LAPORAN PRAKTIKUM 5
PEMROGRAMAN BERORIENTASI OBJEK

Oleh :
Sayyidina Auliya (21091397034) kelas 2021B

D4 MANAJEMEN INFORMATIKA
FAKULTAS VOKASI
UNIVERSITAS NEGERI SURABAYA
2022

SOURCE CODE DAN ANALISIS

1. Analisis 1

```html
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Praktikum 5</title>
</head>
<body style="align-items: center;">
    <div class="container">
        <h2>Soal 1</h2>

        <?php
class Pegawai
{
        public $nama;
        public  function __construct($nama)
        {
                $this->nama = $nama;
        }
        public  function getNama()
        {
                return $this->nama;
        }
}
class Manajer extends Pegawai
{
        public $tunjangan;
        public  function __construct($nama, $tunjangan)
        {
                parent::__construct($nama);
                $this->tunjangan = $tunjangan;
        }
        public  function getTunjangan()
        {
                return $this->tunjangan;
        }
}
```

```php
class Kurir extends Pegawai
{
        public $gaji;

    public      function __construct($nama, $gaji)
        {
                parent::__construct($nama);

                $this->gaji = $gaji;

        }
        public  function getGaji()
        {
                return $this->gaji;
        }
}
class Soal1
{
        public static
        function Proses($peg)
        {
                if ($peg instanceof Manajer)
                {
                        $man = $peg;
                        echo "<br>Nama manajer: ".$man->nama, "\n";
                        echo "<br>Tunjangan: Rp. ".strval($man->tunjangan), "\n";
                }
                else if ($peg instanceof Kurir)
                {
                        $kur =  $peg;
                        echo "<br>Nama kurir: ".$kur->nama, "\n";
                        echo "<br>Gaji: Rp. ".strval($kur->gaji), "\n";
                }
        }
```

```
78          public static
79          function main($args)
80          {
81              echo "Sayyidina Auliya- 210913970034", "\n";
82          echo "<br>", "<br>";
83              $peg1 = new Manajer("gayatri", 9000000);
84              Soal1::Proses($peg1);
85          echo "<br>", "<br>";
86              $peg2 = new Kurir("ica ", 7500000);
87              Soal1::Proses($peg2);
88          }
89      }
90      Soal1::main(array());
91      ?>
92
93      </div>
94  </body>
95  </html>
```

Analisis :

Kasus pada soal diatas, yaitu implementasi polymorphic argument dengan berada di class info dimana method info dibuat static supaya pemanggilnya tidak memerlukan diinisiasi. Sehingga dapat langsung dimasukkan menjadi parameter pada method info di class info.

2. Analisis 2

```
5
6   <!DOCTYPE html>
7   <html lang="id">
8   <head>
9       <meta charset="UTF-8">
10      <meta http-equiv="X-UA-Compatible" content="IE=edge">
11      <meta name="viewport" content="width=device-width, initial-scale=1.0">
12      <title>Praktikum 5</title>
13  </head>
14  <body>
15      <div class="container">
16          <h2>Soal 2</h2>
17
18          <?php
19  class Pegawai
20  {
21      public  $nama;
22      public $gaji;
23      public  function __construct($nama, $gaji)
24      {
25          $this->nama = $nama;
26          $this->gaji = $gaji;
27      }
28      public  function infoGaji()
29      {
30          return $this->gaji;
31      }
32  }
```

```php
33    class Manajer extends Pegawai
34    {
35            private $tunjangan;
36            public  function __construct($nama, $gaji, $tunjangan)
37            {
38                    parent::__construct($nama, $gaji);
39                    $this->tunjangan = $tunjangan;
40            }
41            public  function infoGaji()
42            {
43                    return $this->gaji;
44            }
45            public  function infoTunjangan()
46            {
47                    return $this->tunjangan;
48            }
49    }
50    class Programmer extends Pegawai
51    {
52            private $bonus;
53            public  function __construct($nama, $gaji, $bonus)
54            {
55                    parent::__construct($nama, $gaji);
56                    $this->bonus = $bonus;
57            }
58            public  function infoGaji()
59            {
60                    return $this->gaji;
61            }
62            public  function infoBonus()
63            {
64                    return $this->bonus;
65            }
66    }
```

```php
67    class Bayaran
68    {
69            public function hitungBayaran($peg)
70            {
71                    $uang = $peg->infoGaji();
72
73                    return $uang;
74            }
75            public static function main($args)
76            {
77                    echo "Sayyidina Auliya- 210913970034";
78            echo "<br>";
79                    $man = new Manajer("Gayatri", 6000000, 800000);
80            echo "<br>", "<br>";
81                    $prog = new Programmer("ica", 10000000, 900000);
82            echo "<br>", "<br>";
83                    $hr = new Bayaran();
84                    echo "<br> Gaji untuk Manajer Bernama ". $man->nama." : Rp. ".strval($hr->hitungBayaran($man)) , "\n";
85            echo "<br>", "<br>";
86                    echo "<br>Gaji untuk Programmer Bernama ".$prog->nama. " : Rp. ".strval($hr->hitungBayaran($prog)) , "\n";
87
88            }
89    }
90    Bayaran::main(array());
91            ?>
92            <!-- END PHP  -->
93        </div>
94    </body>
95    </html>
```

Analisis :

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