

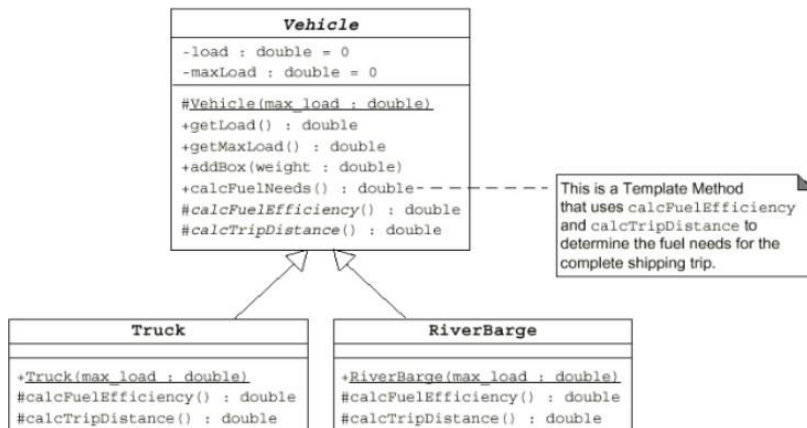
Nama : Dwi Ramadhaniasari

NIM : 21091397057

Kelas : 2021A

## TUGAS PRAKTIKUM 6

1. Buat program berdasarkan UML berikut.



- PHP

```
index no1.php      x      prak6 no1.php      x
1  <!-- Dwi ramadhaniasari
2  | 21091397057 -->
3
4  <?php
5      require_once 'prak6 no1.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="stylesheet"
14         integrity="sha384-18mE4kwBq78iYhF1dvKuhfTAU6auUstT94wrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
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 No.1</u></strong></h4>
25                 <br>
26                 <b><?> $truck->getMaxLoad() . ' kg'; ?> <br></b>
27                 <br>
28                 <?> $truck->addBox(2000) . ' kg'; ?> <br>
29                 <?> $truck->addBox(7000) . ' kg'; ?> <br>
30                 <?> $truck->addBox(9000) . ' kg'; ?> <br>
31
32                 <?php
33                     echo "Jadi, Butuh Bahan Bakar sebanyak " . $truck->calcFuelNeeds() . ' Liter'. '<br>';
34                 ?>
35                 <br>
36                 <hr>
37                 <br>
38                 <b><?> $riverBarge->getMaxLoad() . ' kg'; ?> <br></b>
39                 <br>
40                 <?> $riverBarge->addBox(5000) . ' kg'; ?> <br>
41                 <?> $riverBarge->addBox(7000) . ' kg'; ?> <br>
42                 <?> $riverBarge->addBox(8000) . ' kg'; ?> <br>
43
44                 <?php
45                     echo "Jadi, Butuh Bahan Bakar sebanyak " . $riverBarge->calcFuelNeeds() . ' Liter';
46                 ?>
47             </div>
48         </div>
49     </div>
50 </body>
51 </html>
52
```

```
prak6 no1.php x
1 <!-- Dwi ramadhaniasari
2 | 21091397057 -->
3 <?php
4
5 require_once 'abstract no1.php';
6
7 class Truck extends Vehicle {
8     public function __construct($maxLoad, $name)
9     {
10         $this->maxLoad = $maxLoad;
11         $this->name = $name;
12     }
13
14     public function calcFuelNeeds()
15     {
16         $fuel = $this->calcFuelEfficiency();
17         $trip = $this->calcTripDistance();
18
19         return ceil($fuel /= $trip);
20     }
21 }
22
23 class RiverBarge extends Vehicle {
24     public function __construct($maxLoad, $name)
25     {
26         $this->maxLoad = $maxLoad;
27         $this->name = $name;
28     }
29
30     public function calcFuelNeeds()
31     {
32         $fuel = $this->calcFuelEfficiency();
33         $trip = $this->calcTripDistance();
34
35         return ceil($fuel /= $trip);
36     }
37 }
38 $truck = new Truck(18000, 'Truk');
39 $riverBarge = new RiverBarge(20000, 'Tongkang Sungai');
```

- Output

**Soal No.1**

**Maksimal muatan Truk 18000 kg**

Truk menambah muatan sebesar 2000 kg  
 Truk menambah muatan sebesar 7000 kg  
 Truk menambah muatan sebesar 9000 kg  
 Jadi, Butuh Bahan Bakar sebanyak 6 Liter

---

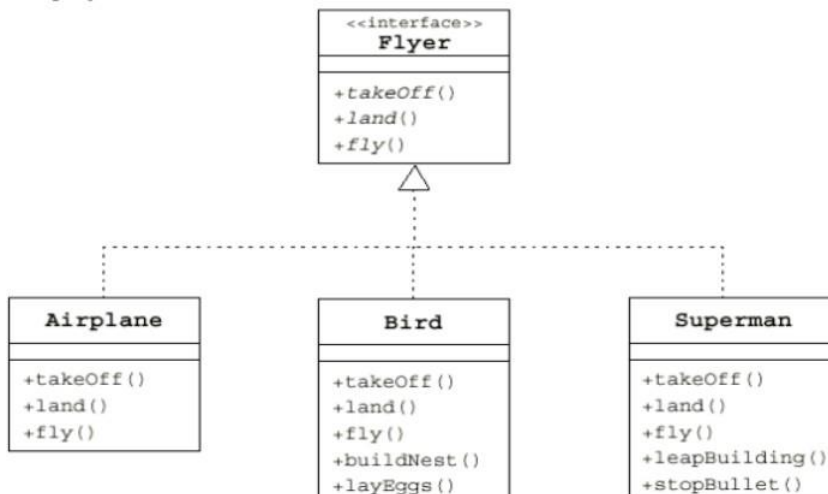
**Maksimal muatan Tongkang Sungai 20000 kg**

Tongkang Sungai menambah muatan sebesar 5000 kg  
 Tongkang Sungai menambah muatan sebesar 7000 kg  
 Tongkang Sungai menambah muatan sebesar 8000 kg  
 Jadi, Butuh Bahan Bakar sebanyak 5 Liter

- Analisa

Program tersebut merupakan 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.





```
no 2.php x
1 <!-- Dwi ramadhaniasari
2 : 21091397057 -->
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     }
34 }
```

```

35     public function buildNest() {
36         return 'Burung membuat sarang';
37     }
38
39     public function layEggs() {
40         return 'Burung bertelur';
41     }
42 }
43
44 class Superman implements Flyer {
45     public function takeOff() {
46         return 'Superman mengejar Batman';
47     }
48
49     public function land() {
50         return 'Superman melawan Batman';
51     }
52
53     public function fly() {
54         return 'Superman melancarkan pukulan';
55     }
56
57     public function leapBuilding() {
58         return 'Batman terpelantai menabrak bangunan pencakar langit';
59     }
60
61     public function stopBullet() {
62         return 'Polisi menembaki superman namun ditangkis';
63     }
64 }
65
66 $airplane = new Airplane;
67 $bird = new Bird;
68 $superman = new Superman;

```

```

interface no2.php
1  <!-- Dwi RamadhaniaSari
2      21091397057 -->
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 }

```

- Output

## Soal 2

### **Superman**

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

### **Bird**

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

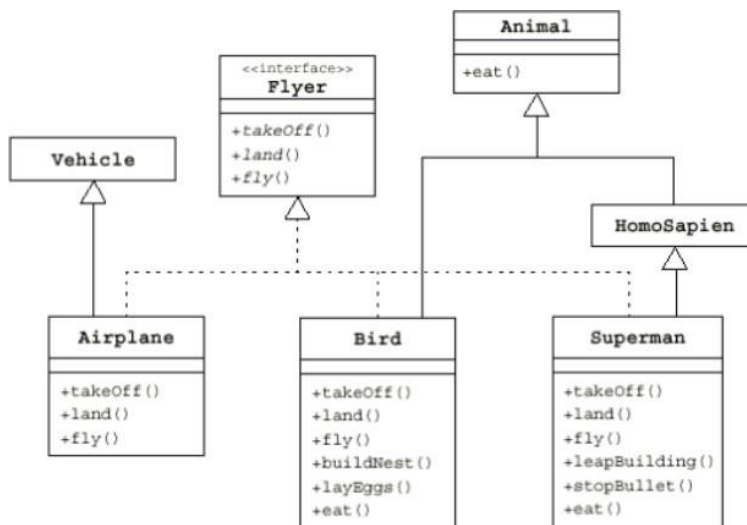
### **Airplane**

Pesawat lepas landas..  
 Pesawat dalam perjalanan  
 Pesawat mendarat

- Analisa

Program tersebut merupakan implementasi Polymorphism dengan penggunaan Interface Flyer. Sehingga semua class yang implementasi dari interface Flyer harus memiliki method take off, land, land, dan fly.

3. Buat program berdasarkan UML berikut.



- PHP

```

1 <!-- Dwi RamadhaniSari
2 21091397057 -->
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="stylesheet"
14         integrity="sha384-18m4K&W781VhFIdVKuHfAU6au8T94MrHfTjDbrCEXSU10BoQy12QVZ6jIM3" crossorigin="anonymous">
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                 <div class="text-center"><strong><u>Soal 3</u></strong></div>
23                 <br><br>
24                 <?=$burung->eat(); ?> <br>
25                 <?=$manusia->eat(); ?> <br>
26                 <br>
27                 <br><?=$airplane2->getMaxLoad() . ' kg'; ?> <br></div>
28                 <?=$airplane2->addBox(6000) . ' kg'; ?> <br>
29                 <?=$airplane2->addBox(2000) . ' kg'; ?> <br>
30                 <?=$airplane2->addBox(7000) . ' kg'; ?> <br>
31                 <?=$airplane2->addBox(5000) . ' kg'; ?> <br>
32                 <?=$airplane2->takeOff(); ?> <br>
33                 <?=$airplane2->fly(); ?> <br>
34                 <?=$airplane2->land(); ?> <br>
35                 <br>
36                 <?php
37                     echo "Jadi, Butuh Bahan Bakar sebanyak " . $airplane2->calcFuelNeeds() . ' Liter'. '<br>';
38                 >
39                 <br>
40                 <?=$superman2->eat(); ?> <br>
41                 <?=$superman2->land(); ?> <br>
42                 <?=$superman2->takeOff(); ?> <br>
43                 <?=$superman2->fly(); ?> <br>
44                 <?=$superman2->leapBuilding(); ?> <br>
45                 <?=$superman2->stopBullet(); ?> <br>
46             </div>
47         </div>
48     </div>
49 </body>
50 </html>
51
52

```

```

1 <?php
2
3 require_once 'abstrac no 3.php';
4 require_once 'Interface no 3.php';
5
6 class Animal
7 {
8     protected $name;
9
10     public function __construct($name)
11     {
12         $this->name = $name;
13     }
14
15     public function eat()
16     {
17         return $this->name . " sedang makan";
18     }
19 }
20
21 class Homosapiens extends Animal {}
22
23 class Airplane2 extends Vehicle implements Flyer
24 {
25     public function __construct($maxLoad, $name)
26     {
27         $this->maxLoad = $maxLoad;
28         $this->name = $name;
29     }
30
31     public function takeOff()
32     {
33         return $this->name lepas landas";
34     }
35
36     public function land()
37     {
38         return $this->name mendarat";
39     }
40
41     public function fly()
42     {
43         return $this->name dalam perjalanan";
44     }
45
46     public function calcFuelNeeds()
47     {
48         $fuel = $this->calcFuelEfficiency();
49
50         $strip = $this->calcTripDistance();
51
52         return ceil($fuel / $strip);
53     }
54 }
55
56
57 class Superman2 extends Homosapiens implements Flyer
58 {
59     public function takeOff()
60     {
61         return $this->name mengejar Batman";
62     }
63
64     public function land()
65     {
66         return $this->name melawan Batman";
67     }
68
69     public function fly()
70     {
71         return $this->name melancarkan pukulan";
72     }
73
74     public function leapBuilding()
75     {
76         return "Batman terpelempar menabrak bangunan pencakar langit";
77     }
78
79     public function stopBullet()
80     {
81         return "Polisi menembaki $this->name namun ditangkis";
82     }
83 }
84
85 $burung = new Animal('Burung');
86 $manusia = new Homosapiens('Aransha');
87 $airplane2 = new Airplane2(5000, 'latik Air');
88 $superman2 = new Superman2('Superman');

```



```
1 | i-- Del Ramadhaniasari
2 | 21091397057 -->
3 |
4 |
5 | <?php
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 "Maksimal muatan telah penuh tidak bisa menambah lagi";
27 |         } else {
28 |             $this->load += $weight;
29 |             echo "Maksimal muatan telah ditambah sebesar $weight";
30 |         }
31 |     }
32 |
33 |
34 |     abstract public function calcFuelNeeds();
35 |
36 |     protected function calcFuelEfficiency() {
37 |         $range = 50000000;
38 |         $range /= $this->load;
39 |         return $range;
40 |     }
41 |
42 |     protected function calcTripDistance() {
43 |         return 500;
44 |     }
45 | }
```

```
1 | <?php
2 |
3 | interface Flyer {
4 |     public function takeoff();
5 |     public function land();
6 |     public function fly();
7 | }
8 |
9 | interface Sailer {
10 |     public function dock();
11 |     public function cruise();
12 | }
```

- 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

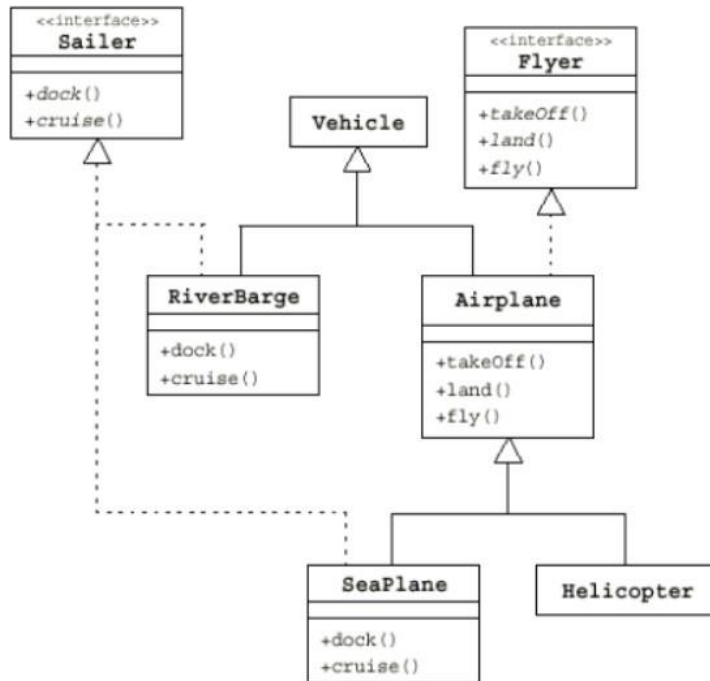
Superman sedang makan  
Superman melawan Batman  
Superman mengejar Batman  
Superman melancarkan pukulan  
Batman terpental menabrak bangunan pencakar langit  
Polisi menembaki Superman namun ditangkis

- Analisa

Pada Program tersebut 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 dan implementasi dari interface Flyer. Maka class Superman memiliki method eat, takeoff, land, fly.

#### 4. Buat Program UML



#### • PHP

```

1 <!-- Dwi Ramadhyaniasari
2 2109139705 -->
3
4
5 <?php
6     require_once 'prak 6 no 4.php';
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="sha384-18m40W8Q781YHfIdvunFtAUGAust194WfHtj0bCEXsU1080y12QvZ6jTm3" crossorigin="anonymous">
15
16     <title>Praktikum 6</title>
17 </head>
18 <body>
19     <div class="container">
20         <div class="row">
21             <div class="col-md-6">
22                 <strong><u>Soal No.4</u></strong></div>
23                 <div class="col-4 mx-auto border p-2 mt-2">
24                     <div><?php $riverBarge2->getMaxLoad(); ' kg'; ?></div>
25                     <div><?php $riverBarge2->addBox(5000); ' kg'; ?></div>
26                     <div><?php $riverBarge2->addBox(10000); ' kg'; ?></div>
27                     <div><?php $riverBarge2->addBox(8000); ' kg'; ?></div>
28                     <div><?php $riverBarge2->addBox(2000); ' kg'; ?></div>
29                     <div><?php $riverBarge2->dock(); ?></div>
30                     <div><?php $riverBarge2->cruise(); ?></div>
31                     <div><?php
32                         echo "Jadi, Butuh Bahan Bakar sebanyak " . $riverBarge2->calcFuelNeeds() . " Liter". "<br>";
33                     ?>
34                 </div>
35                 <div class="col-4 mx-auto border p-2 mt-2">
36                     <div><?php $seaPlane->getMaxLoad(); ' kg'; ?></div>
37                     <div><?php $seaPlane->addBox(5000); ' kg'; ?></div>
38                     <div><?php $seaPlane->addBox(7000); ' kg'; ?></div>
39                     <div><?php $seaPlane->dock(); ?></div>
40                     <div><?php $seaPlane->cruise(); ?></div>
41                     <div><?php $seaPlane->takeOff(); ?></div>
42                     <div><?php $seaPlane->fly(); ?></div>
43                     <div><?php $seaPlane->land(); ?></div>
44                     <div><?php
45                         echo "Jadi, Butuh Bahan Bakar sebanyak " . $seaPlane->calcFuelNeeds() . " Liter". "<br>";
46                     ?>
47                 </div>
48                 <div class="col-4 mx-auto border p-2 mt-2">
49                     <div><?php $helicopter->getMaxLoad(); ' kg'; ?></div>
50                     <div><?php $helicopter->addBox(5000); ' kg'; ?></div>
51                     <div><?php $helicopter->addBox(7000); ' kg'; ?></div>
52                     <div><?php $helicopter->takeOff(); ?></div>
53                     <div><?php $helicopter->fly(); ?></div>
54                     <div><?php $helicopter->land(); ?></div>
55                     <div><?php
56                         echo "Jadi, Butuh Bahan Bakar sebanyak " . $helicopter->calcFuelNeeds() . " Liter". "<br>";
57                     ?>
58                 </div>
59             </div>
60         </div>
61     </div>
62 </body>
63 </html>

```

```

1 <!-- Dwi ramadhaniasari
2 21091397057 -->
3
4 <?php
5
6 require_once 'abstract no 4.php';
7 require_once 'interface no 4.php';
8
9 class RiverBarge2 extends Vehicle implements Soiler {
10     public function __construct($maxload, $name) {
11         $this->maxload = $maxload;
12         $this->name = $name;
13     }
14
15     public function calcFuelNeeds() {
16         $fuel = $this->calcFuelEfficiency();
17         $strip = $this->calcTripDistance();
18
19         return ceil($fuel / $strip);
20     }
21
22     public function dock() {
23         return $this->name . ' berada di dermaga';
24     }
25
26     public function cruise() {
27         return $this->name . ' sedang berlayar';
28     }
29 }
30
31 class Airplane2 implements Flyer {
32     public function takeOff() {
33         return 'Pesawat lepas landas';
34     }
35     public function land() {
36         return 'Pesawat mendarat';
37     }
38     public function fly() {
39         return 'Pesawat dalam perjalanan';
40     }
41 }
42
43 class SeaPlane extends Vehicle implements Soiler {
44     public function __construct($maxload, $name) {
45         $this->maxload = $maxload;
46         $this->name = $name;
47     }
48
49     public function calcFuelNeeds() {
50         $fuel = $this->calcFuelEfficiency();
51         $strip = $this->calcTripDistance();
52
53         return ceil($fuel / $strip);
54     }
55
56     public function dock() {
57         return $this->name . ' berada di dermaga';
58     }
59
60     public function cruise() {
61         return $this->name . ' sedang berlayar';
62     }
63
64     public function takeOff() {
65         return $this->name . ' lepas landas';
66     }
67
68     public function land() {
69         return $this->name . ' mendarat';
70     }
71
72     public function fly() {
73         return $this->name . ' dalam perjalanan';
74     }
75 }
76
77 class Helicopter extends Vehicle {
78     public function __construct($maxload, $name) {
79         $this->maxload = $maxload;
80         $this->name = $name;
81     }
82
83     public function calcFuelNeeds() {
84         $fuel = $this->calcFuelEfficiency();
85         $strip = $this->calcTripDistance();
86
87         return ceil($fuel / $strip);
88     }
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
99 <!-- Dwi ramadhaniasari
100 21091397057 -->
101
102 <?php
103
104 abstract class Vehicle {
105     private $load = 0;
106     protected $maxload = 0, $name;
107
108     protected function __construct($maxload, $name) {
109         $this->maxload = $maxload;
110         $this->name = $name;
111     }
112
113     public function getload() {
114         return $this->load;
115     }
116
117     public function getMaxload() {
118         echo 'Maksimal muatan ' . $this->name . ' : ' . $this->maxload;
119     }
120
121     public function addBox($weight) {
122         if ($this->load >= $this->maxload) {
123             echo "Muatan telah penuh tidak bisa menambah lagi";
124         } else {
125             $this->load += $weight;
126             echo "Muatan telah penuh tidak bisa menambah lagi";
127         }
128     }
129
130     abstract public function calcFuelNeeds();
131
132     protected function calcFuelEfficiency() {
133         $range = 500000000;
134         $range /= $this->load;
135         return $range;
136     }
137
138     protected function calcTripDistance() {
139         return 500;
140     }
141 }
142
143
144
145

```

- Output

**Soal No.4**

<b>Maksimal muatan Atomic 40000 kg</b> Atomic menambah muatan sebesar 15000 kg Atomic menambah muatan sebesar 10000 kg Atomic menambah muatan sebesar 8000 kg Atomic menambah muatan sebesar 2000 kg Atomic berada di dermaga Atomic sedang berlayar Jadi, Butuh Bahan Bakar sebanyak 3 Liter	<b>Maksimal muatan Titanic 30000 kg</b> Titanic menambah muatan sebesar 15000 kg Titanic menambah muatan sebesar 7000 kg Titanic berada di dermaga Titanic sedang berlayar Titanic lepas landas Titanic dalam perjalanan Titanic mendarat Jadi, Butuh Bahan Bakar sebanyak 5 Liter	<b>Maksimal muatan Brocklyn 15000 kg</b> Brocklyn menambah muatan sebesar 5000 kg Brocklyn menambah muatan sebesar 7000 kg Brocklyn lepas landas Brocklyn dalam perjalanan Brocklyn mendarat Jadi, Butuh Bahan Bakar sebanyak 9 Liter
--	--	---

- **Analisa**  
Program tersebut merupakan implementasi polymorphism dengan interface dan abstract class di tunjukkan 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.