

LAPORAN PRAKTIKUM 6

Mata Kuliah Pemograman Berorientasi Obyek



Disusun Oleh :

Rifqi Yudo Dewantoro

(21091397074)

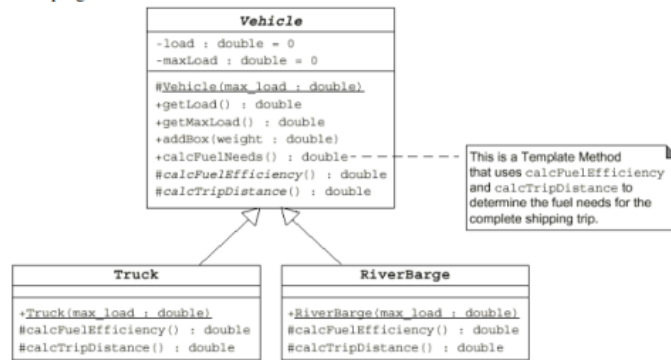
PROGRAM STUDI D4 MANAJEMEN INFORMATIKA

FAKULTAS VOKASI

UNIVERSITAS NEGERI SURABAYA

2022

1. Buat program berdasarkan UML berikut



1.

Source Code



PHP

```

C:\xampp\htdocs> prak 6 > No1 C.php
1 <!-- Nama : Rifqi yudo dewantoro
2 NIM : 21091397074
3 Kelas : 2021 B-->
4
5 <?php
6
7 require_once 'No1 A.php';
8
9 class Truck extends Vehicle {
10     public function __construct($maxLoad, $name)
11     {
12         $this->maxLoad = $maxLoad;
13         $this->name = $name;
14     }
15
16     public function calcFuelNeeds()
17     {
18         $fuel = $this->calcFuelEfficiency();
19         $trip = $this->calcTripDistance();
20
21         return ceil($fuel / $trip);
22     }
23 }
24
25 class RiverBarge extends Vehicle {
26     public function __construct($maxLoad, $name)
27     {
28         $this->maxLoad = $maxLoad;
29         $this->name = $name;
30     }
31
32     public function calcFuelNeeds()
33     {
34         $fuel = $this->calcFuelEfficiency();
35         $trip = $this->calcTripDistance();
36
37         return ceil($fuel / $trip);
38     }
39 }
40
41 $truck = new Truck(16000, 'Truk');
42 $riverBarge = new RiverBarge(15000, 'Perahu');
43
44 
```

```

25 class RiverBarge extends Vehicle {
26     public function __construct($maxLoad, $name)
27     {
28         $this->maxLoad = $maxLoad;
29         $this->name = $name;
30     }
31
32     public function calcFuelNeeds()
33     {
34         $fuel = $this->calcFuelEfficiency();
35         $trip = $this->calcTripDistance();
36
37         return ceil($fuel / $trip);
38     }
39 }
40
41 $truck = new Truck(16000, 'Truk');
42 $riverBarge = new RiverBarge(15000, 'Perahu');
43
44 
```



Abstract PHP

```

C:\xampp\htdocs> prak 6 > No1 A.php
1 <!-- Nama : Rifqi yudo dewantoro
2 NIM : 21091397074
3 Kelas : 2021 B-->
4
5 <?php
6
7 abstract class Vehicle {
8     private $load = 0;
9     protected $maxLoad = 0, $name;
10
11     protected function __construct($maxLoad, $name) {
12         $this->maxLoad = $maxLoad;
13         $this->name = $name;
14     }
15
16     public function getLoad() {
17         return $this->load;
18     }
19
20     public function getMaxLoad() {
21         echo 'Maksimal muatan ' . $this->name . ' ' ;
22         return $this->maxLoad;
23     }
24
25 
```

```

25     public function addBox($weight) {
26         if ($this->load >= $this->maxLoad) {
27             echo "$this->name menambah muatan sebesar $weight <br>";
28             echo 'Muatan telah penuh tidak bisa menambah lagi';
29         } else {
30             $this->load += $weight;
31             echo "$this->name menambah muatan sebesar $weight";
32         }
33     }
34
35     abstract public function calcFuelNeeds();
36
37     protected function calcFuelEfficiency() {
38         $range = 50000000;
39         $range /= $this->load;
40         return $range;
41     }
42
43     protected function calcTripDistance() {
44         return 500;
45     }
46 }
47
48 
```

Index PHP

```
C:\> xampp > htdocs > prak 6 > No18.php
1  <!-- Nama      : Rifqi yudo dewantoro
2      NIM       : 21091397074
3      Kelas    : 2021 B-->
4
5  <?php
6      require_once 'No1 C.php';
7  ?>
8
9  <!DOCTYPE html>
10 <html lang="id">
11
12 <head>
13     <!-- Bootstrap CSS -->
14     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
15           integrity="sha384-18mE4kbQ78iYhF1dvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqy12QvZ6jIW3" crossorigin="anonymous">
16
17     <title>PBO Praktikum 6</title>
18 </head>
19
20 <body>
21     <div class="container">
22         <br>
23         <h2 class="text-center">PBO - Praktikum 6</h2>
24         <div class="row">
25             <div class="col-5 mx-auto border p-3 mt-2">
26                 <h4 class="text-center"><strong>Soal 1</strong></h4>
27                 <br>
28                 <b><?= $truck->getMaxLoad() . ' kg'; ?> <br></b>
29                 <br>
30                 <?= $truck->addBox(1000) . ' kg'; ?> <br>
31                 <?= $truck->addBox(7000) . ' kg'; ?> <br>
32                 <?= $truck->addBox(8000) . ' kg'; ?> <br>
33
34                 <?php
35                     echo "Jadi, Butuh Bahan Bakar sebanyak " . $truck->calcFuelNeeds() . ' Liter'. '<br>';
36                 ?>
37                 <br>
38                 -----
39                 <br>
40                 <br>
41                 <b><?= $riverBarge->getMaxLoad() . ' kg'; ?> <br></b>
42                 <br>
43                 <?= $riverBarge->addBox(2000) . ' kg'; ?> <br>
44                 <?= $riverBarge->addBox(4000) . ' kg'; ?> <br>
45                 <?= $riverBarge->addBox(9000) . ' kg'; ?> <br>
46
47                 <?php
48                     echo "Jadi, Butuh Bahan Bakar sebanyak " . $riverBarge->calcFuelNeeds() . ' Liter';
49                 ?>
50             </div>
51         </div>
52     </div>
53 </body>
54
55 </html>
```

Output

PBO - Praktikum 6

Soal 1

Maksimal muatan Truk 16000 kg

Truk menambah muatan sebesar 1000 kg
Truk menambah muatan sebesar 7000 kg
Truk menambah muatan sebesar 8000 kg
Jadi, Butuh Bahan Bakar sebanyak 7 Liter

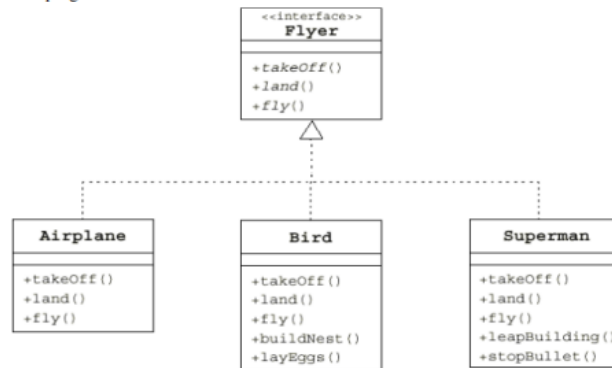
Maksimal muatan Perahu 15000 kg

Perahu menambah muatan sebesar 2000 kg
Perahu menambah muatan sebesar 4000 kg
Perahu menambah muatan sebesar 9000 kg
Jadi, Butuh Bahan Bakar sebanyak 7 Liter

Analisa

Implementasi dari abstract class pada class Vehicle, method calcFuelNeeds digunakan untuk menghitung bahan bakar yang digunakan. Abstract method di letakkan pada class Vehicle sebagai parent class dan diakses oleh child classnya yaitu class Truk, dan class RiverBarge yang akan mengembalikan nilai yang dihasilkan dari pembagian 2 method yaitu calcFuelEfficiency dan calcTripDistance

2. Buat program berdasarkan UML berikut



2.

Source Code

PHP

```
no 2.php x index no2.php interface no2.php
C:\> xampp > htdocs > prak 6 > no 2.php
1 |!- Rifqi Yudo Dewantoro
2 | 21091397074 -->
3
4 <?php
5
6 require_once 'interface no2.php';
7
8 class Airplane implements Flyer {
9     public function takeOff() {
10         return 'Pesawat lepas landas..';
11     }
12
13     public function land() {
14         return 'Pesawat mendarat';
15     }
16
17     public function fly() {
18         return 'Pesawat dalam perjalanan';
19     }
20 }
21
22 class Bird implements Flyer {
23     public function takeOff() {
24         return 'Burung mencari makan';
25     }
26
27     public function land() {
28         return 'Burung kembali pulang';
29     }
30
31     public function fly() {
32         return 'Burung terbang';
33 }
```

```
Run Terminal Help no 2 prak6.php - PBO - Visual Studio Code
no 2 prak6.php x
no 2 prak 6 > no 2 prak6.php > ...
33
34 public function buildNest() {
35     return 'Burung membuat sarang';
36 }
37
38 public function layEggs() {
39     return 'Burung bertelur';
40 }
41 }
42
43 class Superman implements Flyer {
44     public function takeOff() {
45         return 'Superman mengejar Batman';
46     }
47
48     public function land() {
49         return 'Superman melawan Batman';
50     }
51
52     public function fly() {
53         return 'Superman melancarkan pukulan';
54     }
55
56     public function leapBuilding() {
57         return 'Batman terpentak menabrak bangunan pencakar langit';
58     }
59
60     public function stopBullet() {
61         return 'Polisi menembaki superman namun ditangkis';
62     }
63 }
64
```

```
65 $airplane = new Airplane;
66 $bird = new Bird;
67 $superman = new Superman;
```

Interface PHP

```
no 2.php index no2.php interface no2.php X
C: > xampp > htdocs > prak 6 > interface no2.php
1 <!-- Rifqi Yudo Dewantoro
2 | 21091397074 -->
3
4 <?php
5
6 interface Flyer {
7     public function takeOff();
8     public function land();
9     public function fly();
10 }
11
12 interface Sailer {
13     public function dock();
14     public function cruise();
15 }
```

Index PHP

```
no 2.php index no2.php X interface no2.php
C: > xampp > htdocs > prak 6 > index no2.php
1 <!-- Rifqi Yudo Dewantoro
2 | 21091397074 -->
3
4 <?php
5     require_once 'no 2.php';
6 ?>
7
8 <!DOCTYPE html>
9 <html lang="en">
10
11 <head>
12     <!-- Bootstrap CSS -->
13     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
14         integrity="sha384-1BmE4kWBq78iYhFtdvKuhfTAU6auU8tT94WtHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin=
15
16     <title>Praktikum 6</title>
17 </head>
18
19 <body>
20     <div class="container">
21         <br>
22         <div class="row">
23             <div class="col-5 mx-auto border p-3 mt-2">
24                 <h4 class="text-center"><strong><u>Soal 2</u></strong></h4>
25                 <br><br>
26                 <b><?php
27                     echo "Superman";
28                     ?></b> <br>
29                     <?= $superman->land(); ?> <br>
30                     <?= $superman->takeOff(); ?> <br>
31                     <?= $superman->fly(); ?> <br>
32                     <?= $superman->leapBuilding(); ?> <br>
33                 </b></div>
34                 <b><?php
35                     echo "Bird";
36                     ?></b> <br>
37                     <?= $bird->buildNest(); ?> <br>
38                     <?= $bird->takeOff(); ?> <br>
39                     <?= $bird->fly(); ?> <br>
40                     <?= $bird->land(); ?> <br>
41                     <?= $bird->layEggs(); ?> <br>
42                     <br>
43                     <b><?php
44                         echo "Airplane";
45                         ?></b> <br>
46                         <?= $airplane->takeOff(); ?> <br>
47                         <?= $airplane->fly(); ?> <br>
48                         <?= $airplane->land(); ?> <br>
49                     </b></div>
50                 </div>
51             </div>
52         </body>
53
54 </html>
```

Output

Soal 2

Superman

Superman melawan Batman
Superman mengejar Batman
Superman melancarkan pukulan
Batman terpental menabrak bangunan pencakar langit
Polisi menembaki superman namun ditangkis

Bird

Burung membuat sarang
Burung mencari makan
Burung terbang
Burung kembali pulang
Burung bertelur

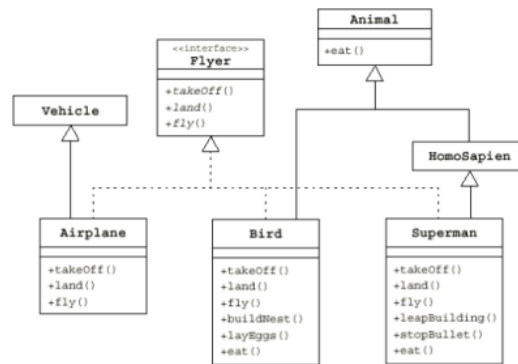
Airplane

Pesawat lepas landas..
Pesawat dalam perjalanan
Pesawat mendarat

Analisa

Implementasi Polymorphism dengan penggunaan Interface Flyer sehingga semua class yang Implements dari interface Flyer harus memiliki method takeoff, land, dan fly.

3. Buat program berdasarkan UML berikut



3.

Source Code

Interface PHP

```
no 3.php interface no 3.php X inde
C: > xampp > htdocs > prak 6 > interface no 3.php
1 <!-- Rifqi Yudo Dewantoro
2 21091397074 -->
3
4 <?php
5
6 interface Flyer {
7     public function takeOff();
8     public function land();
9     public function fly();
10 }
11
12 interface Sailer {
13     public function dock();
14     public function cruise();
15 }
```

PHP

```
no 3.php X interface no 3.php index no 3.php abstrac no 3.php
C: > xampp > htdocs > prak 6 > no 3.php
1 <!-- Rifqi Yudo Dewantoro
2 21091397074 -->
3
4 <?php
5
6 require_once 'abstrac no 3.php';
7 require_once 'interface no 3.php';
8
9 class Animal
10 {
11     protected $name;
12
13     public function __construct($name)
14     {
15         $this->name = $name;
16     }
17
18     public function eat()
19     {
20         return $this->name . ' sedang makan';
21     }
22 }
23
24 class Homosapiens extends Animal {}
25
26 class Airplane2 extends Vehicle implements Flyer
27 {
28     public function __construct($maxLoad, $name)
29     {
30         $this->maxLoad = $maxLoad;
31         $this->name = $name;
32     }
33 }
```

```
33     public function takeOff()
34     {
35         return $this->name lepas landas";
36     }
37
38     public function land()
39     {
40         return $this->name mendarat";
41     }
42
43     public function fly()
44     {
45         return $this->name dalam perjalanan";
46     }
47
48     public function calcFuelNeeds()
49     {
50         $fuel = $this->calcFuelEfficiency();
51         $trip = $this->calcTripDistance();
52
53
54         return ceil($fuel / $trip);
55     }
56 }
57
58
59 class Superman2 extends Homosapiens implements Flyer
60 {
61     public function takeOff()
62     {
63         return $this->name mengejar Batman";
64     }
```

```

65
66     public function land()
67     {
68         return "$this->name melawan Batman";
69     }
70
71     public function fly()
72     {
73         return "$this->name melancarkan pukulan";
74     }
75
76     public function leapBuilding()
77     {
78         return "Batman terpenjalar menabrak bangunan pencakar langit";
79     }
80
81     public function stopBullet()
82     {
83         return "Polisi menembaki $this->name namun ditangkis";
84     }
85 }
86
87 $burung = new Animal('Burung');
88 $manusia = new Homosapiens('Aransha');
89 $airplane2 = new Airplane2(25000, 'Batik Air');
90 $superman2 = new Superman2('Superman');

```

Abstract PHP

```

no 3.php  interface no 3.php  index no 3.php  abstrac no 3.php x
C:\xampp\htdocs>prak 6> abstrac no 3.php
1  |!-- Rifqi Yudo Dewantoro
2  |21091397074 -->
3
4  <?php
5
6  abstract class Vehicle {
7      private $load = 0;
8      protected $maxLoad = 0, $name;
9
10     protected function __construct($maxLoad, $name) {
11         $this->$maxLoad = $maxLoad;
12         $this->$name = $name;
13     }
14
15     public function getLoad() {
16         return $this->load;
17     }
18
19     public function getMaxLoad() {
20         echo "Maksimal muatan ' . $this->name . ' ";
21         return $this->maxLoad;
22     }
23
24     public function addBox($weight) {
25         if ($this->load >= $this->maxLoad) {
26             echo "$this->name menambah muatan sebesar $weight <br>";
27             echo "Muatan telah penuh tidak bisa menambah lagi!";
28         } else {
29             $this->load += $weight;
30             echo "$this->name menambah muatan sebesar $weight";
31         }
32     }

```

```

33     abstract public function calcFuelNeeds();
34
35     protected function calcFuelEfficiency() {
36         $range = 50000000;
37         $range /= $this->load;
38         return $range;
39     }
40
41     protected function calcTripDistance() {
42         return 500;
43     }
44 }

```

Index PHP


```
no 3.php interface no 3.php index no 3.php X abstrac no 3.php
C: > xampp > htdocs > prak 6 > index no 3.php
1 <!-- Rifqi Yudo Dewantoro
2 | 21091397074 -->
3
4 <?php
5 |     require_once 'no 3.php';
6 |
7 |
8 <!DOCTYPE html>
9 <html lang="id">
10
11 <head>
12 |     <!-- Bootstrap CSS -->
13 |     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="styles
14 |     integrity="sha384-18mE4kWBq781YhF1dvKuhF7AU6auU8tT94WrHffjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin=
15 |
16 |     <title>Praktikum 6</title>
17 </head>
18 <body>
19 |     <div class="container">
20 |         <div class="row">
21 |             <div class="col-5 mx-auto border p-3 mt-2">
22 |                 <h4 class="text-center"><strong><u>Soal 3</u></strong></h4>
23 |                 <br><br>
24 |                 <?= $burung->eat(); ?> <br>
25 |                 <?= $manusia->eat(); ?> <br>
26 |                 <br>
27 |
28 |                 <b><?= $airplane2->getMaxLoad() . ' kg'; ?> <br></b>
29 |                 <?= $airplane2->addBox(6000) . ' kg'; ?> <br>
30 |                 <?= $airplane2->addBox(2000) . ' kg'; ?> <br>
31 |                 <?= $airplane2->addBox(7000) . ' kg'; ?> <br>
32 |                 <?= $airplane2->addBox(5000) . ' kg'; ?> <br>
33 |
34 |                 <?= $airplane2->fly(); ?> <br>
35 |                 <?= $airplane2->land(); ?> <br>
36 |
37 |                 <?php
38 |                     echo "Jadi, Butuh Bahan Bakar sebanyak " . $airplane2->calcFuelNeeds() . ' Liter'. "<br>";
39 |                 ?>
40 |                 <br>
41 |                 <?= $superman2->eat(); ?> <br>
42 |                 <?= $superman2->land(); ?> <br>
43 |                 <?= $superman2->takeOff(); ?> <br>
44 |                 <?= $superman2->fly(); ?> <br>
45 |                 <?= $superman2->leapBuilding(); ?> <br>
46 |                 <?= $superman2->stopBullet(); ?> <br>
47 |             </div>
48 |         </div>
49 |     </body>
50 |
51 </html>
```

Output

Soal 3

Burung sedang makan
Aransha sedang makan

Maksimal muatan Batik Air 25000 kg

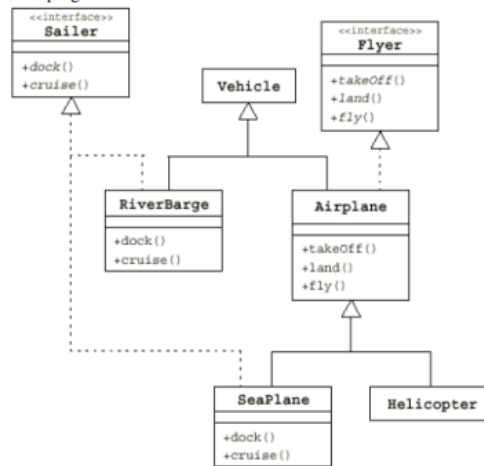
Batik Air menambah muatan sebesar 6000 kg
Batik Air menambah muatan sebesar 2000 kg
Batik Air menambah muatan sebesar 7000 kg
Batik Air menambah muatan sebesar 5000 kg
Batik Air lepas landas
Batik Air dalam perjalanan
Batik Air mendarat
Jadi, Butuh Bahan Bakar sebanyak 5 Liter

Superman sedang makan
Superman melawan Batman
Superman mengejar Batman
Superman melancarkan pukulan
Batman terpental menabrak bangunan pencakar langit
Polisi menembaki Superman namun ditangkis

Analisa

Terdapat interface Flyer dan abstract class Vehicle. Class airplane implementasi dari interface Flyer dan turunan dari Vehicle sehingga class Airplane harus memiliki method calcFuelNeeds, takeoff, land, dan fly. Class Bird implementasi dari Flyer dan turunan dari Animal sehingga memiliki method takeoff, land, fly, dan eat. Class Superman turunan dari homosapiens yang juga turunan dari Animal, serta implementasi dari interface Flyer. Maka class Superman memiliki method eat, takeoff, land, fly.

4. Buat program berdasarkan UML berikut



4.

Source Code



Interface PHP

```

C: > xampp > htdocs > prak 6 > No4 C.php
1 <!-- Nama : Rifqi yudo dewantoro
2 NIM : 21091397074
3 Kelas : 2021 b-->
4
5 <?php
6
7 interface Flyer {
8     public function takeOff();
9     public function land();
10    public function fly();
11 }
12
13 interface Sailer {
14     public function dock();
15     public function cruise();
16 }
    
```



PHP

```

C:\> xampp > htdocs > prak 6 > No4D.php
1  k!--- Nama : Rifqi yudo dewantoro
2      NIM : 21091397074
3      Kelas : 2021 B--->
4
5  <?php
6
7  require_once 'No4 A.php';
8  require_once 'No4 C.php';
9
10 class RiverBarge2 extends Vehicle implements Sailer {
11     public function __construct($maxLoad, $name) {
12         $this->maxLoad = $maxLoad;
13         $this->name = $name;
14     }
15
16     public function calcFuelNeeds() {
17         $fuel = $this->calcFuelEfficiency();
18         $trip = $this->calcTripDistance();
19
20         return ceil($fuel /= $trip);
21     }
22
23     public function dock() {
24         return $this->name . ' berada di dermaga';
25     }
26
27     public function cruise() {
28         return $this->name . ' sedang berlayar';
29     }
30 }
31
32 class Airplane2 implements Flyer {

```

```

32 class Airplane2 implements Flyer {
33     public function takeOff() {
34         return 'Pesawat lepas landas';
35     }
36     public function land() {
37         return 'Pesawat mendarat';
38     }
39     public function fly() {
40         return 'Pesawat dalam perjalanan';
41     }
42 }
43
44 class SeaPlane extends Vehicle implements Sailer {
45     public function __construct($maxLoad, $name) {
46         $this->maxLoad = $maxLoad;
47         $this->name = $name;
48     }
49
50     public function calcFuelNeeds() {
51         $fuel = $this->calcFuelEfficiency();
52         $trip = $this->calcTripDistance();
53
54         return ceil($fuel /= $trip);
55     }
56
57     public function dock() {
58         return $this->name . ' berada di dermaga';
59     }
60
61     public function cruise() {
62         return $this->name . ' sedang berlayar';
63     }
64 }

```

```

65     public function takeOff() {
66         return $this->name . ' lepas landas';
67     }
68
69     public function land() {
70         return $this->name . ' mendarat';
71     }
72
73     public function fly() {
74         return $this->name . ' dalam perjalanan';
75     }
76 }
77
78 class Helicopter extends Vehicle {
79     public function __construct($maxLoad, $name) {
80         $this->maxLoad = $maxLoad;
81         $this->name = $name;
82     }
83
84     public function calcFuelNeeds() {
85         $fuel = $this->calcFuelEfficiency();
86         $trip = $this->calcTripDistance();
87
88         return ceil($fuel /= $trip);
89     }
90     public function takeOff() {
91         return $this->name . ' lepas landas';
92     }
93
94     public function land() {
95         return $this->name . ' mendarat';
96     }
97 }

```

```

98     public function fly() {
99         return $this->name . ' dalam perjalanan';
100     }
101 }
102
103 $riverBarge2 = new RiverBarge2(30000, 'Atomic');
104 $seaPlane = new SeaPlane(20000, 'Titanic');
105 $helicopter = new Helicopter(10000, 'Brocklyn');

```

```

C: > xampp > htdocs > prak 6 > No4 A.php
1 <!-- Nama : Rifqi yudo dewantoro
2 NIM : 21091397074
3 Kelas : 2021 B--->
4
5 <?php
6
7 abstract class Vehicle {
8     private $load = 0;
9     protected $maxLoad = 0, $name;
10
11     protected function __construct($maxLoad, $name) {
12         $this->$maxLoad = $maxLoad;
13         $this->$name = $name;
14     }
15
16     public function getLoad() {
17         return $this->load;
18     }
19
20     public function getMaxLoad() {
21         echo 'Maksimal muatan ' . $this->name . ' ' ;
22         return $this->maxLoad;
23     }
24
25     public function addBox($weight) {
26         if ($this->load >= $this->maxLoad) {
27             echo "$this->name menambah muatan sebesar $weight <br>";
28             echo 'Muatan telah penuh tidak bisa menambah lagi';
29         }else {
30             $this->load += $weight;
31             echo "$this->name menambah muatan sebesar $weight";
32         }

```

```

33     }
34
35     abstract public function calcFuelNeeds();
36
37     protected function calcFuelEfficiency() {
38         $range = 50000000;
39         $range /= $this->load;
40         return $range;
41     }
42
43     protected function calcTripDistance() {
44         return 500;
45     }
46 }

```

Index PHP

```

C: > xampp > htdocs > prak 6 > No4 B.php
1 <!-- Nama : Rifqi yudo dewantoro
2 NIM : 21091397074
3 Kelas : 2021 B--->
4
5 <?php
6     require_once 'No4 D.php';
7 >
8
9 <!DOCTYPE html>
10 <html lang="en">
11
12 <head>
13     <!-- Bootstrap CSS -->
14     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
15         integrity="sha384-1BmE4kWBq78iYhFtdvKuhfTAU6au08ttT94WrRftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin=
16
17     <title>PBO - Praktikum 6</title>
18 </head>
19 <body>
20     <h2 class="text-center">PBO - Praktikum 6</h2>
21     <div class="container">
22         <div class="row">
23             <h4 class="text-center"><strong>Soal 4</strong></h4>
24             <br><br>
25             <div class="col-4 mx-auto border p-2 mt-2">
26                 <b><?= $riverBarge2->getMaxLoad() . ' kg'; ?> <br></b>
27                 <?= $riverBarge2->addBox(12000) . ' kg'; ?> <br>
28                 <?= $riverBarge2->addBox(14000) . ' kg'; ?> <br>
29                 <?= $riverBarge2->addBox(1000) . ' kg'; ?> <br>
30                 <?= $riverBarge2->addBox(3000) . ' kg'; ?> <br>
31                 <?= $riverBarge2->dock(); ?> <br>
32                 <?= $riverBarge2->cruise(); ?> <br>

```

```

33         <?php
34             echo "Jadi, Butuh Bahan Bakar sebanyak " . $riverBarge2->calcFuelNeeds() . ' Liter'. '<br>';
35         ?>
36     </div>
37     <div class="col-4 mx-auto border p-2 mt-2">
38         <b><?= $seaPlane->getMaxLoad() . ' kg'; ?> <br></b>
39         <?= $seaPlane->addBox(12000) . ' kg'; ?> <br>
40         <?= $seaPlane->addBox(8000) . ' kg'; ?> <br>
41         <?= $seaPlane->dock(); ?> <br>
42         <?= $seaPlane->cruise(); ?> <br>
43         <?= $seaPlane->takeOff(); ?> <br>
44         <?= $seaPlane->fly(); ?> <br>
45         <?= $seaPlane->land(); ?> <br>
46         <?php
47             echo "Jadi, Butuh Bahan Bakar sebanyak " . $seaPlane->calcFuelNeeds() . ' Liter'. '<br>';
48         ?>
49     </div>
50     <div class="col mx-auto border p-2 mt-2">
51         <b><?= $helicopter->getMaxLoad() . ' kg'; ?> <br></b>
52         <?= $helicopter->addBox(8000) . ' kg'; ?> <br>
53         <?= $helicopter->addBox(2000) . ' kg'; ?> <br>
54         <?= $helicopter->takeOff(); ?> <br>
55         <?= $helicopter->fly(); ?> <br>
56         <?= $helicopter->land(); ?> <br>
57         <?php
58             echo "Jadi, Butuh Bahan Bakar sebanyak " . $helicopter->calcFuelNeeds() . ' Liter'. '<br>';
59         ?>
60     </div>
61 </div>
62 </div>
63 </body>
64 </html>

```

Output

PBO - Praktikum 6

Soal 4

Maksimal muatan Atomic 30000 kg Atomic menambah muatan sebesar 12000 kg Atomic menambah muatan sebesar 14000 kg Atomic menambah muatan sebesar 1000 kg Atomic menambah muatan sebesar 3000 kg Atomic berada di dermaga Atomic sedang berlayar Jadi, Butuh Bahan Bakar sebanyak 4 Liter	Maksimal muatan Titanic 20000 kg Titanic menambah muatan sebesar 12000 kg Titanic menambah muatan sebesar 8000 kg Titanic berada di dermaga Titanic sedang berlayar Titanic lepas landas Titanic dalam perjalanan Titanic mendarat Jadi, Butuh Bahan Bakar sebanyak 5 Liter	Maksimal muatan Brooklyn 10000 kg Brooklyn menambah muatan sebesar 8000 kg Brooklyn menambah muatan sebesar 2000 kg Brooklyn lepas landas Brooklyn dalam perjalanan Brooklyn mendarat Jadi, Butuh Bahan Bakar sebanyak 10 Liter
--	--	--

Analisa

Implementasi polymorphism dengan interface dan abstract class ditunjukkan pada class SeaPlane yang implements interface Sailer, turunan dari class Airplane yang implements Flyer dan child dari Vehicle sehingga class SeaPlane memiliki method dock, cruise, takeoff, land, fly, dan calcFuelNeeds