# LAPORAN PRAKTIKUM 6

## Mata Kuliah Pemograman Berorientasi Obyek


Disusun Oleh :

Rifqi Yudo Dewantoro

(21091397074)

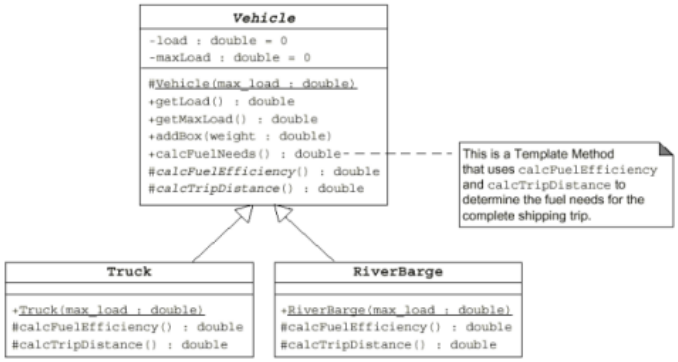
**PROGRAM STUDI D4 MANAJEMEN INFORMATIKA**

**FAKULTAS VOKASI**

**UNIVERSITAS NEGERI SURABAYA**

**2022**

1. Buat program berdasarkan UML berikut




**1.**

## Source Code

### PHP

C: > xampp > htdocs > prak 6 > No1 C.php

```php
<!--- Nama    : Rifqi yudo dewantoro
      NIM     : 21091397074
      Kelas   : 2021 B--->

<?php

require_once 'No1 A.php';

class Truck extends Vehicle {
    public function __construct($maxLoad, $name)
    {
        $this->maxLoad = $maxLoad;
        $this->name = $name;
    }

    public function calcFuelNeeds()
    {
        $fuel = $this->calcFuelEfficiency();
        $trip = $this->calcTripDistance();

        return ceil($fuel /= $trip);
    }
}

class RiverBarge extends Vehicle {
    public function __construct($maxLoad, $name)
    {
        $this->maxLoad = $maxLoad;
        $this->name = $name;
    }

    public function calcFuelNeeds()
```

```php
class RiverBarge extends Vehicle {
    public function __construct($maxLoad, $name)
    {
        $this->maxLoad = $maxLoad;
        $this->name = $name;
    }

    public function calcFuelNeeds()
    {
        $fuel = $this->calcFuelEfficiency();
        $trip = $this->calcTripDistance();

        return ceil($fuel /= $trip);
    }
}
$truck = new Truck(16000, 'Truk');
$riverBarge = new RiverBarge(15000, 'Perahu');
```

### Abstract PHP

C: > xampp > htdocs > prak 6 > No1 A.php

```php
<!--- Nama    : Rifqi yudo dewantoro
      NIM     : 21091397074
      Kelas   : 2021 B--->

<?php

abstract class Vehicle {
    private $load = 0;
    protected $maxLoad = 0, $name;

    protected function __construct($maxLoad, $name) {
        $this->$maxLoad = $maxLoad;
        $this->$name = $name;
    }

    public function getLoad() {
        return $this->load;
    }

    public function getMaxLoad() {
        echo 'Maksimal muatan ' . $this->name . ' ' ;
        return $this->maxLoad;
    }
```

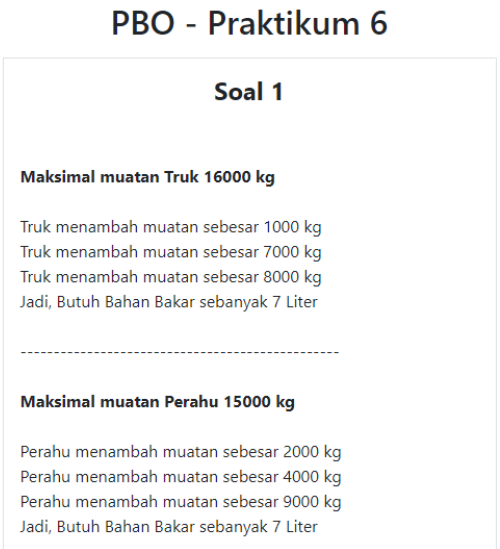
```php
    public function addBox($weight) {
        if ($this->load >= $this->maxLoad) {
            echo "$this->name menambah muatan sebesar $weight <br>";
            echo 'Muatan telah penuh tidak bisa menambah lagi';
        }else {
            $this->load += $weight;
            echo "$this->name menambah muatan sebesar $weight";
        }
    }

    abstract public function calcFuelNeeds();

    protected function calcFuelEfficiency() {
        $range = 50000000;
        $range /= $this->load;
        return $range;
    }

    protected function calcTripDistance() {
        return 500;
    }
}
```

## Index PHP

C: > xampp > htdocs > prak 6 > No1 B.php

```php
<!--- Nama    : Rifqi yudo dewantoro
      NIM     : 21091397074
      Kelas   : 2021 B--->

<?php
    require_once 'No1 C.php';
?>

<!DOCTYPE html>
<html lang="id">

<head>
    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="styles
    integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin=

    <title>PBO Praktikum 6</title>
</head>

<body>
    <div class="container">
        <br>
        <h2 class="text-center">PBO - Praktikum 6</h2>
        <div class="row">
            <div class="col-5 mx-auto border p-3 mt-2">
                <h4 class="text-center"><strong>Soal 1</strong></h4>
                <br><br>
                <b><?= $truck->getMaxLoad() . ' kg'; ?> <br></b>
                <br>
                <?= $truck->addBox(1000) . ' kg'; ?> <br>
                <?= $truck->addBox(7000) . ' kg'; ?> <br>
                <?= $truck->addBox(8000) . ' kg'; ?> <br>

                <?php
                    echo "Jadi, Butuh Bahan Bakar sebanyak " . $truck->calcFuelNeeds() . ' Liter'. '<br>';
                ?>
                <br>
                ---------------------------------------------
                <br>
                <br>
                <b><?= $riverBarge->getMaxLoad() . ' kg'; ?> <br></b>
                <br>
                <?= $riverBarge->addBox(2000) . ' kg'; ?> <br>
                <?= $riverBarge->addBox(4000) . ' kg'; ?> <br>
                <?= $riverBarge->addBox(9000) . ' kg'; ?> <br>

                <?php
                    echo "Jadi, Butuh Bahan Bakar sebanyak " . $riverBarge->calcFuelNeeds() . ' Liter';
                ?>
            </div>
        </div>
    </div>
</body>

</html>
```

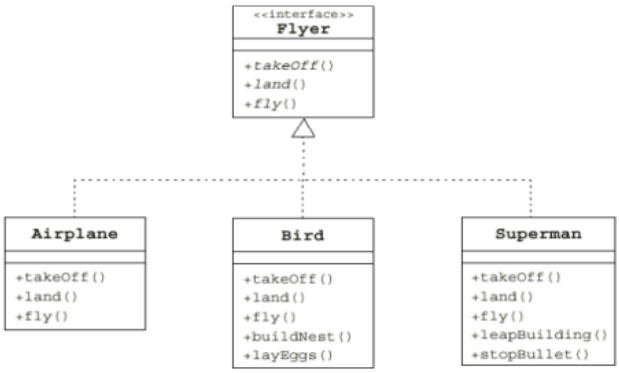
## Output



## 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 | index no2.php | interface no2.php

C: > xampp > htdocs > prak 6 > no 2.php

```php
<!-- Rifqi Yudo Dewantoro
     21091397074 -->

<?php

require_once 'interface no2.php';

class Airplane implements Flyer {
    public function takeOff() {
        return 'Pesawat lepas landas..';
    }

    public function land() {
        return 'Pesawat mendarat';
    }

    public function fly() {
        return 'Pesawat dalam perjalanan';
    }
}

class Bird implements Flyer {
    public function takeOff() {
        return 'Burung mencari makan';
    }

    public function land() {
        return 'Burung kembali pulang';
    }

    public function fly() {
        return 'Burung terbang';
    }
```

Run Terminal Help — no 2 prak6.php - PBO - Visual Studio Code

no 2 prak6.php

no 2 prak 6 > no 2 prak6.php > ...

```php
    public function buildNest() {
        return 'Burung membuat sarang';
    }

    public function layEggs() {
        return 'Burung bertelur';
    }
}

class Superman implements Flyer {
    public function takeOff() {
        return 'Superman mengejar Batman';
    }

    public function land() {
        return 'Superman melawan Batman';
    }

    public function fly() {
        return 'Superman melancarkan pukulan';
    }

    public function leapBuilding() {
        return 'Batman terpental menabrak bangunan pencakar langit';
    }

    public function stopBullet() {
        return 'Polisi menembaki superman namun ditangkis';
    }
}
```

```php
$airplane = new Airplane;
$bird = new Bird;
$superman = new Superman;
```

## Interface PHP

no 2.php | index no2.php | interface no2.php

C: > xampp > htdocs > prak 6 > interface no2.php

```php
<!-- Rifqi Yudo Dewantoro
     21091397074 -->

<?php

interface Flyer {
    public function takeOff();
    public function land();
    public function fly();
}

interface Sailer {
    public function dock();
    public function cruise();
}
```

## Index PHP

no 2.php | index no2.php | interface no2.php

C: > xampp > htdocs > prak 6 > index no2.php

```php
<!-- Rifqi Yudo Dewantoro
     21091397074 -->

<?php
    require_once 'no 2.php';
?>

<!DOCTYPE html>
<html lang="en">

<head>
    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="styles
    integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin=

    <title>Praktikum 6</title>
</head>

<body>
    <div class="container">
        <br>
        <div class="row">
            <div class="col-5 mx-auto border p-3 mt-2">
                <h4 class="text-center"><strong><u>Soal 2</u></strong></h4>
                <br><br>
                <b><?php
                        echo "Superman";
                    ?></b> <br>
                <?= $superman->land(); ?> <br>
                <?= $superman->takeOff(); ?> <br>
                <?= $superman->fly(); ?> <br>
                <?= $superman->leapBuilding(); ?> <br>
                <br>
                <b><?php
                        echo "Bird";
                    ?></b> <br>
                <?= $bird->buildNest(); ?> <br>
                <?= $bird->takeOff(); ?> <br>
                <?= $bird->fly(); ?> <br>
                <?= $bird->land(); ?> <br>
                <?= $bird->layEggs(); ?> <br>
                <br>
                <b><?php
                        echo "Airplane";
                    ?></b> <br>
                <?= $airplane->takeOff(); ?> <br>
                <?= $airplane->fly(); ?> <br>
                <?= $airplane->land(); ?> <br>
            </div>
        </div>
    </div>
</body>

</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

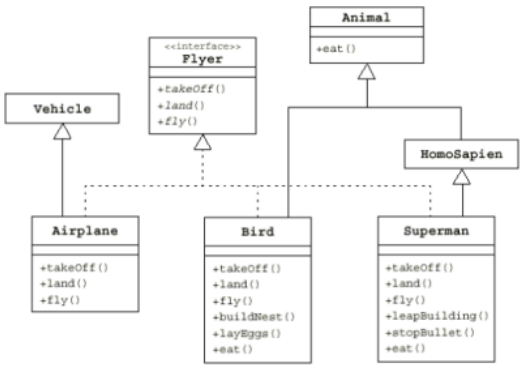
**Airplane**
Pesawat lepas landas..
Pesawat dalam perjalanan
Pesawat mendarat

## Analisa

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

**3.**

3. Buat program berdasarkan UML berikut




## Source Code

## Interface PHP

no 3.php | interface no 3.php | inde

C: > xampp > htdocs > prak 6 > interface no 3.php

```php
<!-- Rifqi Yudo Dewantoro
     21091397074 -->

<?php

interface Flyer {
    public function takeOff();
    public function land();
    public function fly();
}

interface Sailer {
    public function dock();
    public function cruise();
}
```

## PHP

no 3.php | interface no 3.php | index no 3.php | abstrac no 3.php

C: > xampp > htdocs > prak 6 > no 3.php

```php
<!-- Rifqi Yudo Dewantoro
     21091397074 -->

<?php

require_once 'abstrac no 3.php';
require_once 'interface no 3.php';

class Animal
{
    protected $name;

    public function __construct($name)
    {
        $this->name = $name;
    }

    public function eat()
    {
        return $this->name . ' sedang makan';
    }
}

class Homosapiens extends Animal {}

class Airplane2 extends Vehicle implements Flyer
{
    public function __construct($maxLoad, $name)
    {
        $this->maxLoad = $maxLoad;
        $this->name = $name;
    }
```

```php
    public function takeOff()
    {
        return "$this->name lepas landas";
    }

    public function land()
    {
        return "$this->name mendarat";
    }

    public function fly()
    {
        return "$this->name dalam perjalanan";
    }

    public function calcFuelNeeds()
    {
        $fuel = $this->calcFuelEfficiency();
        $trip = $this->calcTripDistance();



        return ceil($fuel /= $trip);
    }
}

class Superman2 extends Homosapiens implements Flyer
{
    public function takeOff()
    {
        return "$this->name mengejar Batman";
    }
```

```php
    public function land()
    {
        return "$this->name melawan Batman";
    }

    public function fly()
    {
        return "$this->name melancarkan pukulan";
    }

    public function leapBuilding()
    {
        return "Batman terpental menabrak bangunan pencakar langit";
    }

    public function stopBullet()
    {
        return "Polisi menembaki $this->name namun ditangkis";
    }
}

$burung = new Animal('Burung');
$manusia = new Homosapiens('Aransha');
$airplane2 = new Airplane2(25000, 'Batik Air');
$superman2 = new Superman2('Superman');
```

## Abstract PHP

no 3.php | interface no 3.php | index no 3.php | abstrac no 3.php

C: > xampp > htdocs > prak 6 > abstrac no 3.php

```php
<!-- Rifqi Yudo Dewantoro
     21091397074 -->

<?php

abstract class Vehicle {
    private $load = 0;
    protected $maxLoad = 0, $name;

    protected function __construct($maxLoad, $name) {
        $this->$maxLoad = $maxLoad;
        $this->$name = $name;
    }

    public function getLoad() {
        return $this->load;
    }

    public function getMaxLoad() {
        echo 'Maksimal muatan ' . $this->name . ' ' ;
        return $this->maxLoad;
    }

    public function addBox($weight) {
        if ($this->load >= $this->maxLoad) {
            echo "$this->name menambah muatan sebesar $weight <br>";
            echo 'Muatan telah penuh tidak bisa menambah lagi';
        }else {
            $this->load += $weight;
            echo "$this->name menambah muatan sebesar $weight";
        }
    }
```

```php
    abstract public function calcFuelNeeds();

    protected function calcFuelEfficiency() {
        $range = 50000000;
        $range /= $this->load;
        return $range;
    }

    protected function calcTripDistance() {
        return 500;
    }
}
```

## Index PHP

no 3.php | interface no 3.php | index no 3.php | abstrac no 3.php

C: > xampp > htdocs > prak 6 > index no 3.php

```php
<!-- Rifqi Yudo Dewantoro
    | 21091397074 -->

<?php
    require_once 'no 3.php';
?>

<!DOCTYPE html>
<html lang="id">

<head>
    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="styles
    integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin=

    <title>Praktikum 6</title>
</head>
<body>
    <div class="container">
        <div class="row">
            <div class="col-5 mx-auto border p-3 mt-2">
                <h4 class="text-center"><strong><u>Soal 3</u></strong></h4>
                <br><br>
                <?= $burung->eat(); ?> <br>
                <?= $manusia->eat(); ?> <br>
                <br>

                <b><?= $airplane2->getMaxLoad() . ' kg'; ?> <br></b>
                <?= $airplane2->addBox(6000) . ' kg'; ?> <br>
                <?= $airplane2->addBox(2000) . ' kg'; ?> <br>
                <?= $airplane2->addBox(7000) . ' kg'; ?> <br>
                <?= $airplane2->addBox(5000) . ' kg'; ?> <br>
                <?= $airplane2->fly(); ?> <br>
                <?= $airplane2->land(); ?> <br>

                <?php
                    echo "Jadi, Butuh Bahan Bakar sebanyak " . $airplane2->calcFuelNeeds() . ' Liter'. '<br>';
                ?>
                <br>
                <?= $superman2->eat(); ?> <br>
                <?= $superman2->land(); ?> <br>
                <?= $superman2->takeOff(); ?> <br>
                <?= $superman2->fly(); ?> <br>
                <?= $superman2->leapBuilding(); ?> <br>
                <?= $superman2->stopBullet(); ?> <br>
            </div>
        </div>
    </div>
</body>

</html>
```

## Output

**Soal 3**

Burung sedang makan
Aransha sedang makan

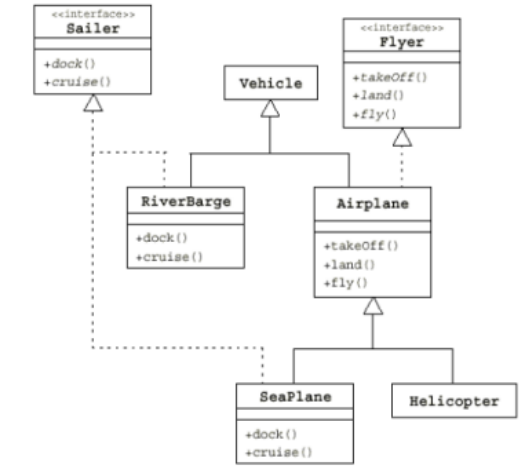
**Maksimal muatan Batik Air 25000 kg**
Batik Air menambah muatan sebesar 6000 kg
Batik Air menambah muatan sebesar 2000 kg
Batik Air menambah muatan sebesar 7000 kg
Batik Air menambah muatan sebesar 5000 kg
Batik Air lepas landas
Batik Air dalam perjalanan
Batik Air mendarat
Jadi, Butuh Bahan Bakar sebanyak 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 abtract 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

```php
<!--- Nama    : Rifqi yudo dewantoro
      NIM     : 21091397074
      Kelas   : 2021 b--->

<?php

interface Flyer {
    public function takeOff();
    public function land();
    public function fly();
}

interface Sailer {
    public function dock();
    public function cruise();
}
```

### PHP

C: > xampp > htdocs > prak 6 > No4 D.php

```php
<!--- Nama     : Rifqi yudo dewantoro
      NIM      : 21091397074
      Kelas    : 2021 B--->

<?php

require_once 'No4 A.php';
require_once 'No4 C.php';

class RiverBarge2 extends Vehicle implements Sailer {
    public function __construct($maxLoad, $name) {
        $this->maxLoad = $maxLoad;
        $this->name = $name;
    }

    public function calcFuelNeeds() {
        $fuel = $this->calcFuelEfficiency();
        $trip = $this->calcTripDistance();

        return ceil($fuel /= $trip);
    }

    public function dock() {
        return $this->name . ' berada di dermaga';
    }

    public function cruise() {
        return $this->name . ' sedang berlayar';
    }
}

class Airplane2 implements Flyer {
    public function takeOff() {
        return $this->name . ' lepas landas';
    }

    public function land() {
        return $this->name . ' mendarat';
    }

    public function fly() {
        return $this->name . ' dalam perjalanan';
    }
}

class Helicopter extends Vehicle {
    public function __construct($maxLoad, $name) {
        $this->maxLoad = $maxLoad;
        $this->name = $name;
    }

    public function calcFuelNeeds() {
        $fuel = $this->calcFuelEfficiency();
        $trip = $this->calcTripDistance();

        return ceil($fuel /= $trip);
    }
    public function takeOff() {
        return $this->name . ' lepas landas';
    }

    public function land() {
        return $this->name . ' mendarat';
    }
}
```

```php
class Airplane2 implements Flyer {
    public function takeOff() {
        return 'Pesawat lepas landas';
    }
    public function land() {
        return 'Pesawat mendarat';
    }
    public function fly() {
        return 'Pesawat dalam perjalanan';
    }
}

class SeaPlane extends Vehicle implements Sailer {
    public function __construct($maxLoad, $name) {
        $this->maxLoad = $maxLoad;
        $this->name = $name;
    }

    public function calcFuelNeeds() {
        $fuel = $this->calcFuelEfficiency();
        $trip = $this->calcTripDistance();

        return ceil($fuel /= $trip);
    }

    public function dock() {
        return $this->name . ' berada di dermaga';
    }

    public function cruise() {
        return $this->name . ' sedang berlayar';
    }
```

```php
    public function fly() {
        return $this->name . ' dalam perjalanan';
    }
}

$riverBarge2 = new RiverBarge2(30000, 'Atomic');
$seaPlane = new SeaPlane(20000, 'Titanic');
$helicopter = new Helicopter(10000, 'Brocklyn');
```

## Abstract PHP

C: > xampp > htdocs > prak 6 > No4 A.php

```php
<!--- Nama    : Rifqi yudo dewantoro
      NIM     : 21091397074
      Kelas   : 2021 B--->

<?php

abstract class Vehicle {
    private $load = 0;
    protected $maxLoad = 0, $name;

    protected function __construct($maxLoad, $name) {
        $this->$maxLoad = $maxLoad;
        $this->$name = $name;
    }

    public function getLoad() {
        return $this->load;
    }

    public function getMaxLoad() {
        echo 'Maksimal muatan ' . $this->name . ' ' ;
        return $this->maxLoad;
    }

    public function addBox($weight) {
        if ($this->load >= $this->maxLoad) {
            echo "$this->name menambah muatan sebesar $weight <br>";
            echo 'Muatan telah penuh tidak bisa menambah lagi';
        }else {
            $this->load += $weight;
            echo "$this->name menambah muatan sebesar $weight";
        }
    }

    abstract public function calcFuelNeeds();

    protected function calcFuelEfficiency() {
        $range = 50000000;
        $range /= $this->load;
        return $range;
    }

    protected function calcTripDistance() {
        return 500;
    }
}
```

## Index PHP

C: > xampp > htdocs > prak 6 > No4 B.php

```php
<!--- Nama    : Rifqi yudo dewantoro
      NIM     : 21091397074
      Kelas   : 2021 B--->

      <?php
    require_once 'No4 D.php';
?>

<!DOCTYPE html>
<html lang="en">

<head>
    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="styles
    integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin=

    <title>PBO - Praktikum 6</title>
</head>
<body>
        <h2 class="text-center">PBO - Praktikum 6</h2>
        <div class="container">
  <div class="row">
  <h4 class="text-center"><strong>Soal 4</strong></h4>
  <br><br>
    <div class="col-4 mx-auto border p-2 mt-2">
                <b><?= $riverBarge2->getMaxLoad() . ' kg'; ?> <br></b>
                <?= $riverBarge2->addBox(12000) . ' kg'; ?> <br>
                <?= $riverBarge2->addBox(14000) . ' kg'; ?> <br>
                <?= $riverBarge2->addBox(1000) . ' kg'; ?> <br>
                <?= $riverBarge2->addBox(3000) . ' kg'; ?> <br>
                <?= $riverBarge2->dock(); ?> <br>
                <?= $riverBarge2->cruise(); ?> <br>
```

```php
                <?php
                    echo "Jadi, Butuh Bahan Bakar sebanyak " . $riverBarge2->calcFuelNeeds() . ' Liter'. '<br>';
                ?>
    </div>
    <div class="col-4 mx-auto border p-2 mt-2">
                <b><?= $seaPlane->getMaxLoad() . ' kg'; ?> <br></b>
                <?= $seaPlane->addBox(12000) . ' kg'; ?> <br>
                <?= $seaPlane->addBox(8000) . ' kg'; ?> <br>
                <?= $seaPlane->dock(); ?> <br>
                <?= $seaPlane->cruise(); ?> <br>
                <?= $seaPlane->takeOff(); ?> <br>
                <?= $seaPlane->fly(); ?> <br>
                <?= $seaPlane->land(); ?> <br>
                <?php
                    echo "Jadi, Butuh Bahan Bakar sebanyak " . $seaPlane->calcFuelNeeds() . ' Liter'. '<br>';
                ?>
    </div>
    <div class="col mx-auto border p-2 mt-2">
                <b><?= $helicopter->getMaxLoad() . ' kg'; ?> <br></b>
                <?= $helicopter->addBox(8000) . ' kg'; ?> <br>
                <?= $helicopter->addBox(2000) . ' kg'; ?> <br>
                <?= $helicopter->takeOff(); ?> <br>
                <?= $helicopter->fly(); ?> <br>
                <?= $helicopter->land(); ?> <br>
                <?php
                    echo "Jadi, Butuh Bahan Bakar sebanyak " . $helicopter->calcFuelNeeds() . ' Liter'. '<br>';
                ?>
    </div>
  </div>
</div>
</body>
</html>
```

## Output

PBO - Praktikum 6

**Soal 4**

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

## Analisa

Implementasi polymhorphism dengan interface dan abstact class detinjukkan 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