | {
27 | | | public $nama;
28 | | | public function __construct($nama)
29 | | | {
30 | | | | $this->nama = $nama;
31 | | | }
32 | | | public function getNama()
33 | | | {
34 | | | | return $this->nama;
35 | | | }
36 | | }
37 | | class Manajer extends Pegawai
38 | | {
39 | | | public $tunjangan;
40 | | | public function __construct($nama, $tunjangan)
41 | | | {
42 | | | | parent::__construct($nama);
43 | | | | $this->tunjangan = $tunjangan;
44 | | | }
45 | | | public function getTunjangan()
46 | | | {
47 | | | | return $this->tunjangan;
48 | | | }
49 | | }
50 | | class Kurir extends Pegawai
51 | | {
52 | | | public $gaji;
53 | | |
54 | | | public function __construct($nama, $gaji)
55 | | | {
56 | | | | parent::__construct($nama);
57 | | | | $this->gaji = $gaji;
58 | | | }
59 | | | public function getGaji()
60 | | | {
61 | | | | return $this->gaji;
62 | | | }
63 | | }
64 | | class Soal1
65 | | {
66 | | | public static
67 | | | function Proses($peg)
68 | | | {
69 | | | | if ($peg instanceof Manajer)
70 | | | | {
71 | | | | | $man = $peg;
72 | | | | | echo "Nama manajer: ".$man->nama, "\n";
73 | | | | | echo "<br>Tunjangan: Rp. ".strval($man->tunjangan), "\n";
74 | | | | }
75 | | | | else if ($peg instanceof Kurir)
76 | | | | {
77 | | | | | $kur = $peg;
78 | | | | | echo "Nama kurir: ".$kur->nama, "\n";
79 | | | | | echo "<br>Gaji: Rp. ".strval($kur->gaji), "\n";
80 | | | | }
81 | | | }
82 | | | public static
83 | | | function main($args)
84 | | | {
85 | | | | $peg1 = new Manajer("Dwi", 5000000);
86 | | | | Soal1::Proses($peg1);
87 | | | | echo "<br>", "<br>";
88 | | | | $peg2 = new Kurir("Sri", 200000);
89 | | | | Soal1::Proses($peg2);
90 | | | }
91 | | }
92 | | Soal1::main(array());
93 | <?>
94 |
95 | </div>
96 </body>
97 </html>
```

Ouput

```

<!DOCTYPE html>
<html lang="id">
<head>
  <title>Praktikum 5</title>
</head>
<body>
  <div class="container">
    <style>
      body {
        align-items: center;
      }
      h2{
        text-align: center;
      }
      .container{
        margin: auto;
        padding: 15px;
        width: 350px;
        border-style: groove;
      }
    </style>
    <h2><u>Soal No.1</u></h2>

    Nama manajer: Dwi
    <br>Tunjangan: Rp. 5000000
    <br><br>Nama kurir: Sri
    <br>Gaji: Rp. 200000

  </div>
</body>
</html>

```

Analisa : Implementasi argumen polimorfik ada di kelas info di mana metode info dibuat statis sehingga tidak perlu memulai panggilan. Mengizinkan input langsung sebagai parameter metode info dari kelas info.

2. Source Code

```
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4 <title>Praktikum 5</title>
5 </head>
6 <body>
7 <div class="container">
8 <style>
9 {
10 align-items: center;
11 }
12 h2{
13 text-align: center;
14 }
15 .container{
16 margin: auto;
17 padding: 15px;
18 width: 350px;
19 border-style: groove;
20 }
21 </style>
22 <h2><?Soal No.2</h2>
23
24 <?php
25 class Pegawai
26 {
27     public $nama;
28     public $gaji;
29     public function __construct($nama, $gaji)
30     {
31         $this->nama = $nama;
32         $this->gaji = $gaji;
33     }
34     public function infoGaji()
35     {
36         return $this->gaji;
37     }
38 }
39 class Manager extends Pegawai
40 {
41     private $tunjangan;
42     public function __construct($nama, $gaji, $tunjangan)
43     {
44         parent::__construct($nama, $gaji);
45         $this->tunjangan = $tunjangan;
46     }
47     public function infoGaji()
48     {
49         return $this->gaji;
50     }
51     public function infoTunjangan()
52     {
53         return $this->tunjangan;
54     }
55 }
56
57 class Programmer extends Pegawai
58 {
59     private $bonus;
60     public function __construct($nama, $gaji, $bonus)
61     {
62         parent::__construct($nama, $gaji);
63         $this->bonus = $bonus;
64     }
65     public function infoGaji()
66     {
67         return $this->gaji;
68     }
69     public function infoBonus()
70     {
71         return $this->bonus;
72     }
73 }
74
75 class Bayaran
76 {
77     public function hitungBayaran($peg)
78     {
79         $uang = $peg->infoGaji();
80         return $uang;
81     }
82     public static function main($args)
83     {
84         $man = new Manager("Dwi", 7000000, 3000000);
85         $prog = new Programmer("Sri", 2000000, 1100000);
86         $br = new Bayaran();
87         echo "<br>Nama Manager : ".$man->nama."<br> Gaji : Rp. ".strval($br->hitungBayaran($man)) , "<br>";
88         echo "<br>";
89         echo "<br>Nama Programmer : ".$prog->nama. "<br> Gaji : Rp. ".strval($br->hitungBayaran($prog)) , "<br>";
90     }
91 }
92
93 Bayaran::main(array());
94
95 </div>
96 </body>
97 </html>
```

Ouput :

```

<!DOCTYPE html>
<html lang="id">
<head>
  <title>Praktikum 5</title>
</head>
<body>
  <div class="container">
    <style>
      body {
        align-items: center;
      }
      h2{
        text-align: center;
      }
      .container{
        margin: auto;
        padding: 15px;
        width: 350px;
        border-style: groove;
      }
    </style>
    <h2><u>Soal No.2</u></h2>

    <br> Nama Manajer : Dwi<br> Gaji : Rp. 7000000
    <br><br>Nama Programmer : Sri<br> Gaji : Rp. 2000000
  </div>
</body>
</html>

```

Analisa :

Program ini menggunakan konsep overriding untuk mengimplementasikan pewarisan. Saat Anda memanggil konstruk

pada setiap kelas turunan, hanya properti yang dimilikinya yang dimulai dengan visibilitas pribadi, properti turunan lainnya mulai langsung pada konstruk kelas induk.