

# **LAPORAN ANALISIS PRAKTIKUM 3 INHERITANCE**

**MATA KULIAH : PEMROGRAMAN BERORIENTASI OBYEK**



**DISUSUN OLEH:**

**REVINA AURIGHA FIRDAUS**

**21091397003**

**PROGRAM STUDI D IV MANAJEMEN INFORMATIKA**

**FAKULTAS VOKASI**

**UNIVERSITAS NEGERI SURABAYA**

## 1. Rancangan program Class “Perbankan”

- Rancangan program (PHP)

```
terminal Help • tugasno1.php - Visual Studio Code
graphberbot.cpp 2 tugasno1.php tugasno2.a tugasno 2.a tugas no 2a.php tugasno2.b.php
C:\Users\Asuspro\Documents\Semester 3\Pemrograman Berorientasi Obyek\tugasno1.php

1 <?php
2
3 class Tabungan{
4     protected $saldo;
5 }
6
7 //Pengambilan Saldo
8 class PengambilanUang extends Tabungan{
9     private $proteksi;
10
11     public function __construct($saldo)
12     {
13         $this->saldo=$saldo;
14         $this->proteksi = 1000;
15     }
16
17     public function getSaldo(){
18         return "Uang yang ditabung : " . $this->saldo . "<br> Uang yang diproteksi : " . $this->proteksi;
19     }
20
21     public function ambilUang($jumlah){
22         if($jumlah == 4500){
23             return "Uang yang akan diambil : " . $jumlah . " false <br> Saldo sekarang : " . $this->saldo;
24         }else{
25             return "Uang yang akan diambil : " . $jumlah . " true <br> Saldo sekarang : " . $this->saldo - $jumlah;
26         }
27     }
28 }
29
30 $tabungan = new PengambilanUang(5000);
31 echo $tabungan->getSaldo();
32 echo "<br> ----- <br>";
33 echo $tabungan->ambilUang(4500);
34 echo "<br> ----- <br>";
35 echo $tabungan->ambilUang(2500);
```

- Output program

```
localhost/pbo/NOMOR%201.php

Uang yang ditabung : 5000
Uang yang diproteksi : 1000
-----
Uang yang akan diambil : 4500 false
Saldo sekarang : 5000
-----
Uang yang akan diambil : 2500 true
Saldo sekarang : 2500
```

- Analisis

Dalam kasus tersebut, saya memanfaatkan aturan overriding. Saldo yang diinisialisasi di awal akan dimasukkan ke method setSaldo yang ada di parent class. Uang proteksi disini bersifat sementara dan akan hilang ketika tidak digunakan lagi. Oleh karena itu, ketika ambilUang(2500) dijalankan, maka saldo sekarang adalah  $5000 - 2500 = 2500$  bukan 1500. Hal ini bisa terjadi karena method getSaldo() yang ada di subclass akan meng- override method getSaldo() yang ada di parent class.

2. a. Buatlah kelas-kelas berdasarkan UML class “Kendaraan”

```
terminal Help tugas no 2a.php - Visual Studio Code
C: graphberbot.cpp 2 tugasno1.php tugasno2a tugasno 2a tugas no 2a.php tugasno2.b.php
C: > Users > Asuspro > Documents > Semester 3 > Pemrograman Berorientasi Obyek > tugas no 2a.php

1 <?php
2
3 class Kendaraan {
4     protected $jmlRoda, $warna;
5     function setjmlRoda($jmlRoda) {
6         $this->jmlRoda = $jmlRoda;
7     }
8     public function getjmlRoda() {
9         return $this->jmlRoda;
10    }
11    public function setWarna($warna) {
12        $this->warna = $warna;
13    }
14    public function getWarna() {
15        return $this->warna;
16    }
17 }
18
19 class Mobil extends Kendaraan {
20     protected $bahanBakar, $kapasitasMesin;
21     public function setBahanBakar($bahanBakar) {
22         $this->bahanBakar = $bahanBakar;
23     }
24     public function getBahanBakar() {
25         return $this->bahanBakar;
26     }
27     public function setKapasitasMesin($kapasitasMesin) {
28         $this->kapasitasMesin = $kapasitasMesin;
29     }
30     public function getKapasitasMesin() {
31         return $this->kapasitasMesin;
32     }
33 }
34
35 class Sepeda extends Kendaraan {
36     protected $jmlSadel, $jmlGir;
37     public function setJmlSadel($jmlSadel) {
38         $this->jmlSadel = $jmlSadel;
39     }
40     public function getJmlSadel() {
41         return $this->jmlSadel;
42     }
43     public function setJmlGir($jmlGir) {
44         $this->jmlGir = $jmlGir;
45     }
46     public function getJmlGir() {
47         return $this->jmlGir;
48     }
49 }
50
51 class Truk extends Mobil {
52     private $muatanMaks;
53     public function setMuatanMaks($muatanMaks) {
54         $this->muatanMaks = $muatanMaks;
55     }
56     public function getMuatanMaks() {
57         return $this->muatanMaks;
58     }
59 }
60
61 class Taksi extends Mobil {
62     private $tarifAwal, $tarifperKm;
63     public function setTarifAwal($tarifAwal) {
64         $this->tarifAwal = $tarifAwal;
65     }
66     public function getTarifAwal() {
67         return $this->tarifAwal;
68     }
69     public function setTarifperKm($tarifperKm) {
70         $this->tarifperKm = $tarifperKm;
71     }
72     public function getTarifperKm() {
73         return $this->tarifperKm;
74     }
75 }
76
77 $truk1 = new Truk;
78 $truk1->setjmlRoda(4);
79 $truk1->setWarna('Kuning');
80 $truk1->setBahanBakar('Solar');
81 $truk1->setKapasitasMesin(1500);
82 $truk1->setMuatanMaks(1000);
83
84 $truk2 = new Truk;
85 $truk2->setjmlRoda(6);
86 $truk2->setWarna('Merah');
87 $truk2->setBahanBakar('Solar');
88 $truk2->setKapasitasMesin(2000);
89 $truk2->setMuatanMaks(5000);
```

```

terminal Help tugas no 2a.php - Visual Studio Code
graphberbobot.cpp 2 tugasno1.php tugasno2.a tugasno 2.a tugas no 2a.php X tugasno2.b.php
C:\Users> Asuspro > Documents > Semester 3 > Pemrograman Berorientasi Obyek > tugas no 2a.php
93 $taksi1->setWarna('Oranye');
94 $taksi1->setBahanBakar('Bensin');
95 $taksi1->setKapasitasMesin(1500);
96 $taksi1->setTarifAwal(10000);
97 $taksi1->setTarifPerKm(5000);
98
99 $taksi2 = new Taksi;
100 $taksi2->setJmlRoda(4);
101 $taksi2->setWarna('Biru');
102 $taksi2->setBahanBakar('Bensin');
103 $taksi2->setKapasitasMesin(1300);
104 $taksi2->setTarifAwal(7000);
105 $taksi2->setTarifPerKm(3500);
106
107 $sepeda1 = new Sepeda;
108 $sepeda1->setJmlRoda(3);
109 $sepeda1->setWarna('Hitam');
110 $sepeda1->setJmlSadel(1);
111 $sepeda1->setJmlGir(2);
112
113 $sepeda2 = new Sepeda;
114 $sepeda2->setJmlRoda(2);
115 $sepeda2->setWarna('Putih');
116 $sepeda2->setJmlSadel(2);
117 $sepeda2->setJmlGir(5);
118
119 >>

```

b. Selanjutnya buatlah kelas Tes.php yang membuat objek-objek serta mengeset nilai variabel seperti pada Tabel 1 dan tampilkan data per objek

```

terminal Help tugasno2.b.php - Visual Studio Code
graphberbobot.cpp 2 tugasno1.php tugasno2.a tugasno 2.a tugas no 2a.php tugasno2.b.php
C:\Users> Asuspro > Documents > Semester 3 > Pemrograman Berorientasi Obyek > tugasno2.b.php
1 <?php
2 | require_once 'Soal No 2 A.php';
3 >
4
5 <!DOCTYPE html>
6 <html lang="en">
7 <head>
8 | <title>Praktikum Modul 3</title>
9 </head>
10 <body>
11 <table>
12 | <tr>
13 | | <td>No</td>
14 | | <td>Object</td>
15 | | <td>Jumlah Roda</td>
16 | | <td>Warna</td>
17 | | <td>Bahan Bakar</td>
18 | | <td>Kapasitas Mesin</td>
19 | | <td>Muatan Maksimal</td>
20 | | <td>Tarif Awal</td>
21 | | <td>Tarif Per Km</td>
22 | | <td>Jumlah Sadel</td>
23 | | <td>Jumlah Gir</td>
24 | </tr>
25 | <tr>
26 | | <td>1</td>
27 | | <td>Truk 1</td>
28 | | <td><?= $truk1->getJmlRoda(); ?></td>
29 | | <td><?= $truk1->getWarna(); ?></td>
30 | | <td><?= $truk1->getBahanBakar(); ?></td>
31 | | <td><?= $truk1->getKapasitasMesin(); ?></td>
32 | | <td><?= $truk1->getMuatanMaks(); ?></td>

```

```

33 | <td></td>
34 | <td></td>
35 | <td></td>
36 | <td></td>
37 </tr>
38 <tr>
39 | <td>2</td>
40 | <td>Truk 2</td>
41 | | <td><?= $truk2->getJmlRoda(); ?></td>
42 | | <td><?= $truk2->getWarna(); ?></td>
43 | | <td><?= $truk2->getBahanBakar(); ?></td>
44 | | <td><?= $truk2->getKapasitasMesin(); ?></td>
45 | | <td><?= $truk2->getMuatanMaks(); ?></td>
46 | | <td></td>
47 | | <td></td>
48 | | <td></td>
49 | | <td></td>
50 </tr>
51 <tr>
52 | <td>3</td>
53 | <td>Taksi 1</td>
54 | | <td><?= $taksi1->getJmlRoda(); ?></td>
55 | | <td><?= $taksi1->getWarna(); ?></td>
56 | | <td><?= $taksi1->getBahanBakar(); ?></td>
57 | | <td><?= $taksi1->getKapasitasMesin(); ?></td>
58 | | <td></td>
59 | | <td><?= $taksi1->getTarifAwal(); ?></td>
60 | | <td><?= $taksi1->getTarifPerKm(); ?></td>
61 | | <td></td>
62 | | <td></td>
63 </tr>

```

```

64 |         <tr>
65 |             <td>4</td>
66 |             <td>Taksi 2</td>
67 |             <td><?= $taksi2->getJmlRoda(); ?></td>
68 |             <td><?= $taksi2->getWarna(); ?></td>
69 |             <td><?= $taksi2->getBahanBakar(); ?></td>
70 |             <td><?= $taksi2->getKapasitasMesin(); ?></td>
71 |             <td><\/td>
72 |             <td><?= $taksi2->getTarifAwal(); ?></td>
73 |             <td><?= $taksi2->getTarifPerKm(); ?></td>
74 |             <td><\/td>
75 |             <td><\/td>
76 |         <\/tr>
77 |         <tr>
78 |             <td>5</td>
79 |             <td>Sepeda 1</td>
80 |             <td><?= $sepeda1->getJmlRoda(); ?></td>
81 |             <td><?= $sepeda1->getWarna(); ?></td>
82 |             <td><\/td>
83 |             <td><\/td>
84 |             <td><\/td>
85 |             <td><\/td>
86 |             <td><\/td>
87 |             <td><?= $sepeda1->getJmlSadel(); ?></td>
88 |             <td><?= $sepeda1->getJmlGir(); ?></td>
89 |         <\/tr>
90 |         <tr>
91 |             <td>6</td>
92 |             <td>Sepeda 2</td>

```

← → ↻ ⓘ localhost/pbo/NOMOR%20B.php

No	Object	Jumlah Roda	Warna	Bahan Bakar	Kapasitas Mesin	Muatan Maksimal	Tarif Awal	Tarif Per KM	Jumlah Sadel	Jumlah Gir
1	Truk 1	4	Kuning	Solar	1500	1000	-	-	-	-
2	Truk 2	6	Merah	Solar	2000	5000	-	-	-	-
3	Taksi 1	4	Oranye	Bensin	1500	-	10000	5000	-	-
4	Taksi 2	4	Biru	Bensin	1300	-	7000	3500	-	-
5	Sepeda 1	3	Hitam	-	-	-	-	-	1	2
6	Sepeda 2	2	Putih	-	-	-	-	-	2	5

Program pada soal nomor 2 merupakan contoh program yang menerapkan konsep inheritance, overriding method dan keyword super. Pertama ada class Kendaraan yaitu class yang sebagai induk. Class Kendaraan mempunyai variable roda dan warna. Kemudian anak dari class Kendaraan tersebut ada dua yaitu class Mobil dan class Sepeda. Jadi kedua anak class tersebut mempunyai variable roda dan warna yang diturunkan dari class Kendaraan tadi. Penulisan class seperti biasa menggunakan extends. Syntax-nya sebagai berikut public class Mobil extends Kendaraan dan public class Sepeda extends Kendaraan. Class Mobil memiliki variable bahan bakar dan kapasitas mesin dan Class Sepeda mempunyai variable sadel dan gir. Kemudian class mobil menurunkan sifatnya kepada class Truk dan Taksi. Jadi kedua class ini mempunyai data variabel dari mobil dan kendaraan. Penulisan class seperti biasa menggunakan extends. Syntax-nya sebagai berikut public class Truk extends Mobil dan public class Taksi extends Mobil. Class Truk mempunyai

variable muatan maksimal dan class Taksi mempunyai variable tarif awal dan tarif per kilometer. Class Sepeda tidak menurunkan sifatnya karena kendaraan dalam UML yang lain tidak