

LAPORAN PRAKTIKUM 5
PEMPROGRAMAN BERORIENTASI OBJEK



Dibuat oleh :
TEGUH ALDIANTO
21091397076

FAKULTAS VOKASI
D4 MANAJEMEN INFORMATIKA
UNIVERSITAS NEGERI SURABAYA
2022

PRAKTIKUM 5

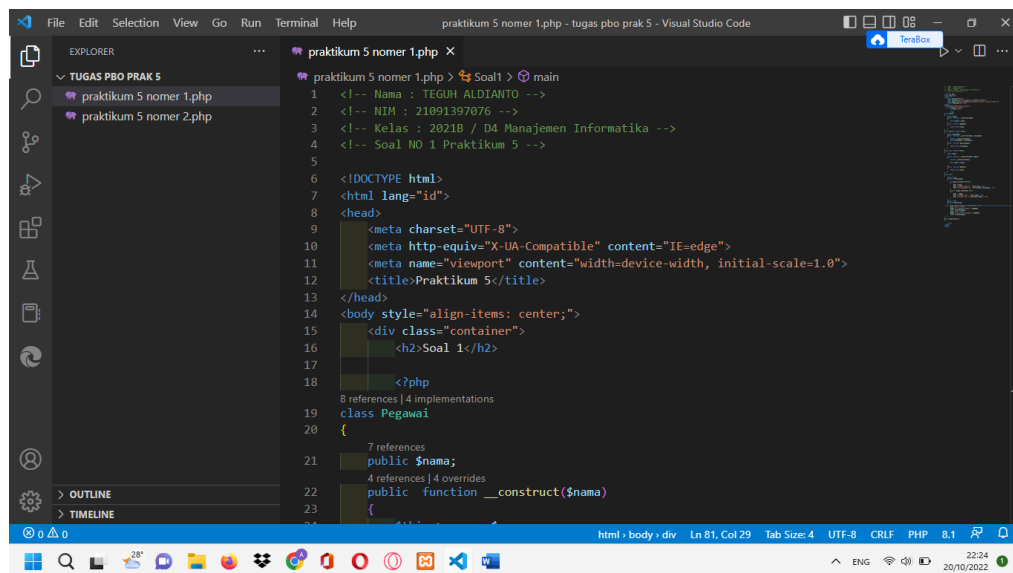
POLIMORFISME

TUGAS PRAKTIKUM

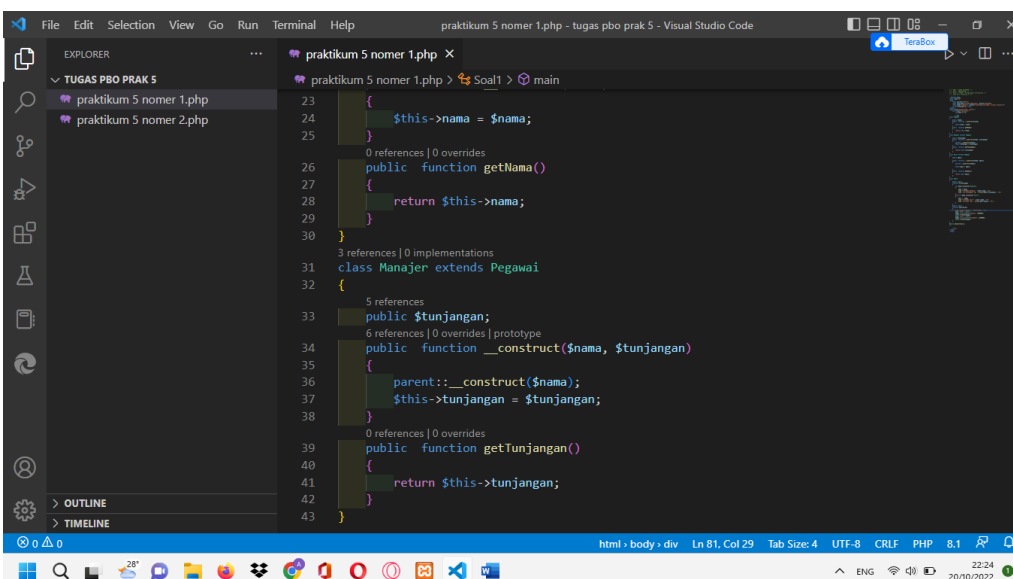
1. Suatu program terdiri dari class Pegawai sebagai parent class, class Manajer dan class Kurir sebagai subclass. Buatlah suatu program yang menerapkan konsep polymorphic argument (Polymorphic arguments adalah tipe suatu parameter yang menerima suatu nilai yang bertipe subclass-nya)

Penyelesaian :

- Codingan



```
praktikum 5 nomor 1.php X
praktikum 5 nomor 1.php > Soal1 > main
1 <!-- Nama : TEGUH ALDIANTO -->
2 <!-- NIM : 21091397076 -->
3 <!-- Kelas : 2021B / D4 Manajemen Informatika -->
4 <!-- Soal NO 1 Praktikum 5 -->
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 style="align-items: center;">
15     <div class="container">
16         <h2>Soal 1</h2>
17     </div>
18     <?php
19         8 references | 4 implementations
20         class Pegawai
21         {
22             7 references
23             public $nama;
24             4 references | 4 overrides
25             public function __construct($nama)
26             {
```



```
23     {
24         $this->nama = $nama;
25     }
26     0 references | 0 overrides
27     public function getNama()
28     {
29         return $this->nama;
30     }
31     3 references | 0 implementations
32     class Manajer extends Pegawai
33     {
34         5 references
35         public $tunjangan;
36         6 references | 0 overrides | prototype
37         public function __construct($nama, $tunjangan)
38         {
39             parent::__construct($nama);
40             $this->tunjangan = $tunjangan;
41         }
42         0 references | 0 overrides
43         public function getTunjangan()
44         {
45             return $this->tunjangan;
46         }
47     }
```

This screenshot shows the Visual Studio Code editor with the file 'praktikum 5 nomor 1.php' open. The Explorer sidebar on the left shows a folder named 'TUGAS PBO PRAK 5' containing two files: 'praktikum 5 nomor 1.php' and 'praktikum 5 nomor 2.php'. The main editor area displays the PHP code for the 'Kurir' class, which extends the 'Pegawai' class. The 'Kurir' class has a public property '\$gaji' and a public method '__construct(\$nama, \$gaji)' that calls the parent's constructor. It also has a public method 'getGaji()' that returns the '\$gaji' property. Below the 'Kurir' class is the 'Soal1' class, which has a public static method 'Proses(\$peg)'. The status bar at the bottom indicates the file is 'html > body > div' at line 81, column 29, with a tab size of 4, UTF-8 encoding, CRLF line endings, and PHP 8.1 version.

```
praktikum 5 nomor 1.php > Soal1 > main
2 references | 0 implementations
class Kurir extends Pegawai
{
    7 references
    public $gaji;
    5 references | 0 overrides | prototype
    public function __construct($nama, $gaji)
    {
        parent::__construct($nama);
        $this->gaji = $gaji;
    }
    0 references | 0 overrides
    public function getGaji()
    {
        return $this->gaji;
    }
    3 references | 0 implementations
class Soal1
{
    2 references | 0 overrides
    public static
    function Proses($peg)
    {
```

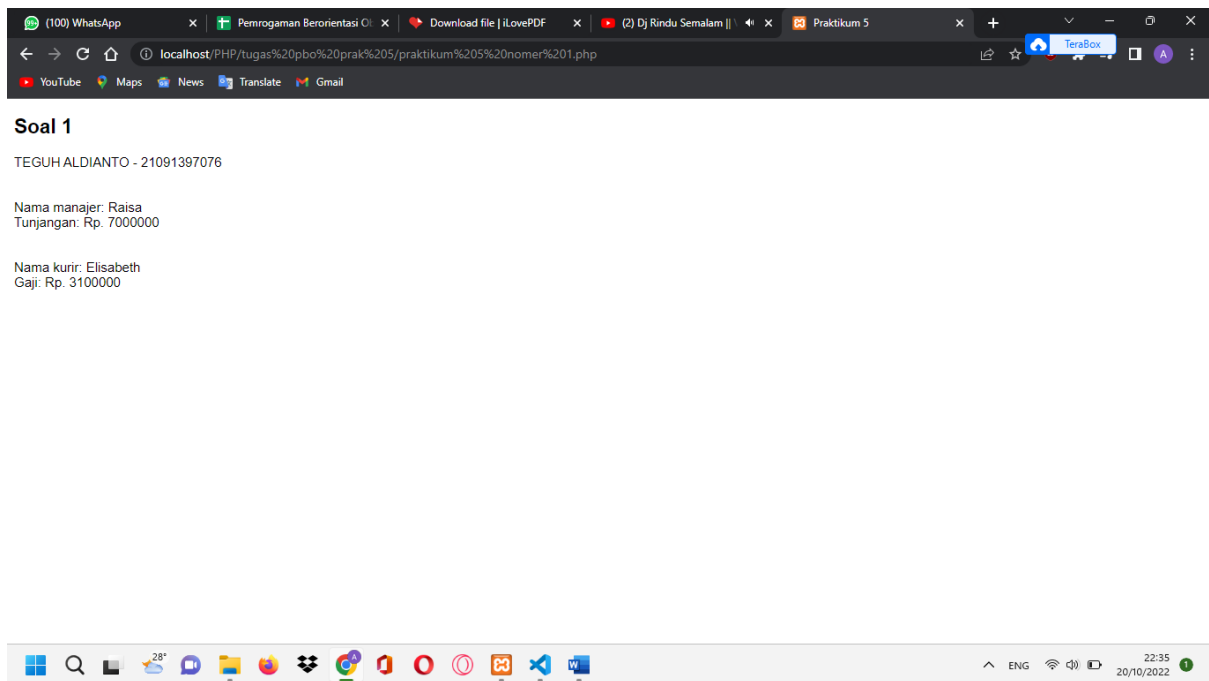
This screenshot shows the Visual Studio Code editor with the file 'praktikum 5 nomor 1.php' open. The Explorer sidebar on the left shows the same folder structure as the previous screenshot. The main editor area displays the PHP code for the 'main' function. It starts with an 'if' statement checking if '\$peg' is an instance of 'Manajer'. If true, it assigns '\$man = \$peg' and echoes the name and salary. Then, it checks if '\$peg' is an instance of 'Kurir'. If true, it assigns '\$kur = \$peg' and echoes the name and salary. Below the 'if' statements, there is a 'public static function main(\$args)' method. Inside this function, it echoes a message, creates a new 'Manajer' object '\$peg1' with name 'Raisa' and salary 7000000, and a new 'Kurir' object '\$peg2' with name 'Elisabeth' and salary 3100000. It then calls 'Soal1::Proses(\$peg1)' and 'Soal1::Proses(\$peg2)'. The status bar at the bottom indicates the file is 'html > body > div' at line 81, column 29, with a tab size of 4, UTF-8 encoding, CRLF line endings, and PHP 8.1 version.

```
praktikum 5 nomor 1.php > Soal1 > main
65 if ($peg instanceof Manajer)
66 {
67     $man = $peg;
68     echo "<br>Nama manajer: ".$man->nama, "\n";
69     echo "<br>Tunjangan: Rp. ".strval($man->tunjangan), "\n";
70 }
71 else if ($peg instanceof Kurir)
72 {
73     $kur = $peg;
74     echo "<br>Nama kurir: ".$kur->nama, "\n";
75     echo "<br>Gaji: Rp. ".strval($kur->gaji), "\n";
76 }
77 }
78 1 reference | 0 overrides
79 public static
80 function main($args)
81 {
82     echo "TEGUH ALDIANTO - 21091397076", "\n";
83     echo "<br>", "<br>";
84     $peg1 = new Manajer("Raisa", 7000000);
85     Soal1::Proses($peg1);
86     echo "<br>", "<br>";
87     $peg2 = new Kurir("Elisabeth", 3100000);
88     Soal1::Proses($peg2);
89 }
```

This screenshot shows the Visual Studio Code editor with the file 'praktikum 5 nomor 1.php' open. The Explorer sidebar on the left shows the same folder structure as the previous screenshots. The main editor area displays the PHP code for the 'main' function. It starts with an 'if' statement checking if '\$peg' is an instance of 'Manajer'. If true, it assigns '\$man = \$peg' and echoes the name and salary. Then, it checks if '\$peg' is an instance of 'Kurir'. If true, it assigns '\$kur = \$peg' and echoes the name and salary. Below the 'if' statements, there is a 'public static function main(\$args)' method. Inside this function, it echoes a message, creates a new 'Manajer' object '\$peg1' with name 'Raisa' and salary 7000000, and a new 'Kurir' object '\$peg2' with name 'Elisabeth' and salary 3100000. It then calls 'Soal1::Proses(\$peg1)' and 'Soal1::Proses(\$peg2)'. The status bar at the bottom indicates the file is 'html > body > div' at line 81, column 29, with a tab size of 4, UTF-8 encoding, CRLF line endings, and PHP 8.1 version.

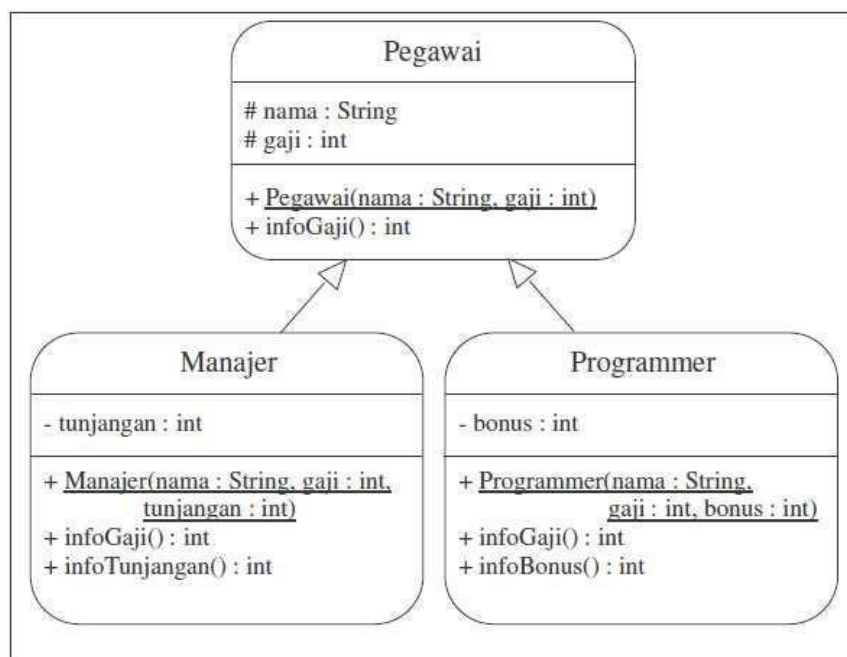
```
praktikum 5 nomor 1.php > Soal1 > main
71 else if ($peg instanceof Kurir)
72 {
73     $kur = $peg;
74     echo "<br>Nama kurir: ".$kur->nama, "\n";
75     echo "<br>Gaji: Rp. ".strval($kur->gaji), "\n";
76 }
77 }
78 1 reference | 0 overrides
79 public static
80 function main($args)
81 {
82     echo "TEGUH ALDIANTO - 21091397076", "\n";
83     echo "<br>", "<br>";
84     $peg1 = new Manajer("Raisa", 7000000);
85     Soal1::Proses($peg1);
86     echo "<br>", "<br>";
87     $peg2 = new Kurir("Elisabeth", 3100000);
88     Soal1::Proses($peg2);
89 }
90 Soal1::main(array());
91 ?>
92 </div>
93 </body>
94 </html>
```

- Output



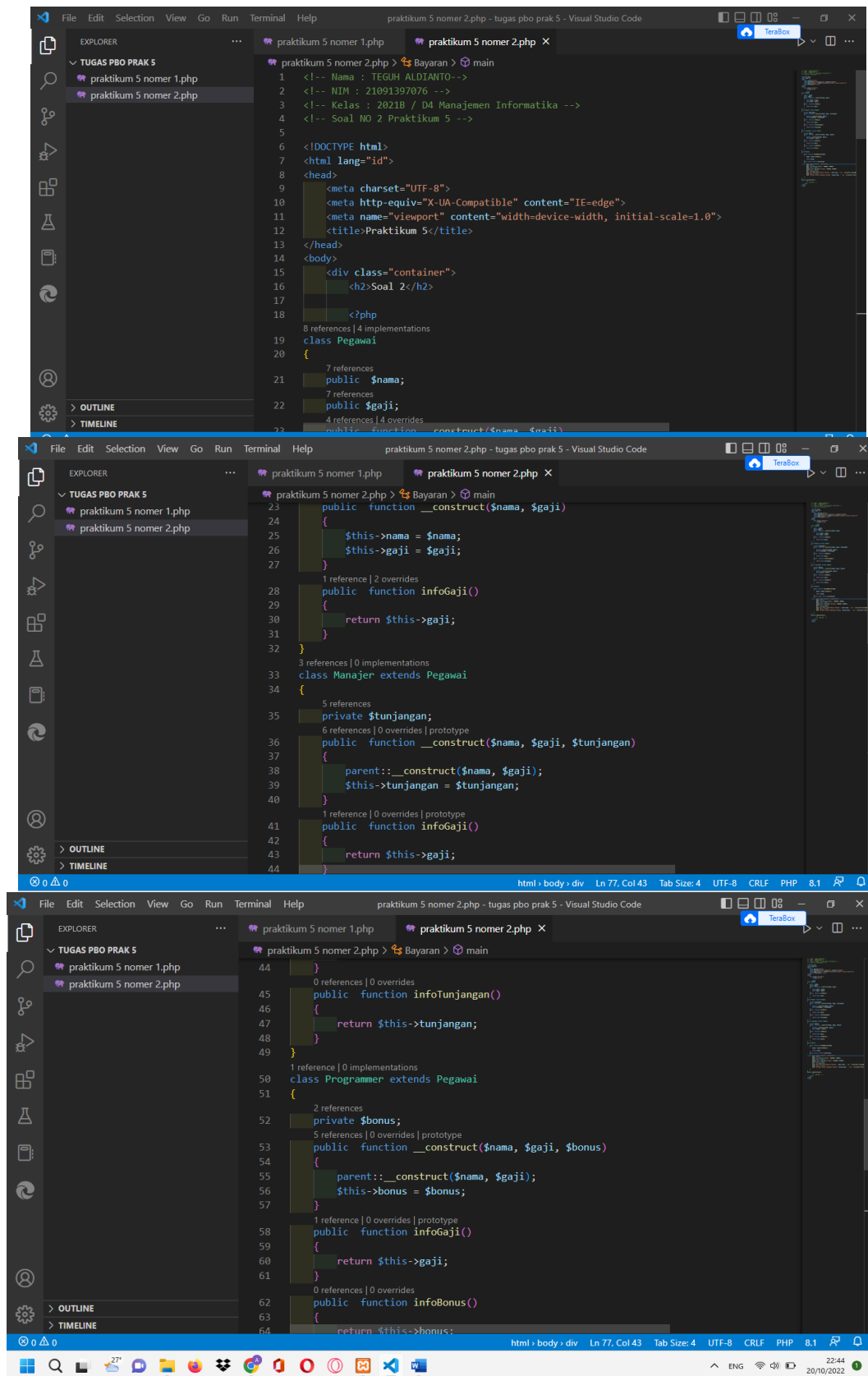
- Analisa : Pada kasus yang terdapat pada soal diatas implementasi dari polymorphic argument berada di class info dimana method info dibuat static agar pemanggilannya tidak memerlukan diinisiasi, sehingga bisa langsung dimasukkan menjadi parameter pada method info di class info.

2. buat program berdasarkan UML berikut :



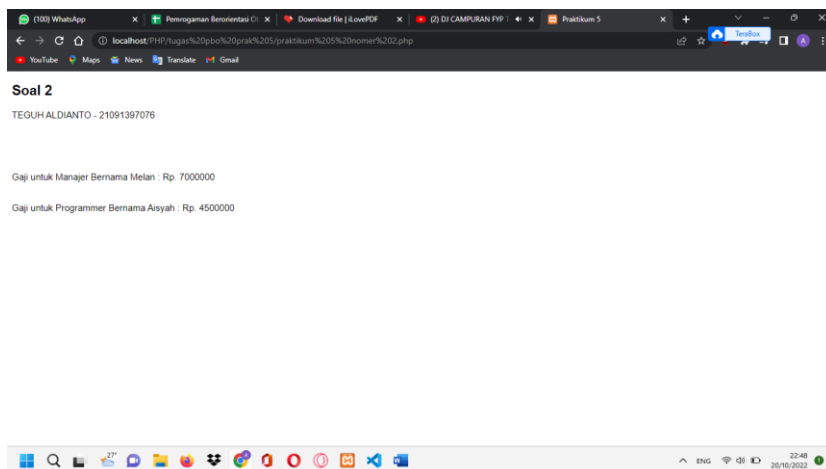
Penyelesaian :

- Codingan



```
praktikum 5 nomer 2.php - tugas pbo prak 5 - Visual Studio Code
praktikum 5 nomer 1.php
praktikum 5 nomer 2.php
TUGAS PBO PRAK 5
praktikum 5 nomer 1.php
praktikum 5 nomer 2.php
praktikum 5 nomer 2.php > Bayaran > main
64     return $this->bonus;
65 }
66 }
67 2 references | 0 implementations
68 class Bayaran
69 {
70     2 references | 0 overrides
71     public function hitungBayaran($peg)
72     {
73         $uang = $peg->infoGaji();
74         return $uang;
75     }
76     1 reference | 0 overrides
77     public static function main($args)
78     {
79         echo "TEGUH ALDIANTO - 21091397076";
80         echo "<br>";
81         $man = new Manajer("Melan", 7000000, 700000);
82         echo "<br>", "<br>";
83         $prog = new Programmer("Aisyah", 4500000, 550000);
84         echo "<br>", "<br>";
85         $hr = new Bayaran();
86         echo "<br> Gaji untuk Manajer Bernama ". $man->nama. " : Rp. ".strval($hr->hitungBayaran($man));
87         echo "<br>";
88         echo "<br>Gaji untuk Programmer Bernama ".$prog->nama. " : Rp. ".strval($hr->hitungBayaran($prog));
89     }
90 }
91 Bayaran::main(array());
92 }
93 <!-- END PHP -->
94 </div>
95 </body>
96 </html>
```

Output :



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.