

LAPORAN PRAKTIKUM 5
PEMROGRAMAN BERORIENTASI OBJEK



Disusun Oleh :

Ahmed Nur Sidik (21091397038)

PROGRAM STUDI D4 MANAJEMEN INFORMATIKA

FAKULTAS VOKASI

UNIVERSITAS NEGERI SURABAYA

2022

1. Suatu program terdiri dari class Pegawai sebagai parent class, class Manajer dan class Kurir sebagai subclass. Buatlah suatu program yang menerapkan konsep polymorphic argument (Polymorphic arguments adalah tipe suatu parameter yang menerima suatu nilai yang bertipe subclass-nya)

❖ PHP

```
Prak5 no1.php > PHP Intelephense > html > head > title
1 <!-- Ahmed Nur Sidik 21091397038 2021B-->
2
3 <!DOCTYPE html>
4 <html lang="id">
5 <head>
6     <meta charset="UTF-8">
7     <meta http-equiv="X-UA-Compatible" content="IE=edge">
8     <meta name="viewport" content="width=device-width, initial-scale=1.0">
9     <title>Tugas PBO Praktikum 5</title>
10 </head>
11 <body style="align-items: center;">
12     <div class="container">
13         <h2>No 1</h2>
14
15         <?php
16 class Pegawai
17 {
18     public $nama;
19     public function __construct($nama)
20     {
21         $this->nama = $nama;
22     }
23     public function getNama()
24     {
25         return $this->nama;
26     }
27 }
28 class Manajer extends Pegawai
29 {
30     public $tunjangan;
31     public function __construct($nama, $tunjangan)
32     {
33         parent::__construct($nama);
34         $this->tunjangan = $tunjangan;
35     }
36     public function getTunjangan()
37     {
38         return $this->tunjangan;
39     }
40 }
41 class Kurir extends Pegawai
42 {
43     public $gaji;
44
45     public function __construct($nama, $gaji)
46     {
47         parent::__construct($nama);
48         $this->gaji = $gaji;
49     }
50     public function getGaji()
51     {
52         return $this->gaji;
53     }
54 }
55
56 class Soal1
57 {
58     public static
59     function Proses($peg)
60     {
61         if ($peg instanceof Manajer)
62         {
63             $man = $peg;
64             echo "<br>Nama manajer: ".$man->nama, "\n";
65             echo "<br>Tunjangan: Rp. ".strval($man->tunjangan), "\n";
66         }
67         else if ($peg instanceof Kurir)
68         {
69             $kur = $peg;
70             echo "<br>Nama kurir: ".$kur->nama, "\n";
71             echo "<br>Gaji: Rp. ".strval($kur->gaji), "\n";
72         }
73     }
74
75     public static
76     function main($args)
77     {
78         echo "Ahmed Nur Sidik - 21091397038", "\n";
79         echo "<br>", "<br>";
80         $peg1 = new Manajer("Gabriel", 23000000);
81         Soal1::Proses($peg1);
82         echo "<br>", "<br>";
83         $peg2 = new Kurir("Angel", 14000000);
84         Soal1::Proses($peg2);
85     }
86 }
87 Soal1::main(array());
88 ?>
89
90 </div>
91 </body>
92 </html>
```

❖ Output

No 1

Ahmed Nur Sidik - 21091397038

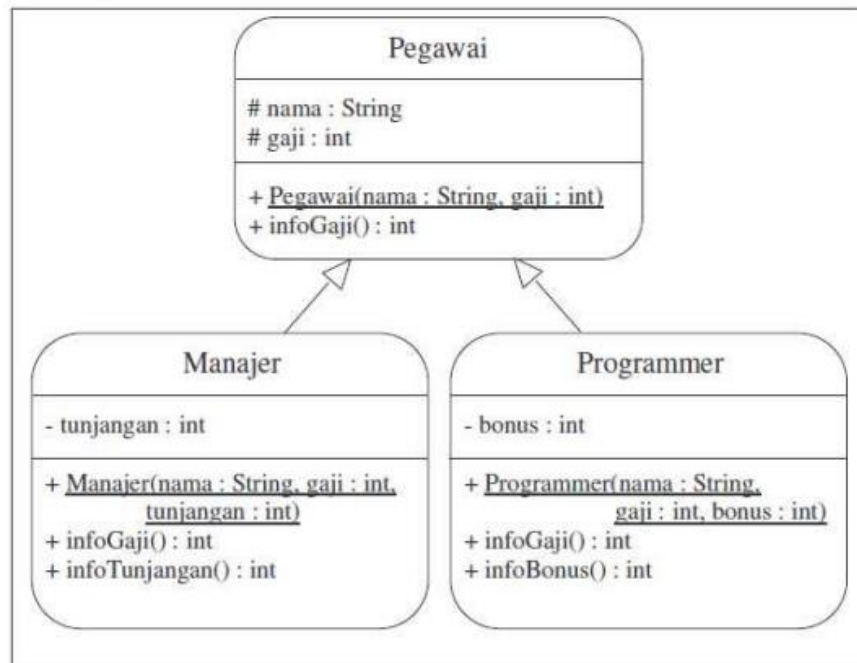
Nama manajer: Gabriel
Tunjangan: Rp. 23000000

Nama kurir: Angel
Gaji: Rp. 14000000

❖ Analisa

implementasi dari polymorphic argument berada di class info dimana method info dibuat statik agar pemanggilannya tidak memerlukan diinisiasi, sehingga bisa langsung dimasukkan menjadi parameter pada method info di class info.

2. Buat program berdasarkan UML berikut



❖ PHP

```
Prak5 no2.php > PHP Intelephense > html > body > div.container > Bayaran >
1 <!-- Ahmed Nur Sidik 21091397038 20218-->
2
3 <!DOCTYPE html>
4 <html lang="id">
5 <head>
6 <meta charset="UTF-8">
7 <meta http-equiv="X-UA-Compatible" content="IE=edge">
8 <meta name="viewport" content="width=device-width, initial-scale=1.0">
9 <title>Tugas PBO Praktikum 5</title>
10 </head>
11 <body>
12 <div class="container">
13 <h2>No 2</h2>
14
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 "Ahmed Nur Sidik - 21091397038";
75         echo "<br>";
76         $man = new Manajer("Yui", 5000000, 700000);
77
78         echo "<br>", "<br>";
79         $prog = new Programmer("Eren", 3000000, 200000);
80         echo "<br>", "<br>";
81         $hr = new Bayaran();
82         echo "<br> Gaji untuk Manajer Bernama ". $man->nama." : Rp. ".strval($hr->hitungBayaran($man)) , "<br>";
83         echo "<br>", "<br>";
84         echo "<br>Gaji untuk Programmer Bernama ".$prog->nama." : Rp. ".strval($hr->hitungBayaran($prog)) , "<br>";
85     }
86 }
87 Bayaran::main(array());
88 <?>
89 <!-- END PHP -->
90 </div>
91 </body>
92 </html>
```

❖ Output

No 2

Ahmed Nur Sidik - 21091397038

Gaji untuk Manajer Bernama Yui : Rp. 5000000

Gaji untuk Programmer Bernama Eren : Rp. 3000000

❖ Analisa

Program tersebut merupakan penerapan inheritance dengan menggunakan konsep overriding, dimana pemanggilan construct di masing-masing class turunan hanya akan menginisiasi properti yang dimiliki dengan visibilitas private dan properti lain yang diturunkan akan langsung diinisiasi dengan construct dari parentnya