

Nama : FIFIT SYAFAATY

Nim : 21091397001

Kelas : 2021A

PRAKTIKUM 5

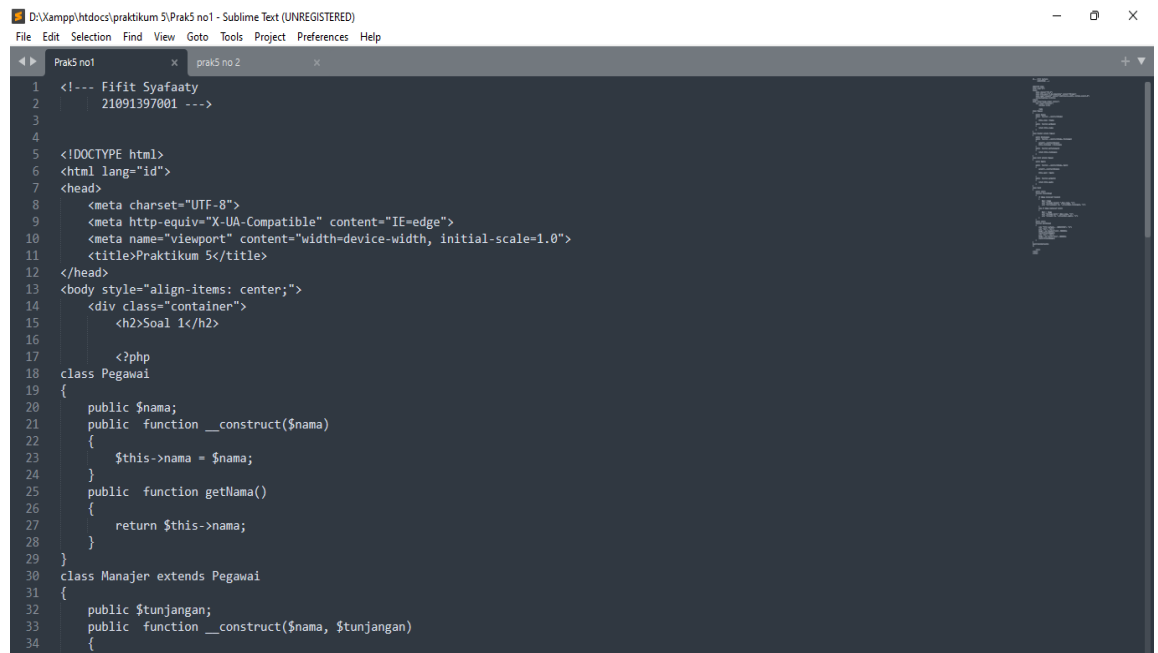
POLIMORFISME

TUGAS PRAKTIKUM

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)

Source Code

- Php



```
D:\Xampp\htdocs\praktikum 5\Prak5 no1 - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

Prak5 no1 x prak5 no2 x
1 <!-- Fifit Syaafaaty
2 21091397001 -->
3
4
5 <!DOCTYPE html>
6 <html lang="id">
7 <head>
8 <meta charset="UTF-8">
9 <meta http-equiv="X-UA-Compatible" content="IE=edge">
10 <meta name="viewport" content="width=device-width, initial-scale=1.0">
11 <title>Praktikum 5</title>
12 </head>
13 <body style="align-items: center;">
14 <div class="container">
15 <h2>Soal 1</h2>
16
17 <?php
18 class Pegawai
19 {
20     public $nama;
21     public function __construct($nama)
22     {
23         $this->nama = $nama;
24     }
25     public function getNama()
26     {
27         return $this->nama;
28     }
29 }
30 class Manajer extends Pegawai
31 {
32     public $tunjangan;
33     public function __construct($nama, $tunjangan)
34     {
35         parent::__construct($nama);
36     }
37 }
```

```
D:\Xampp\htdocs\praktikum 5\Prak5 no1 - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

Prak5 no1 x prak5 no2 x

34 {
35     parent::__construct($nama);
36     $this->tunjangan = $tunjangan;
37 }
38 public function getTunjangan()
39 {
40     return $this->tunjangan;
41 }
42 }
43 class Kurir extends Pegawai
44 {
45     public $gaji;
46
47     public function __construct($nama, $gaji)
48     {
49         parent::__construct($nama);
50
51         $this->gaji = $gaji;
52     }
53     public function getGaji()
54     {
55         return $this->gaji;
56     }
57 }
58 }
59 class Soal1
60 {
61     public static
62     function Proses($peg)
63     {
64         if ($peg instanceof Manajer)
65         {
66             $man = $peg;
67             echo "<br>Nama manajer: ".$man->nama, "\n";
```

```
D:\Xampp\htdocs\praktikum 5\Prak5 no1 - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

Prak5 no1 x prak5 no2 x

67         echo "<br>Nama manajer: ".$man->nama, "\n";
68         echo "<br>Tunjangan: Rp. ".strval($man->tunjangan), "\n";
69     }
70     else if ($peg instanceof Kurir)
71     {
72         $kur = $peg;
73         echo "<br>Nama kurir: ".$kur->nama, "\n";
74         echo "<br>Gaji: Rp. ".strval($kur->gaji), "\n";
75     }
76 }
77 public static
78 function main($args)
79 {
80     echo "fifit syafaaty. - 20051397001", "\n";
81     echo "<br>", "<br>";
82     $peg1 = new Manajer("ocha", 5000000);
83     Soal1::Proses($peg1);
84     echo "<br>", "<br>";
85     $peg2 = new Kurir("farel", 2000000);
86     Soal1::Proses($peg2);
87 }
88 }
89 Soal1::main(array());
90 ?>
91
92 </div>
93 </body>
94 </html>
```

• Output

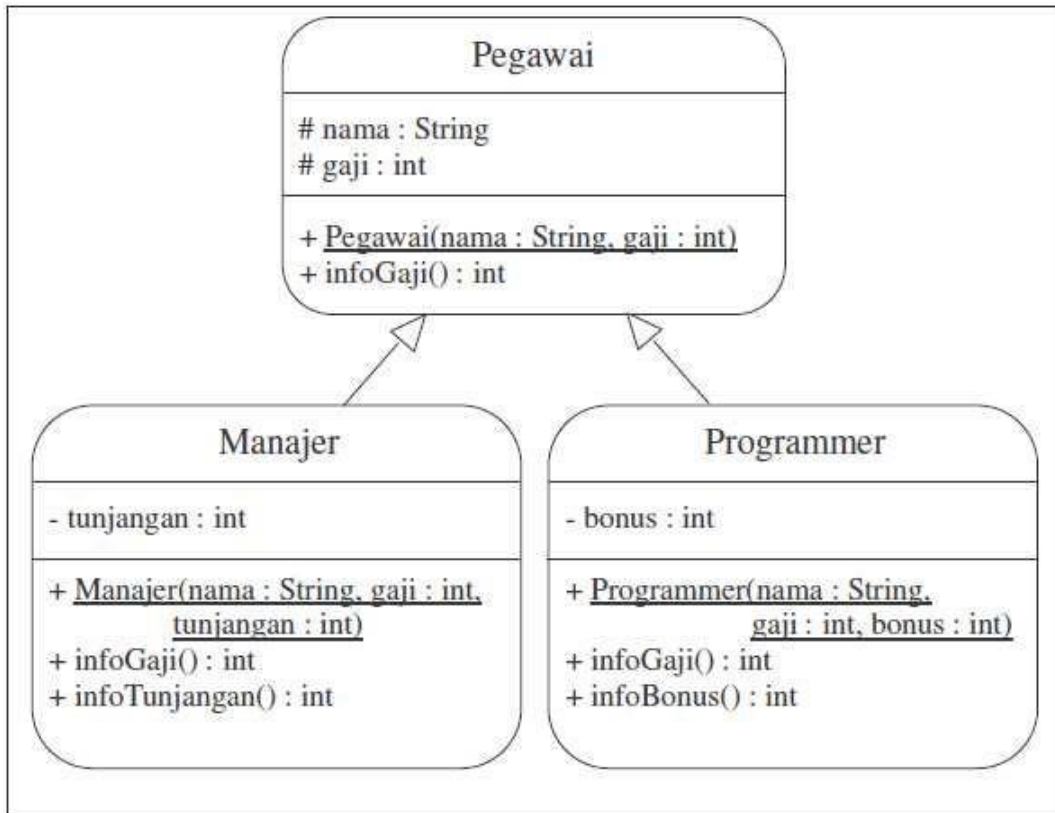
Soal 1

```
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; } } 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 = $peg; echo "
Nama manajer: ".$man->nama, "\n"; echo "
Tunjangan: Rp. ".strval($man->tunjangan), "\n"; } else if ($peg instanceof Kurir) { $kur = $peg; echo "
Nama kurir: ".$kur->nama, "\n"; echo "
Gaji: Rp. ".strval($kur->gaji), "\n"; } } public static function main($args) { echo "fifit syafaaty. - 20051397001", "\n"; echo "
";
"; $peg1 = new Manajer("ocha", 5000000); Soal1::Proses($peg1); echo "
";
"; $peg2 = new Kurir("farel", 2000000); Soal1::Proses($peg2); } } Soal1::main(array()); ?>
```

- **Analisa**

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

2. Buat program berdasarkan UML berikut



Source Code

- **Php**

```
D:\Xampp\htdocs\praktikum 5\prak5 no 2 - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

Prak5 no1 x prak5 no2 x

1 <!-- Fifi Syafaaty
2 21091397001 -->
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>
13     <div class="container">
14         <h2>Soal 2</h2>
15     </div>
16     <?php
17     class Pegawai
18     {
19         public $nama;
20         public $gaji;
21         public function __construct($nama, $gaji)
22         {
23             $this->nama = $nama;
24             $this->gaji = $gaji;
25         }
26         public function infoGaji()
27         {
28             return $this->gaji;
29         }
30     }
31     class Manajer extends Pegawai
32     {
33         private $tunjangan;
34         public function __construct($nama, $gaji, $tunjangan)
35         {
36             parent::__construct($nama, $gaji);
37             $this->tunjangan = $tunjangan;
38         }
39         public function infoGaji()
40         {
41             return $this->gaji;
42         }
43         public function infoTunjangan()
44         {
45             return $this->tunjangan;
46         }
47     }
48     class Programmer extends Pegawai
49     {
50         private $bonus;
51         public function __construct($nama, $gaji, $bonus)
52         {
53             parent::__construct($nama, $gaji);
54             $this->bonus = $bonus;
55         }
56         public function infoGaji()
57         {
58             return $this->gaji;
59         }
60         public function infoBonus()
61         {
62             return $this->bonus;
63         }
64     }
65     class Bayaran
66     {
67         public function hitungBayaran($peg)
```

```
D:\xampp\htdocs\praktikum 5\prak5 no 2 - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

Prak5 no1 x prak5 no2 x
67 public function hitungBayaran($peg)
68 {
69     $uang = $peg->infoGaji();
70
71     return $uang;
72 }
73 public static function main($args)
74 {
75     echo "fifit syafaaty. - 20051397001";
76     echo "<br>";
77     $man = new Manajer("linda", 5000000, 500000);
78     echo "<br>", "<br>";
79     $prog = new Programmer("farel", 4000000, 300000);
80     echo "<br>", "<br>";
81     $hr = new Bayaran();
82     echo "<br> Gaji untuk Manajer Bernama ". $man->nama." : Rp. ".strval($hr->hitungBayaran($man)), "\n";
83     echo "<br>", "<br>";
84     echo "<br>Gaji untuk Programmer Bernama ". $prog->nama." : Rp. ".strval($hr->hitungBayaran($prog)), "\n";
85
86 }
87 }
88 Bayaran::main(array());
89 ?>
90 <!-- END PHP -->
91 </div>
92 </body>
93 </html>
```

• Output

Soal 2

```
nama = $nama; $this->gaji = $gaji; } public function infoGaji() { return $this->gaji; } } class Manajer extends Pegawai { private $tunjangan; public function __construct($nama, $gaji, $tunjangan) {
parent::__construct($nama, $gaji); $this->tunjangan = $tunjangan; } public function infoGaji() { return $this->gaji; } public function infoTunjangan() { return $this->tunjangan; } } class Programmer extends
Pegawai { private $bonus; public function __construct($nama, $gaji, $bonus) { parent::__construct($nama, $gaji); $this->bonus = $bonus; } public function infoGaji() { return $this->gaji; } public function
infoBonus() { return $this->bonus; } } class Bayaran { public function hitungBayaran($peg) { $uang = $peg->infoGaji(); return $uang; } public static function main($args) { echo "fifit syafaaty. -
20051397001"; echo "
"; $man = new Manajer("linda", 5000000, 500000); echo "
"
"
"; $prog = new Programmer("farel", 4000000, 300000); echo "
"
"
"; $hr = new Bayaran(); echo "
"; echo "
Gaji untuk Manajer Bernama ". $man->nama." : Rp. ".strval($hr->hitungBayaran($man)), "\n"; echo "
"
"; echo "
Gaji untuk Programmer Bernama ". $prog->nama." : Rp. ".strval($hr->hitungBayaran($prog)), "\n"; } } Bayaran::main(array()); ?>
```

• Analisa

Program diatas adalah penerapan inheritance dengan 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.