LAPORAN PRAKTIKUM 5

Mata Kuliah Pemrograman Berorientasi Obyek

Oleh:

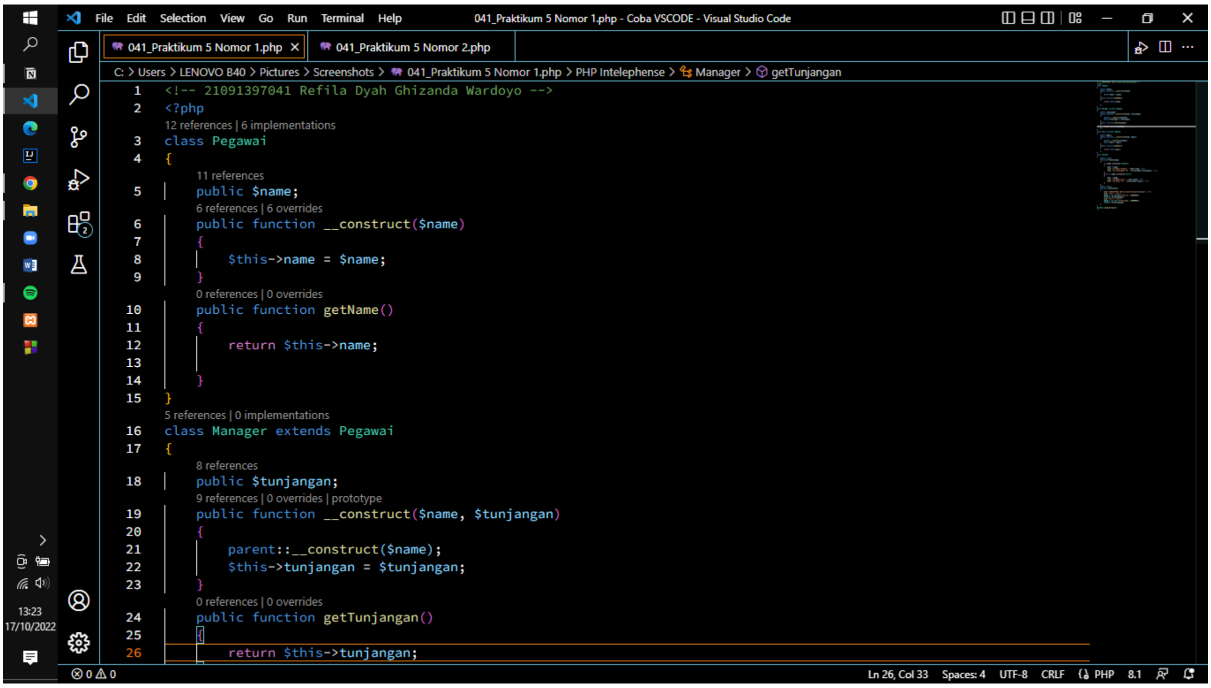
Refila Dyah Ghizanda Wardoyo
21091397041

PROGRAM STUDI D4 MANAJEMEN INFORMATIKA

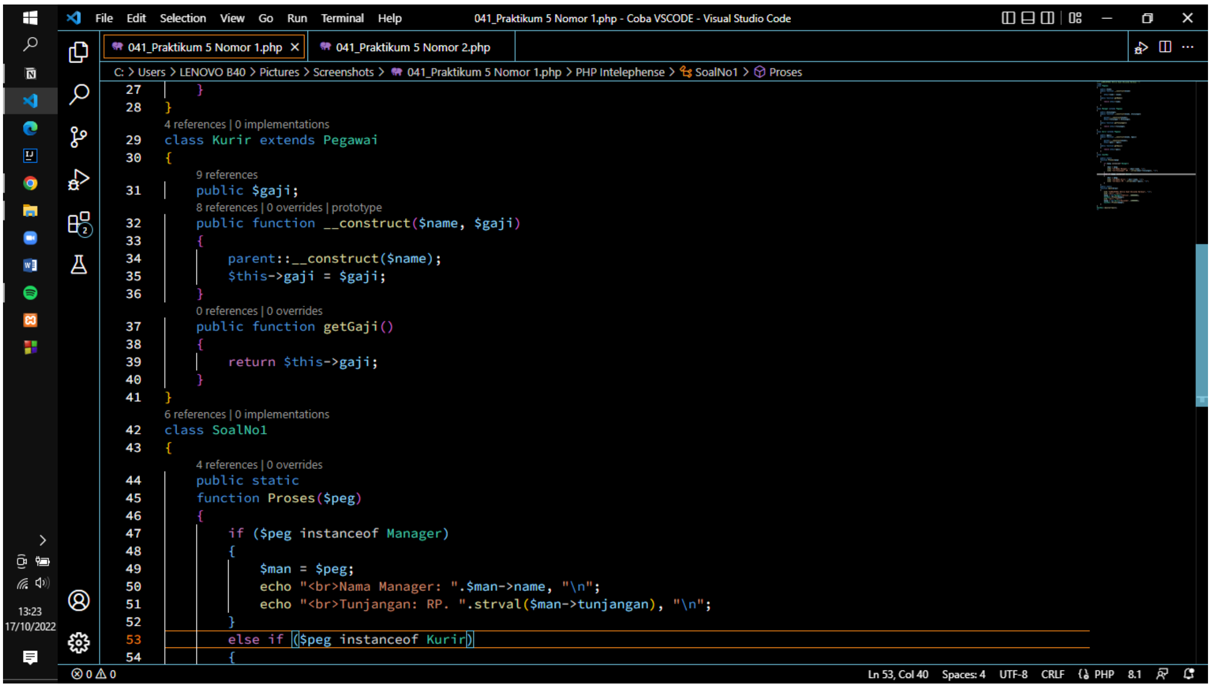
FAKULTAS VOKASI

UNIVERSITAS NEGERI SURABAYA

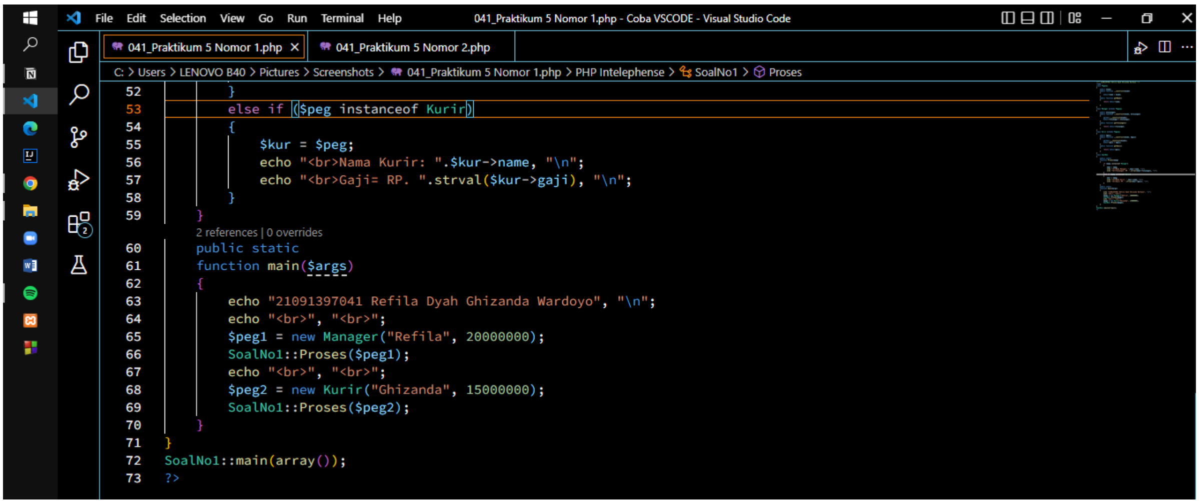
1. Source Code: PHP


File Edit Selection View Go Run Terminal Help
041_Praktikum 5 Nomor 1.php - Coba VSCODE - Visual Studio Code
041_Praktikum 5 Nomor 1.php
041_Praktikum 5 Nomor 2.php
C: > Users > LENOVO B40 > Pictures > Screenshots > 041_Praktikum 5 Nomor 1.php > PHP Intelephense > Manager > getTunjangan
Ln 26, Col 33 Spaces: 4 UTF-8 CRLF PHP 8.1
13:23
17/10/2022

```php
<!-- 21091397041 Refila Dyah Ghizanda Wardoyo -->
<?php
12 references | 6 implementations
class Pegawai
{
    11 references
    public $name;
    6 references | 6 overrides
    public function __construct($name)
    {
        $this->name = $name;
    }
    0 references | 0 overrides
    public function getName()
    {
        return $this->name;

    }
}
5 references | 0 implementations
class Manager extends Pegawai
{
    8 references
    public $tunjangan;
    9 references | 0 overrides | prototype
    public function __construct($name, $tunjangan)
    {
        parent::__construct($name);
        $this->tunjangan = $tunjangan;
    }
    0 references | 0 overrides
    public function getTunjangan()
    {
        return $this->tunjangan;
```


File Edit Selection View Go Run Terminal Help
041_Praktikum 5 Nomor 1.php - Coba VSCODE - Visual Studio Code
041_Praktikum 5 Nomor 1.php
041_Praktikum 5 Nomor 2.php
C: > Users > LENOVO B40 > Pictures > Screenshots > 041_Praktikum 5 Nomor 1.php > PHP Intelephense > SoalNo1 > Proses
Ln 53, Col 40 Spaces: 4 UTF-8 CRLF PHP 8.1
13:23
17/10/2022

```php
    }
}
4 references | 0 implementations
class Kurir extends Pegawai
{
    9 references
    public $gaji;
    8 references | 0 overrides | prototype
    public function __construct($name, $gaji)
    {
        parent::__construct($name);
        $this->gaji = $gaji;
    }
    0 references | 0 overrides
    public function getGaji()
    {
        return $this->gaji;
    }
}
6 references | 0 implementations
class SoalNo1
{
    4 references | 0 overrides
    public static
    function Proses($peg)
    {
        if ($peg instanceof Manager)
        {
            $man = $peg;
            echo "<br>Nama Manager: ".$man->name, "\n";
            echo "<br>Tunjangan: RP. ".strval($man->tunjangan), "\n";
        }
        else if ($peg instanceof Kurir)
        {
```


File
Edit
Selection
View
Go
Run
Terminal
Help
041_Praktikum 5 Nomor 1.php - Coba VSCODE - Visual Studio Code
041_Praktikum 5 Nomor 1.php
041_Praktikum 5 Nomor 2.php
C: > Users > LENOVO B40 > Pictures > Screenshots > 041_Praktikum 5 Nomor 1.php > PHP Intelephense > SoalNo1 > Proses
2 references | 0 overrides

```php
        }
        else if ($peg instanceof Kurir)
        {
            $kur = $peg;
            echo "<br>Nama Kurir: ".$kur->name, "\n";
            echo "<br>Gaji= RP. ".strval($kur->gaji), "\n";
        }
    }
    public static
    function main($args)
    {
        echo "21091397041 Refila Dyah Ghizanda Wardoyo", "\n";
        echo "<br>", "<br>";
        $peg1 = new Manager("Refila", 20000000);
        SoalNo1::Proses($peg1);
        echo "<br>", "<br>";
        $peg2 = new Kurir("Ghizanda", 15000000);
        SoalNo1::Proses($peg2);
    }
}
SoalNo1::main(array());
?>
```

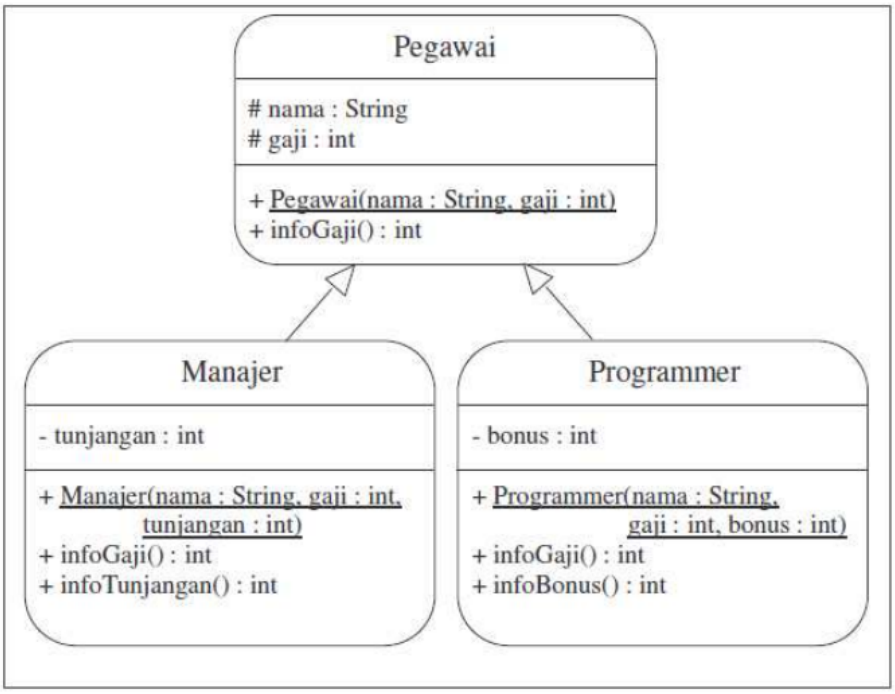
Output

```
21091397041 Refila Dyah Ghizanda Wardoyo


Nama Manager: Refila
Tunjangan: RP. 20000000


Nama Kurir: Ghizanda
Gaji= RP. 15000000
```

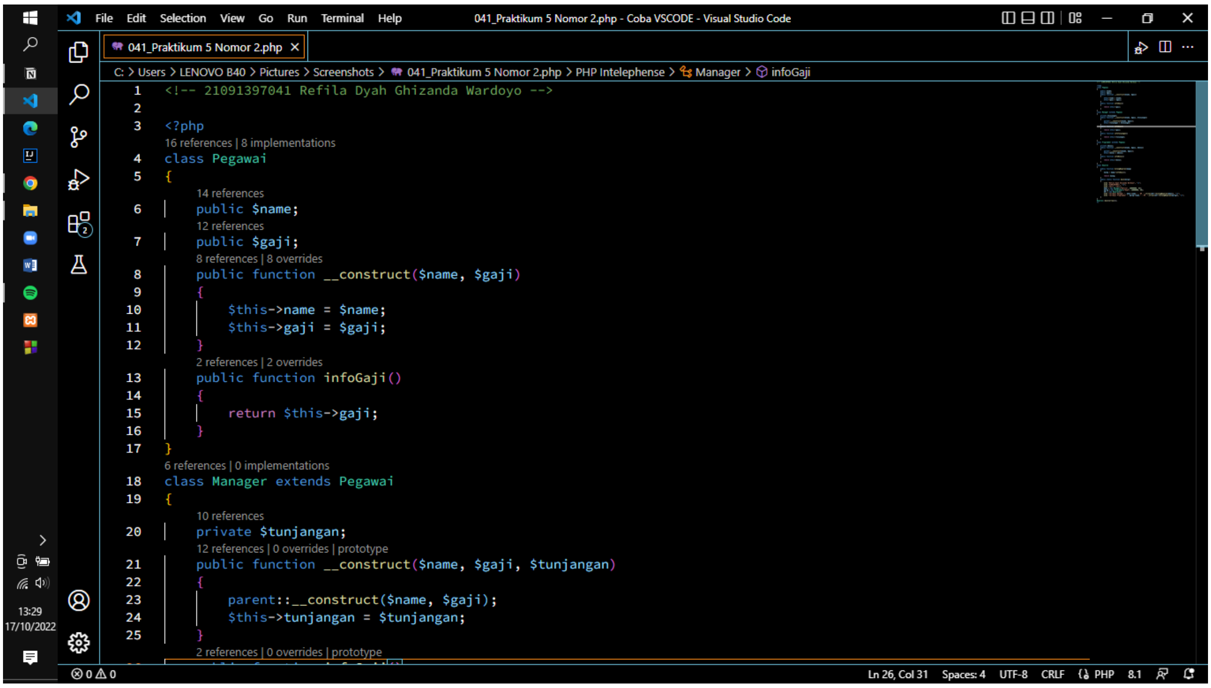
Analisa

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

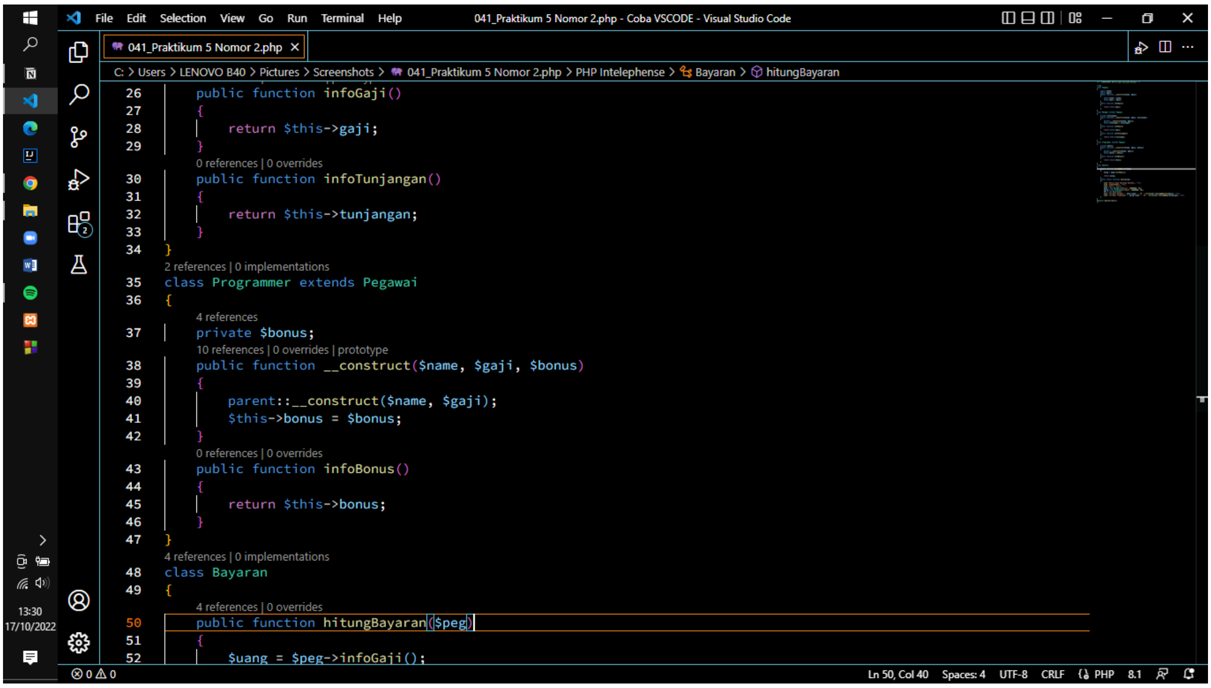
2. Buat program berdasarkan UML berikut



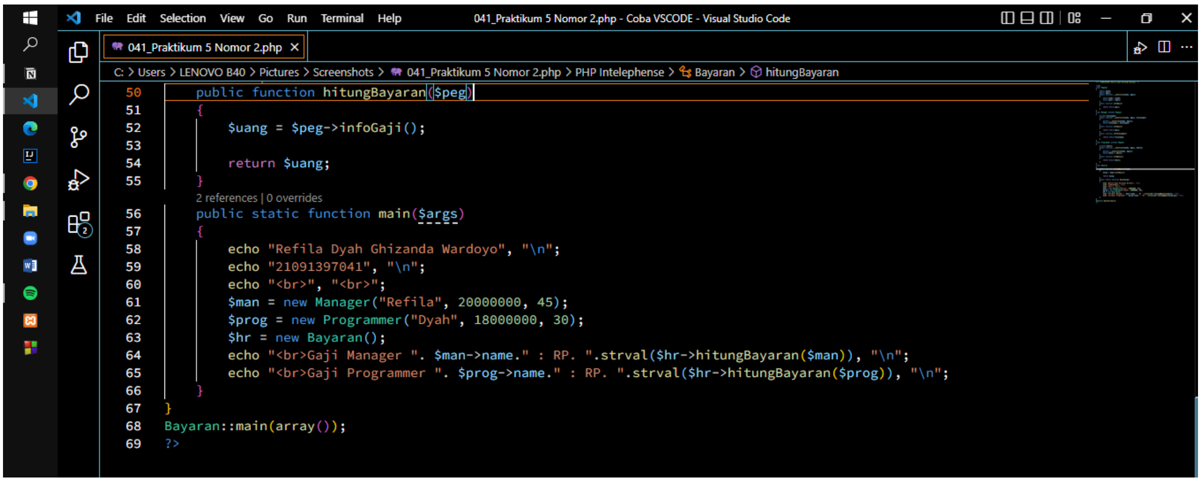
Source Code: PHP


File Edit Selection View Go Run Terminal Help
041_Praktikum 5 Nomor 2.php - Coba VSCODE - Visual Studio Code
041_Praktikum 5 Nomor 2.php
C: > Users > LENOVO B40 > Pictures > Screenshots > 041_Praktikum 5 Nomor 2.php > PHP Intelephense > Manager > infoGaji
16 references | 8 implementations
14 references
12 references
8 references | 8 overrides
2 references | 2 overrides
6 references | 0 implementations
10 references
12 references | 0 overrides | prototype
2 references | 0 overrides | prototype
13:29
17/10/2022
⊗ 0 ⚠ 0
Ln 26, Col 31 Spaces: 4 UTF-8 CRLF PHP 8.1

```php
<!-- 21091397041 Refila Dyah Ghizanda Wardoyo -->

<?php
class Pegawai
{
    public $name;
    public $gaji;
    public function __construct($name, $gaji)
    {
        $this->name = $name;
        $this->gaji = $gaji;
    }
    public function infoGaji()
    {
        return $this->gaji;
    }
}
class Manager extends Pegawai
{
    private $tunjangan;
    public function __construct($name, $gaji, $tunjangan)
    {
        parent::__construct($name, $gaji);
        $this->tunjangan = $tunjangan;
    }
```


File Edit Selection View Go Run Terminal Help
041_Praktikum 5 Nomor 2.php - Coba VSCODE - Visual Studio Code
041_Praktikum 5 Nomor 2.php
C: > Users > LENOVO B40 > Pictures > Screenshots > 041_Praktikum 5 Nomor 2.php > PHP Intelephense > Bayaran > hitungBayaran
13:30
17/10/2022
⊗ 0 ⚠ 0
Ln 50, Col 40 Spaces: 4 UTF-8 CRLF {} PHP 8.1

```
    public function infoGaji()
    {
        return $this->gaji;
    }
    public function infoTunjangan()
    {
        return $this->tunjangan;
    }
}
class Programmer extends Pegawai
{
    private $bonus;
    public function __construct($name, $gaji, $bonus)
    {
        parent::__construct($name, $gaji);
        $this->bonus = $bonus;
    }
    public function infoBonus()
    {
        return $this->bonus;
    }
}
class Bayaran
{
    public function hitungBayaran($peg)
    {
        $suang = $peg->infoGaji();
```


File Edit Selection View Go Run Terminal Help
041_Praktikum 5 Nomor 2.php - Coba VSCODE - Visual Studio Code
041_Praktikum 5 Nomor 2.php
C: > Users > LENOVO B40 > Pictures > Screenshots > 041_Praktikum 5 Nomor 2.php > PHP Intelephense > Bayaran > hitungBayaran

```
    public function hitungBayaran($peg)
    {
        $suang = $peg->infoGaji();

        return $suang;
    }
    public static function main($args)
    {
        echo "Refila Dyah Ghizanda Wardoyo", "\n";
        echo "21091397041", "\n";
        echo "<br>", "<br>";
        $man = new Manager("Refila", 20000000, 45);
        $prog = new Programmer("Dyah", 18000000, 30);
        $hr = new Bayaran();
        echo "<br>Gaji Manager ". $man->name." : RP. ".strval($hr->hitungBayaran($man)), "\n";
        echo "<br>Gaji Programmer ". $prog->name." : RP. ".strval($hr->hitungBayaran($prog)), "\n";
    }
}
Bayaran::main(array());
?>
```

Output

```
Refila Dyah Ghizanda Wardoyo 21091397041


Gaji Manager Refila : RP. 20000000
Gaji Programmer Dyah : RP. 18000000
```

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.