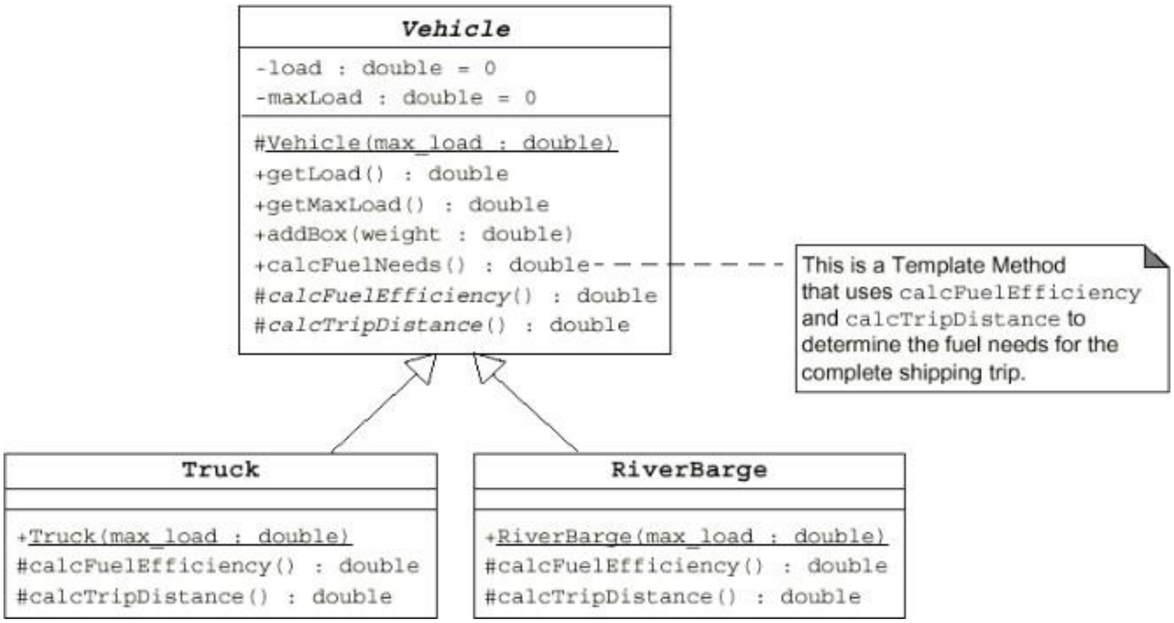
LAPORAN PRAKTIKUM 6 MATAKULIAH PEMROGRAMAN BERORIENTASI OBYEK

Oleh:
Revina Aurigha Firdaus
21091397003

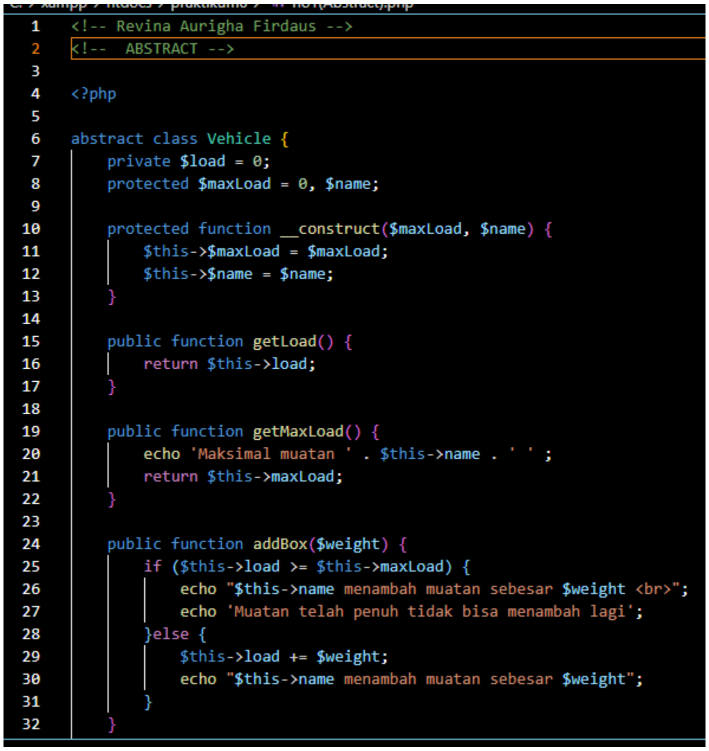
D4 MANAJEMEN INFORMATIKA
UNIVERSITAS NEGERI SURABAYA
2021/2022

1. Buat program berdasarkan UML berikut

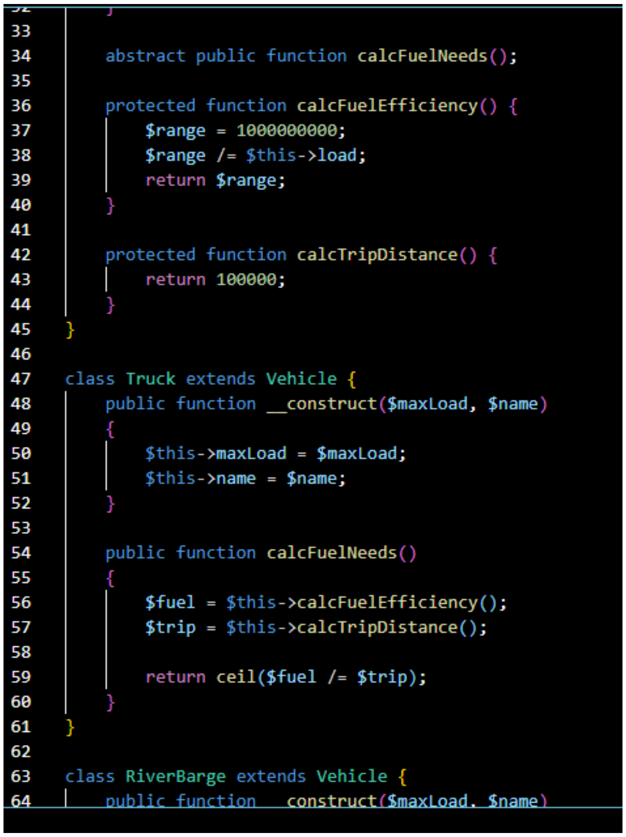

Vehicle
-load : double = 0
-maxLoad : double = 0
#Vehicle(max_load : double)
+getLoad() : double
+getMaxLoad() : double
+addBox(weight : double)
+calcFuelNeeds() : double
#calcFuelEfficiency() : double
#calcTripDistance() : double
Truck
+Truck(max_load : double)
#calcFuelEfficiency() : double
#calcTripDistance() : double
RiverBarge
+RiverBarge(max_load : double)
#calcFuelEfficiency() : double
#calcTripDistance() : double

This is a Template Method that uses calcFuelEfficiency and calcTripDistance to determine the fuel needs for the complete shipping trip.

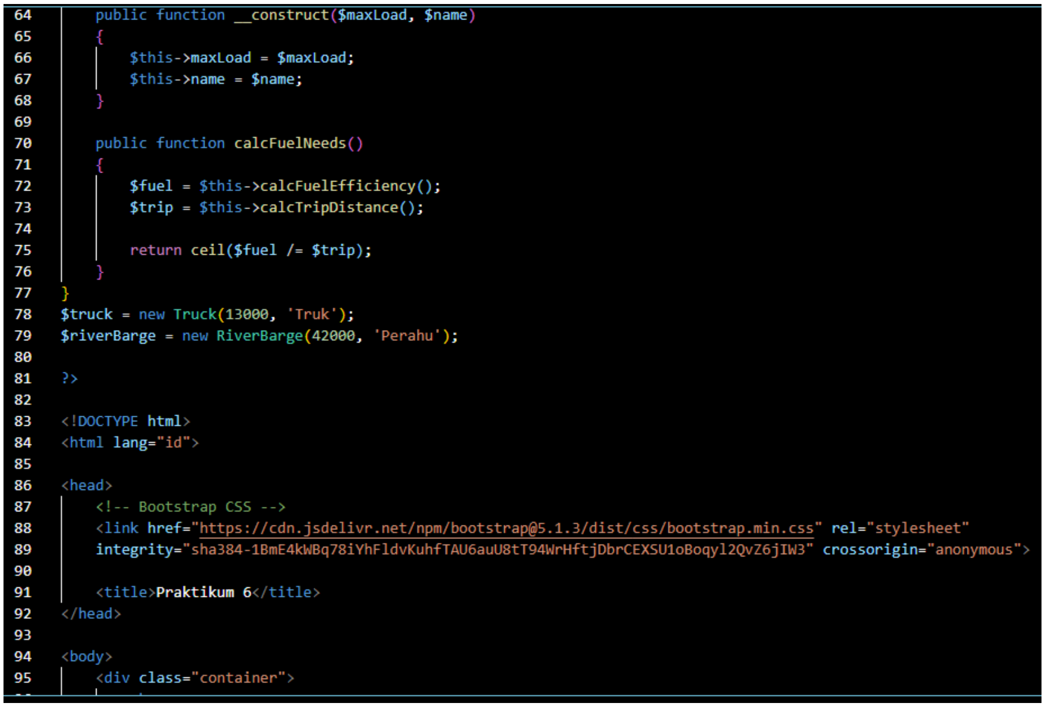
- Source Code



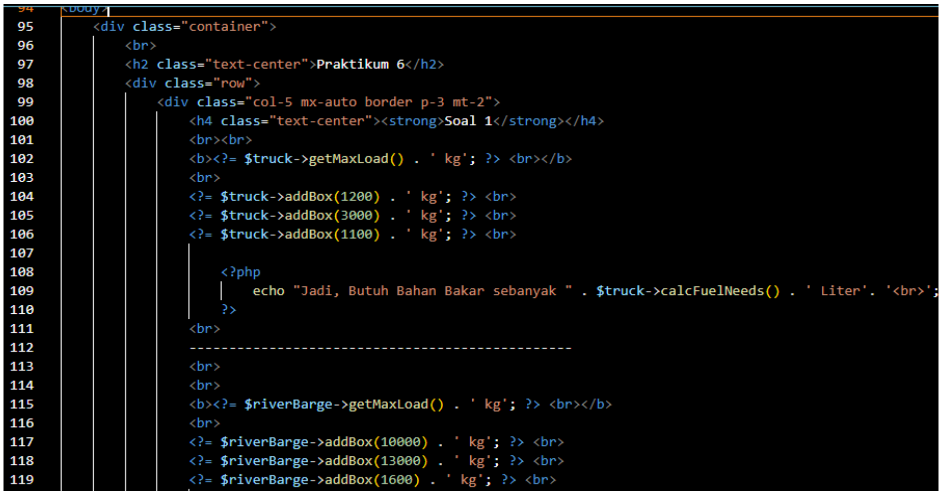
```php
<!-- Revina Aurigha Firdaus -->
<!--  ABSTRACT -->

<?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";
        }
    }
```



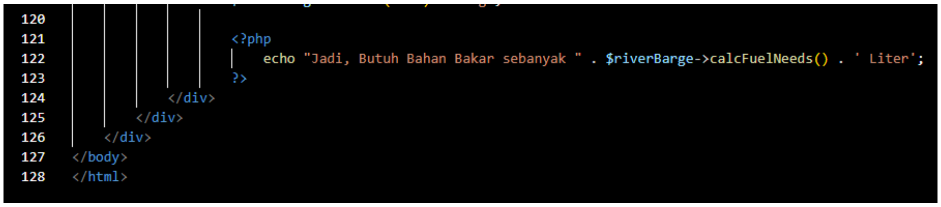
```
    abstract public function calcFuelNeeds();

    protected function calcFuelEfficiency() {
        $range = 1000000000;
        $range /= $this->load;
        return $range;
    }

    protected function calcTripDistance() {
        return 100000;
    }
}

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)
```



```
    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(13000, 'Truk');
$riverBarge = new RiverBarge(42000, 'Perahu');

?>

<!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="stylesheet"
    integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">

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

<body>
    <div class="container">
```

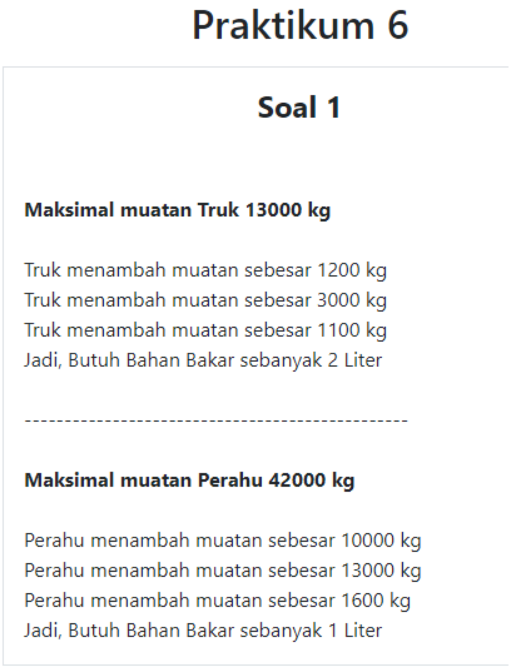


```
<body>
    <div class="container">
        <br>
        <h2 class="text-center">Praktikum 6</h2>
        <div class="row">
            <div class="col-5 mx-auto border p-3 mt-2">
                <h4 class="text-center"><strong>Soal 1</strong></h4>
                <br><br>
                <b><?= $truck->getMaxLoad() . ' kg'; ?> <br></b>
                <br>
                <?= $truck->addBox(1200) . ' kg'; ?> <br>
                <?= $truck->addBox(3000) . ' kg'; ?> <br>
                <?= $truck->addBox(1100) . ' 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(10000) . ' kg'; ?> <br>
                <?= $riverBarge->addBox(13000) . ' kg'; ?> <br>
                <?= $riverBarge->addBox(1600) . ' kg'; ?> <br>
```



```

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

- Output



Praktikum 6

Soal 1

Maksimal muatan Truk 13000 kg

Truk menambah muatan sebesar 1200 kg
Truk menambah muatan sebesar 3000 kg
Truk menambah muatan sebesar 1100 kg
Jadi, Butuh Bahan Bakar sebanyak 2 Liter

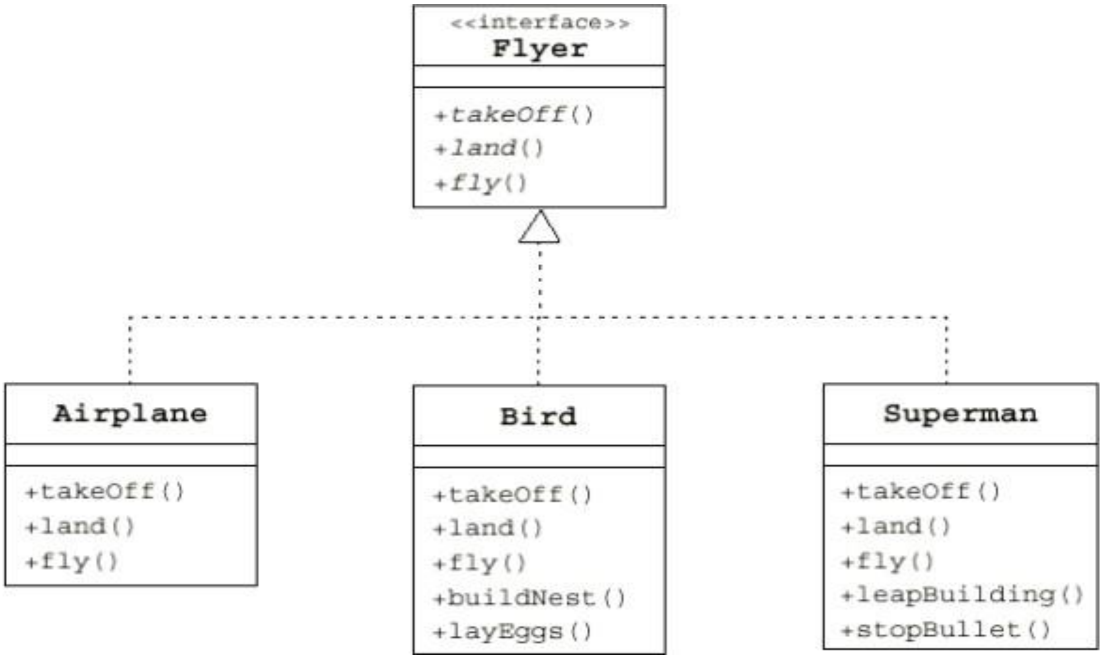
Maksimal muatan Perahu 42000 kg

Perahu menambah muatan sebesar 10000 kg
Perahu menambah muatan sebesar 13000 kg
Perahu menambah muatan sebesar 1600 kg
Jadi, Butuh Bahan Bakar sebanyak 1 Liter

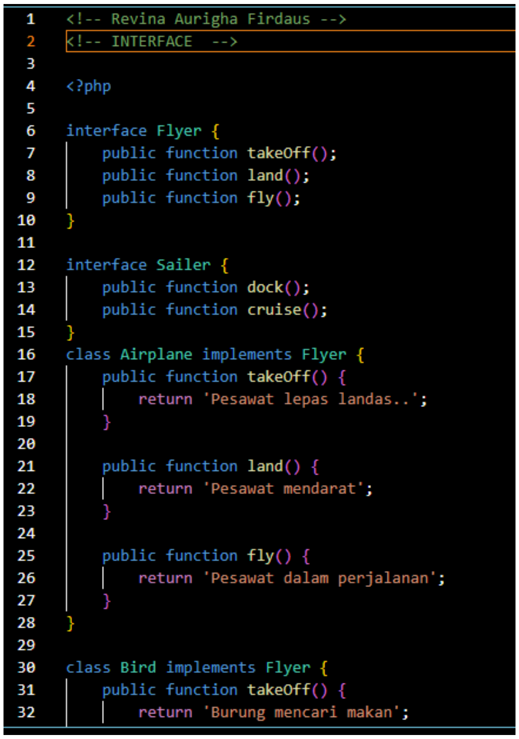
- Analisis

Implementasi dari abstract class pada class Vehicle, method calcFuelNeeds digunakan untuk menghitung bahan bakar yang digunakan. Abstract method diletakkan pada class Vehicle sebagai parent class dan diakses oleh child classnya yaitu class truk, dan class perahu yang akan mengembalikan nilai yang dihasilkan dari pembagian 2 method yaitu calcFuelEfficiency dancalcTripDistance

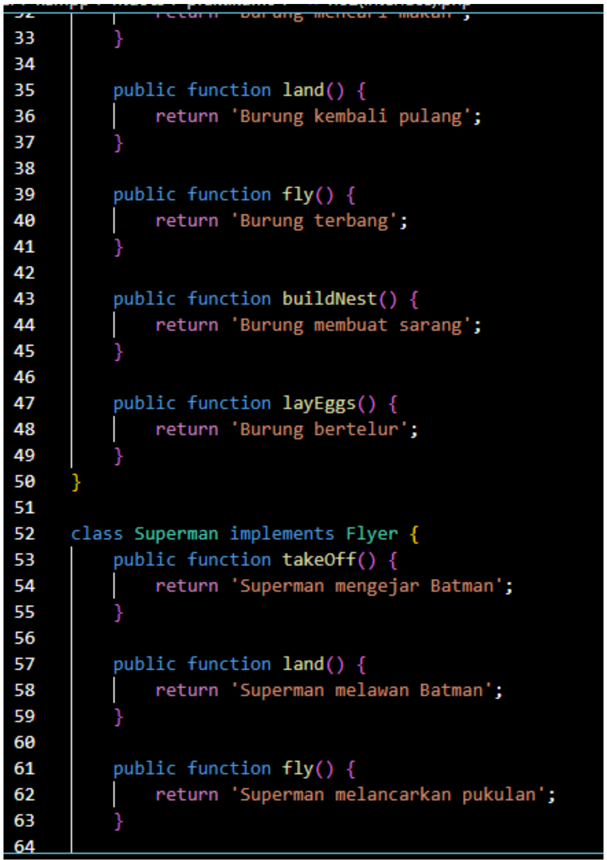
2. Buat program berdasarkan UML berikut



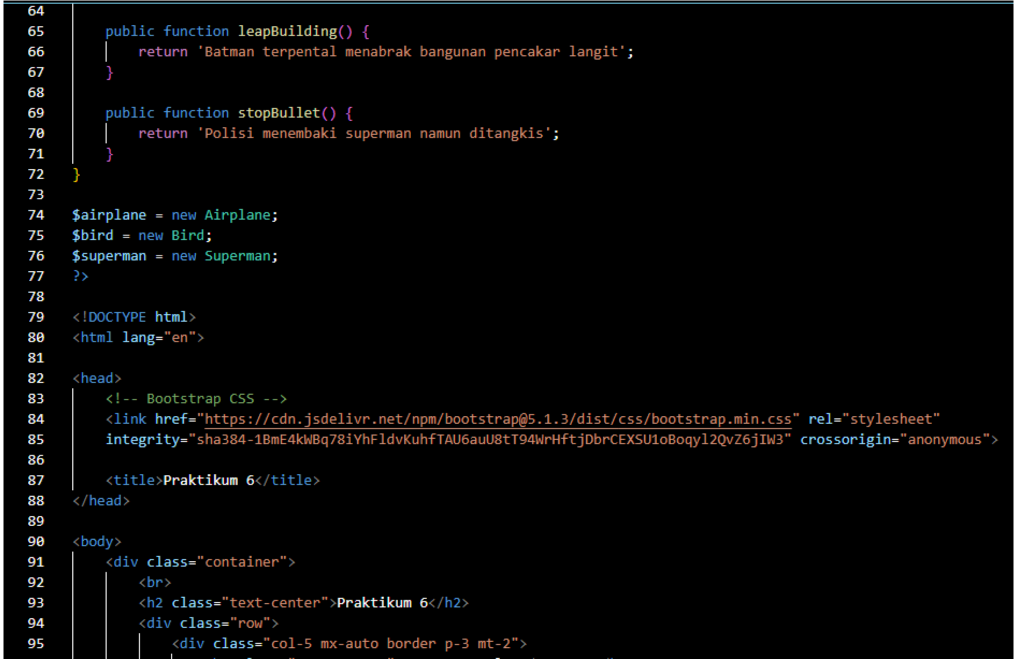
- Source Code



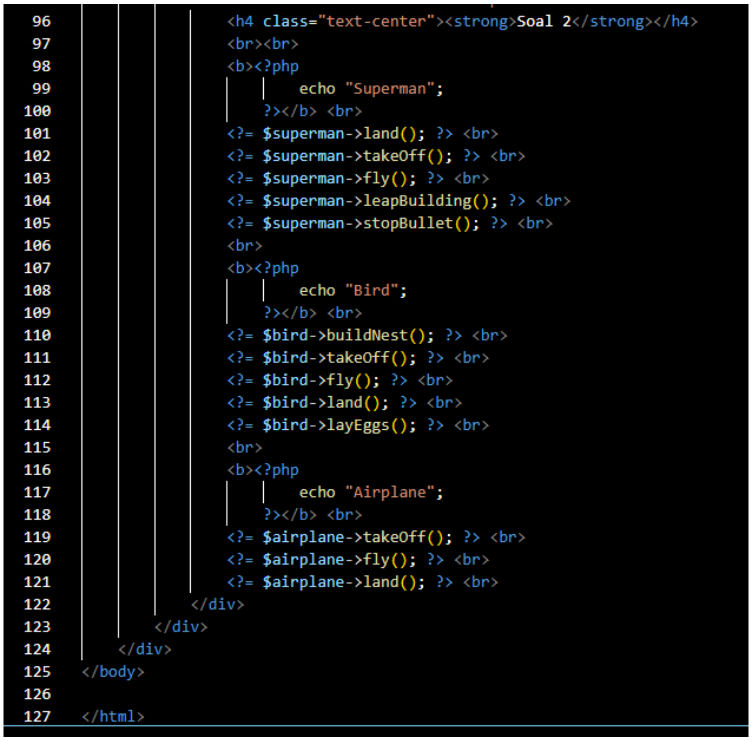
```
<!-- Revina Aurigha Firdaus -->
<!-- INTERFACE  -->

<?php

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

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



```php
        return 'Burung mencari makan';
    }

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

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

    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';
    }

```

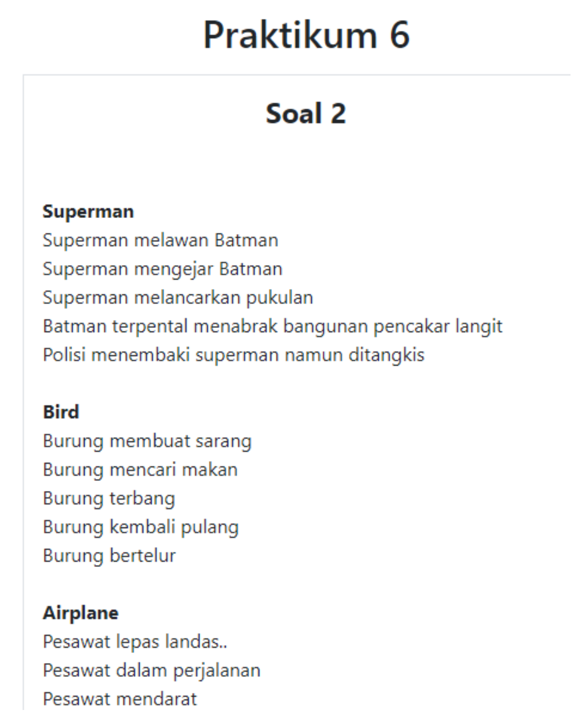


```php

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

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

$airplane = new Airplane;
$bird = new Bird;
$superman = new Superman;
?>

<!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="stylesheet"
    integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">

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

<body>
    <div class="container">
        <br>
        <h2 class="text-center">Praktikum 6</h2>
        <div class="row">
            <div class="col-5 mx-auto border p-3 mt-2">
```



```
                <h4 class="text-center"><strong>Soal 2</strong></h4>
                <br><br>
                <b><?php
                        echo "Superman";
                    ?></b> <br>
                <?= $superman->land(); ?> <br>
                <?= $superman->takeOff(); ?> <br>
                <?= $superman->fly(); ?> <br>
                <?= $superman->leapBuilding(); ?> <br>
                <?= $superman->stopBullet(); ?> <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

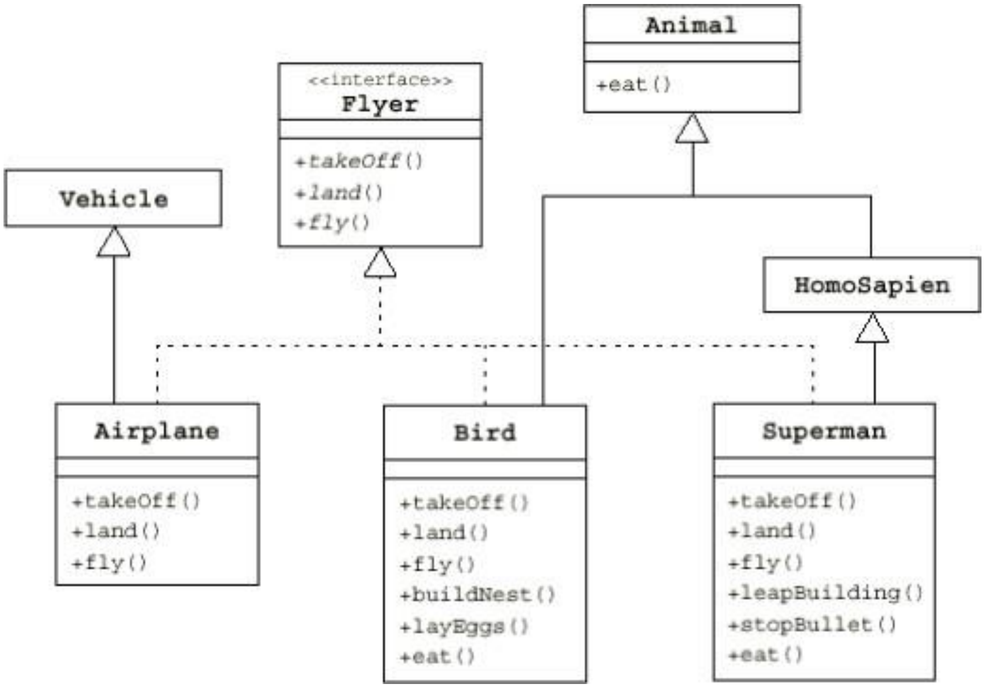


```
Praktikum 6

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
```

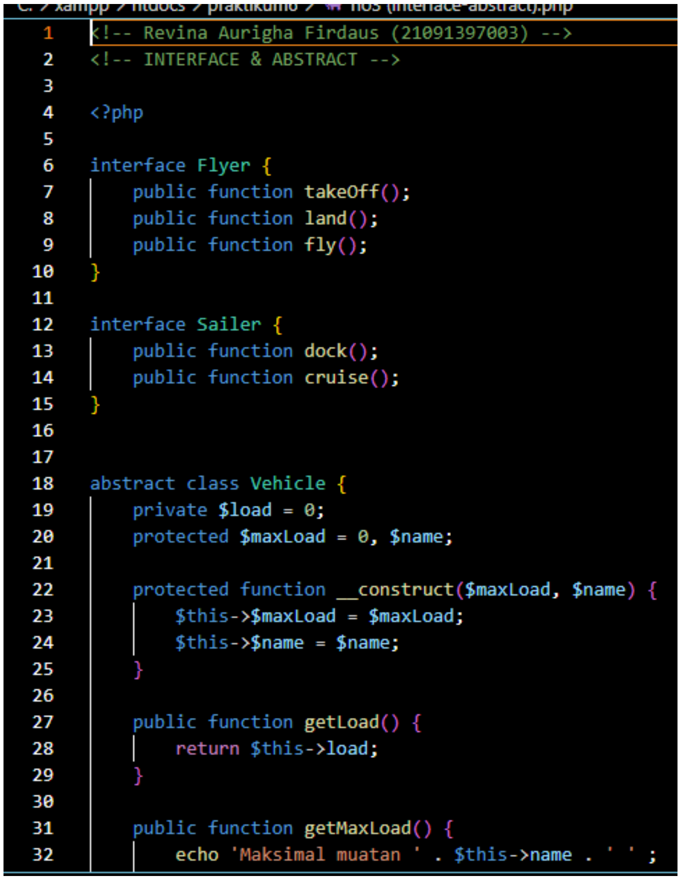
- Analisis

  Implementasi Polymhorpism dengan penggunaan Interface Flyer sehingga semua class dari superman, bird, airplane harus memiliki method takeoff, land, dan fly.

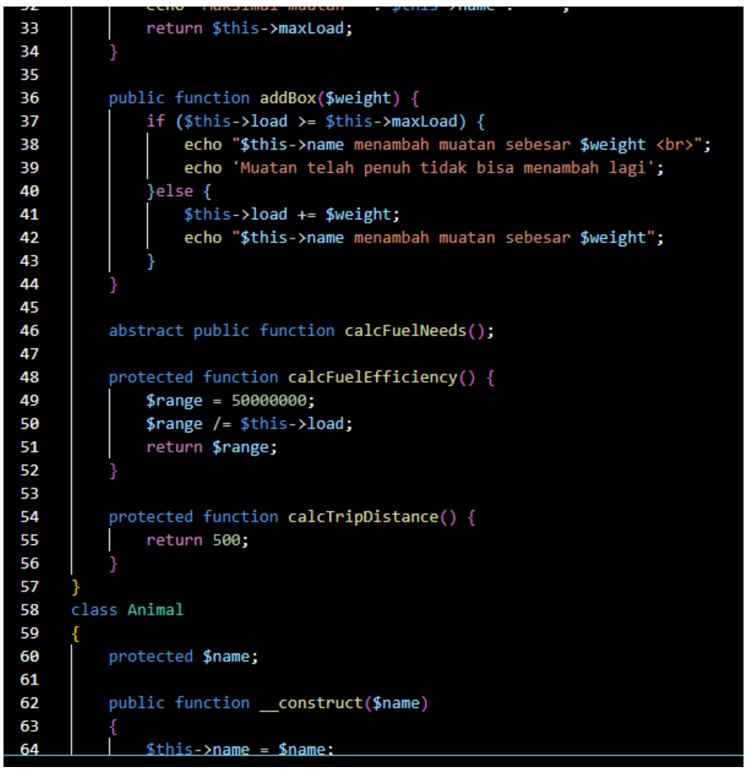
3. Buat program berdasarkan UML berikut



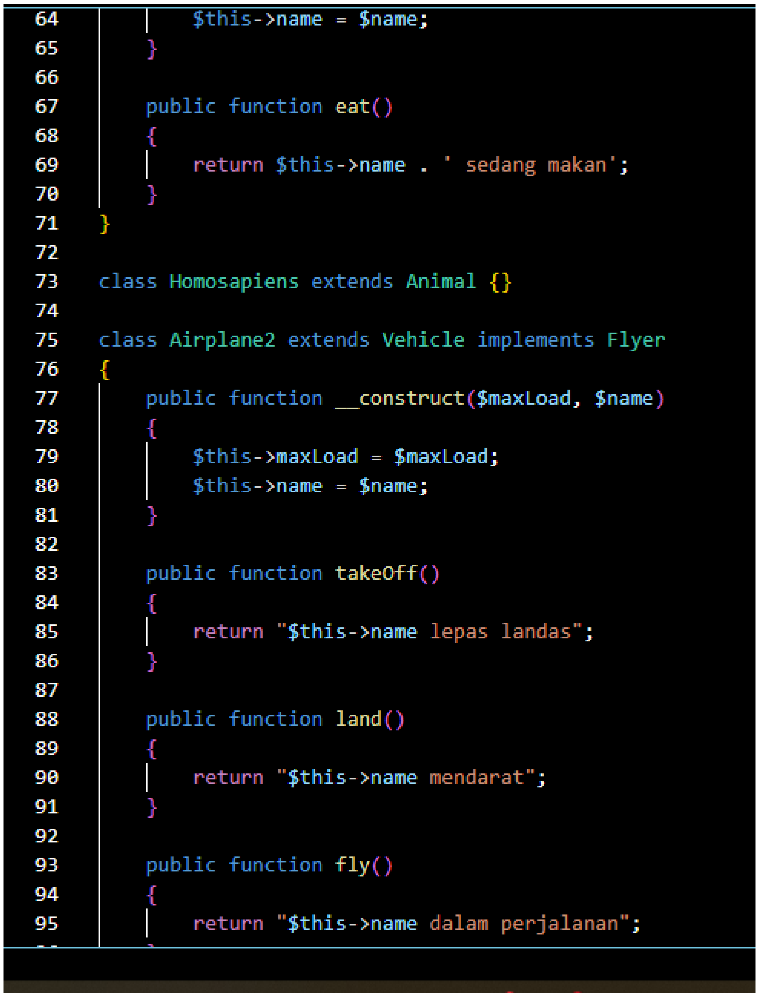
- Source Code


C: > xampp > htdocs > praktikum6 > no3 (interface-abstract).php

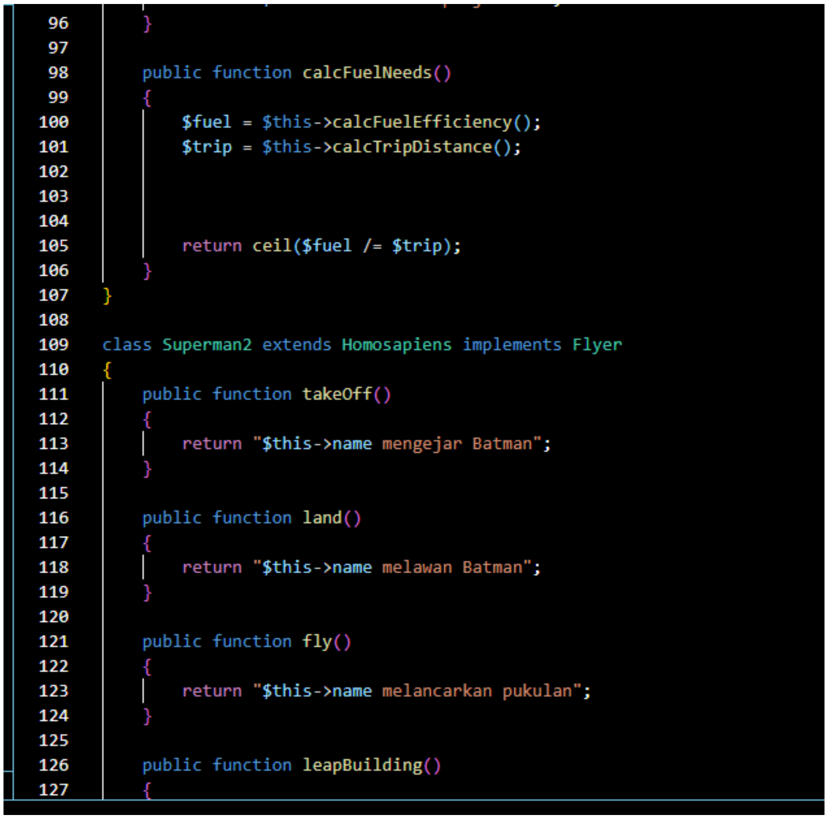
```
<!-- Revina Aurigha Firdaus (21091397003) -->
<!-- INTERFACE & ABSTRACT -->

<?php

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

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


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 . ' ' ;
```



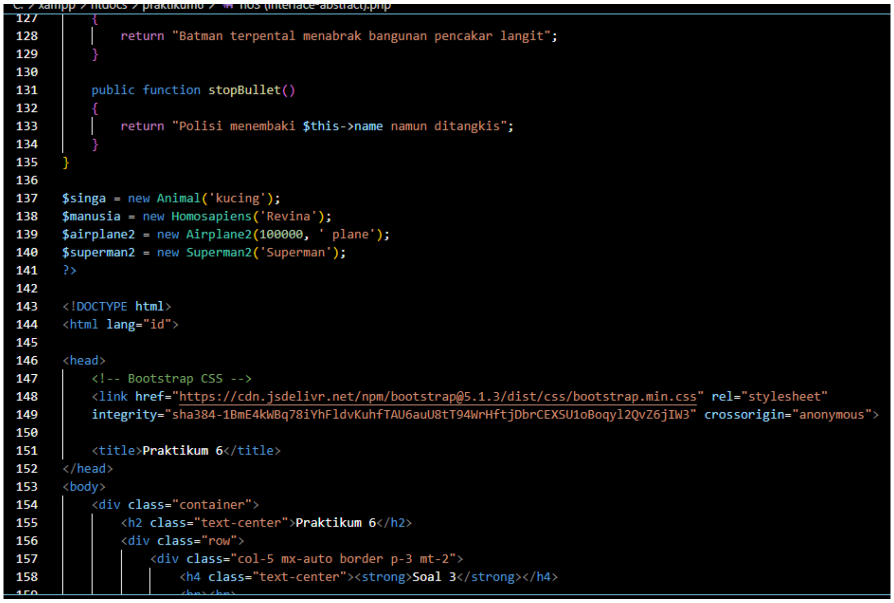
```php
        echo 'Maksimal muatan ' . $this->name . ' ';
        return $this->maxLoad;
    }

    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;
    }
}
class Animal
{
    protected $name;

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



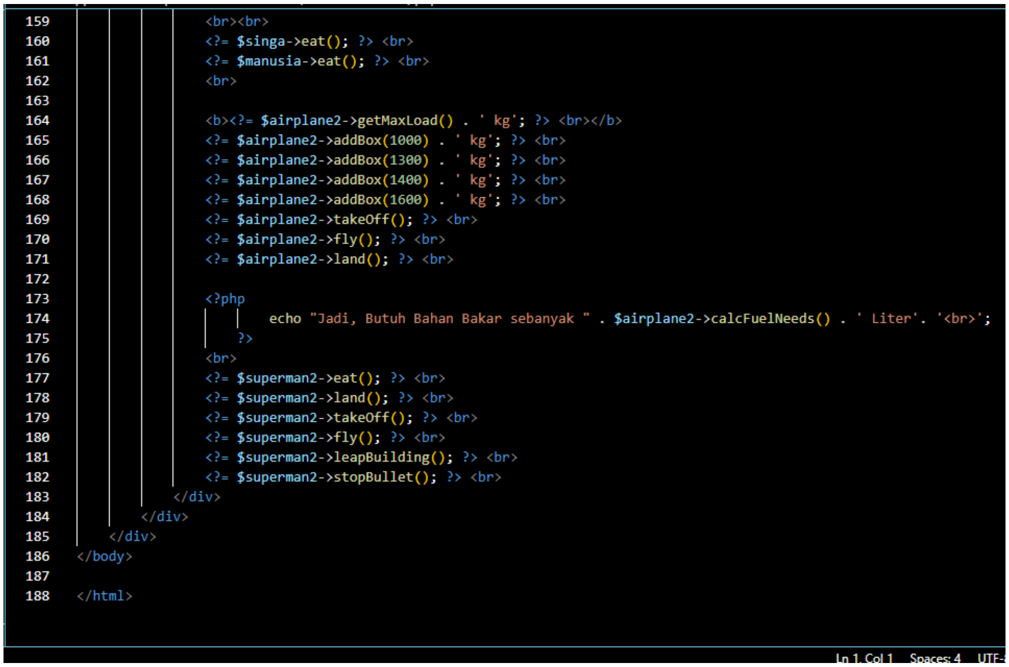
```php
        $this->name = $name;
    }

    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;
    }

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

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

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



```php
    }

    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";
    }

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

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

    public function leapBuilding()
    {
```

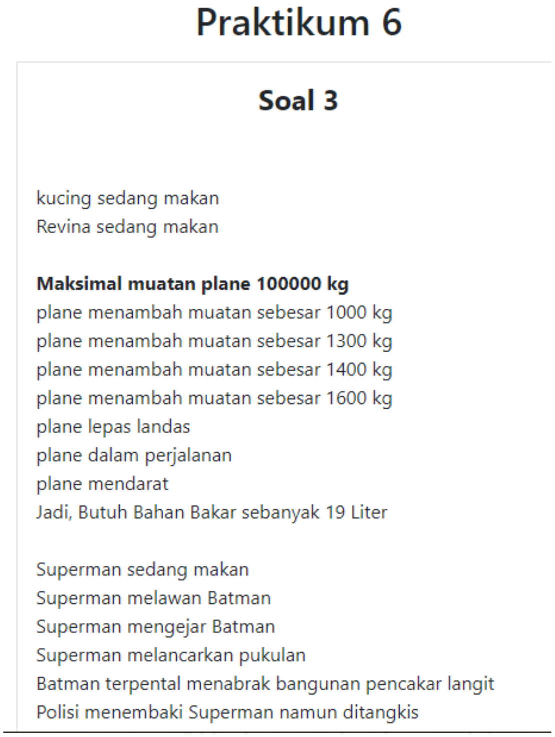


```php
        return "Batman terpental menabrak bangunan pencakar langit";
    }

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

$singa = new Animal('kucing');
$manusia = new Homosapiens('Revina');
$airplane2 = new Airplane2(100000, ' plane');
$superman2 = new Superman2('Superman');
?>

<!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="stylesheet"
    integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">

    <title>Praktikum 6</title>
</head>
<body>
    <div class="container">
        <h2 class="text-center">Praktikum 6</h2>
        <div class="row">
            <div class="col-5 mx-auto border p-3 mt-2">
                <h4 class="text-center"><strong>Soal 3</strong></h4>
```



```
                <br><br>
                <?= $singa->eat(); ?> <br>
                <?= $manusia->eat(); ?> <br>
                <br>

                <b><?= $airplane2->getMaxLoad() . ' kg'; ?> <br></b>
                <?= $airplane2->addBox(1000) . ' kg'; ?> <br>
                <?= $airplane2->addBox(1300) . ' kg'; ?> <br>
                <?= $airplane2->addBox(1400) . ' kg'; ?> <br>
                <?= $airplane2->addBox(1600) . ' kg'; ?> <br>
                <?= $airplane2->takeOff(); ?> <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

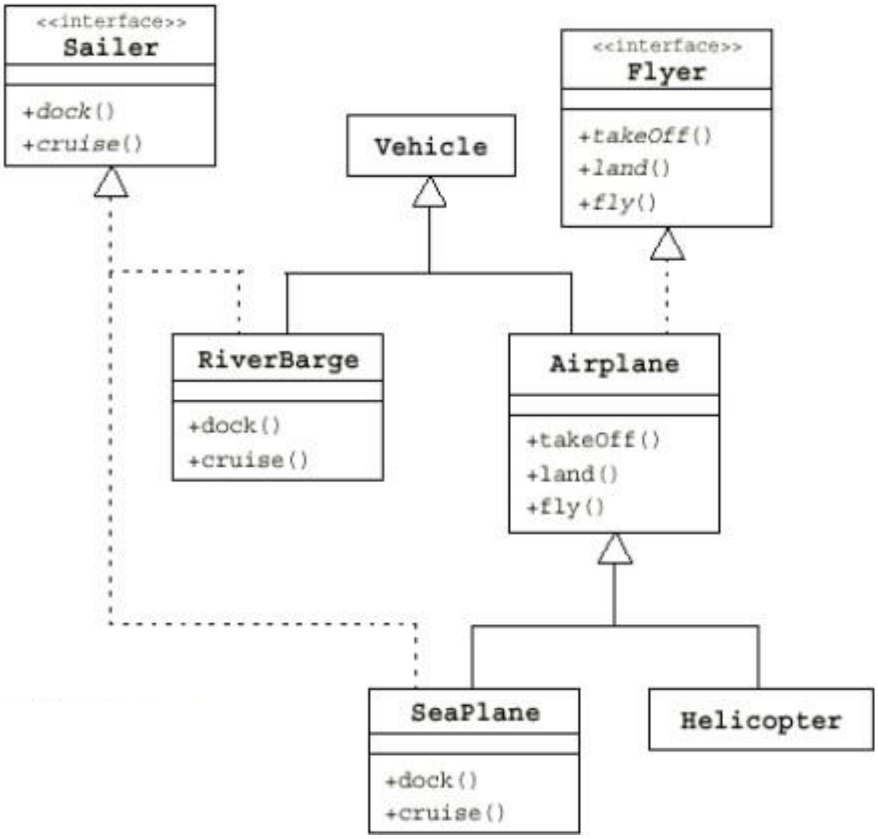


```
Praktikum 6

Soal 3

kucing sedang makan
Revina sedang makan

Maksimal muatan plane 100000 kg
plane menambah muatan sebesar 1000 kg
plane menambah muatan sebesar 1300 kg
plane menambah muatan sebesar 1400 kg
plane menambah muatan sebesar 1600 kg
plane lepas landas
plane dalam perjalanan
plane mendarat
Jadi, Butuh Bahan Bakar sebanyak 19 Liter

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

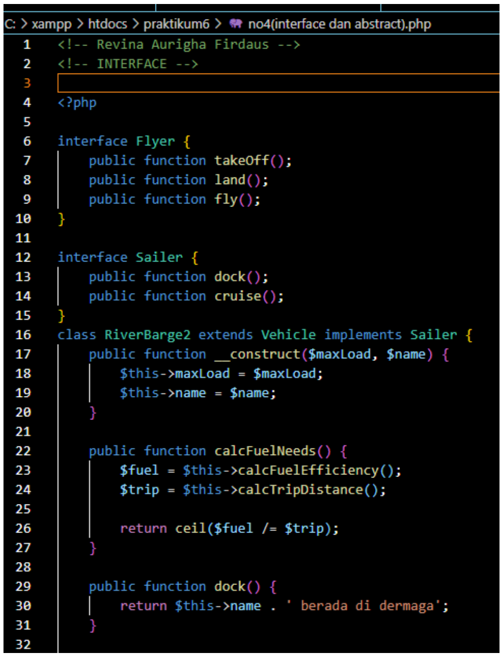
- Analisis

Terdapat interface Flyer dan abtract class Vehicle. Class airplane dari turunan interface Flyer dan turunan dari Vehicle sehingga class Airplane harus memiliki method calcFuelNeeds, takeoff, land, dan fly. Class Bird turunan 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

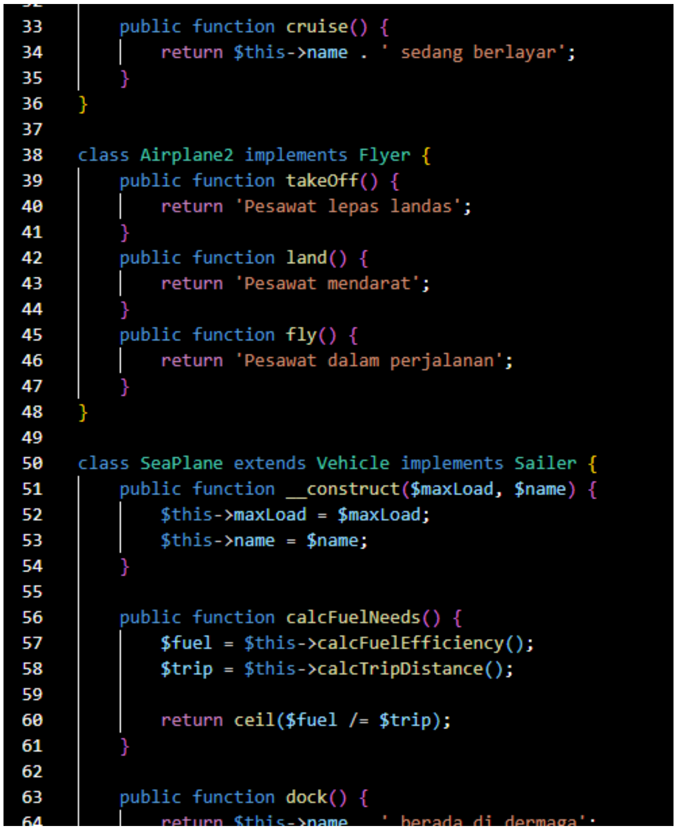


- Source Code

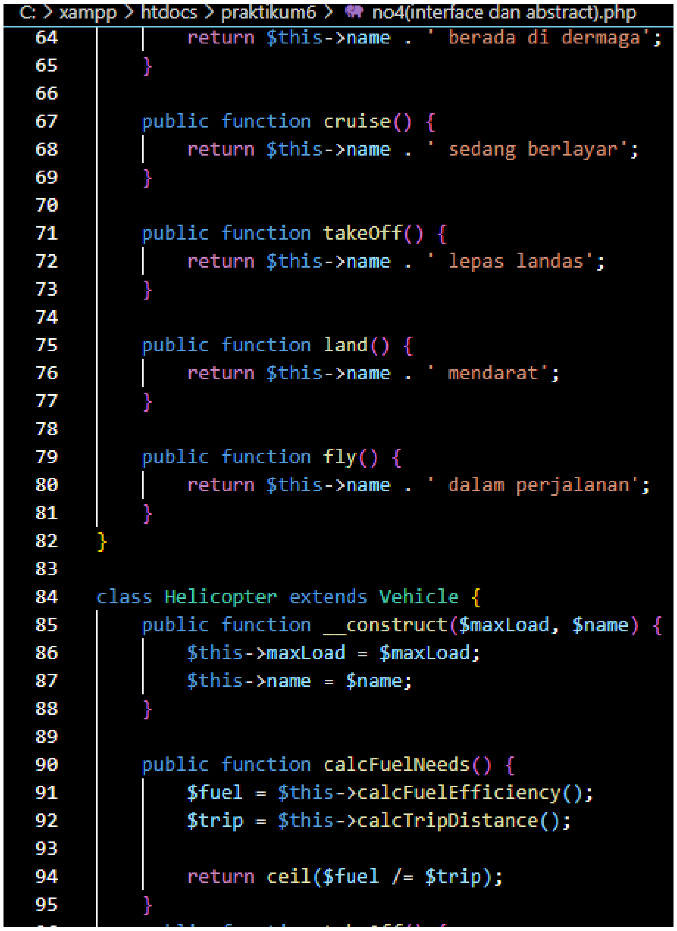


C: > xampp > htdocs > praktikum6 > no4(interface dan abstract).php

```php
<!-- Revina Aurigha Firdaus -->
<!-- INTERFACE -->

<?php

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

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

```

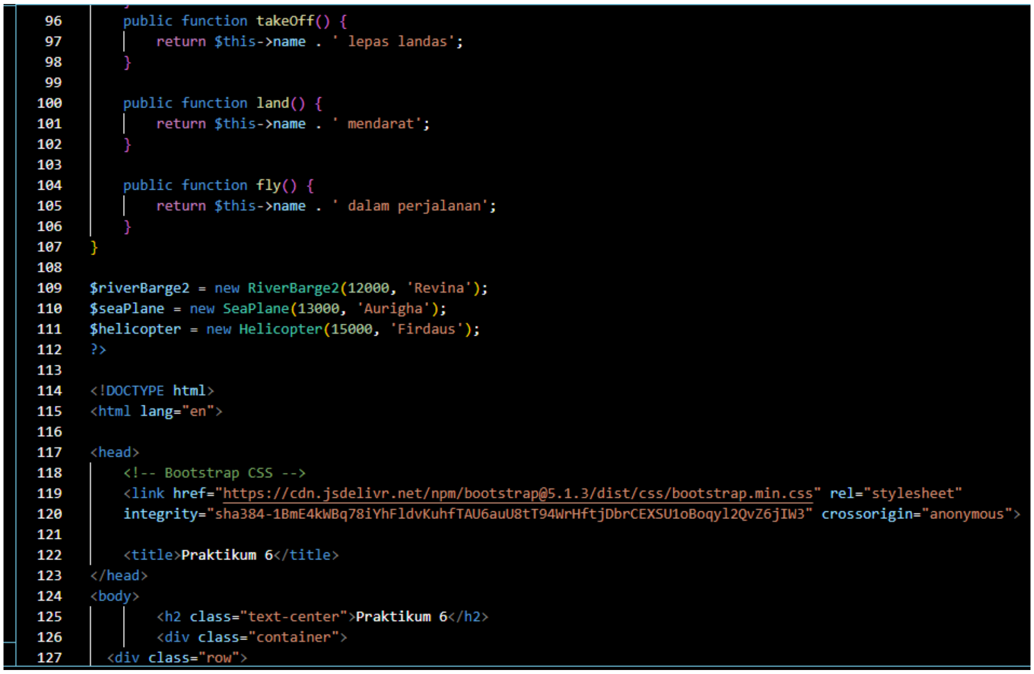


```php
    public function cruise() {
        return $this->name . ' sedang berlayar';
    }
}

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';
```

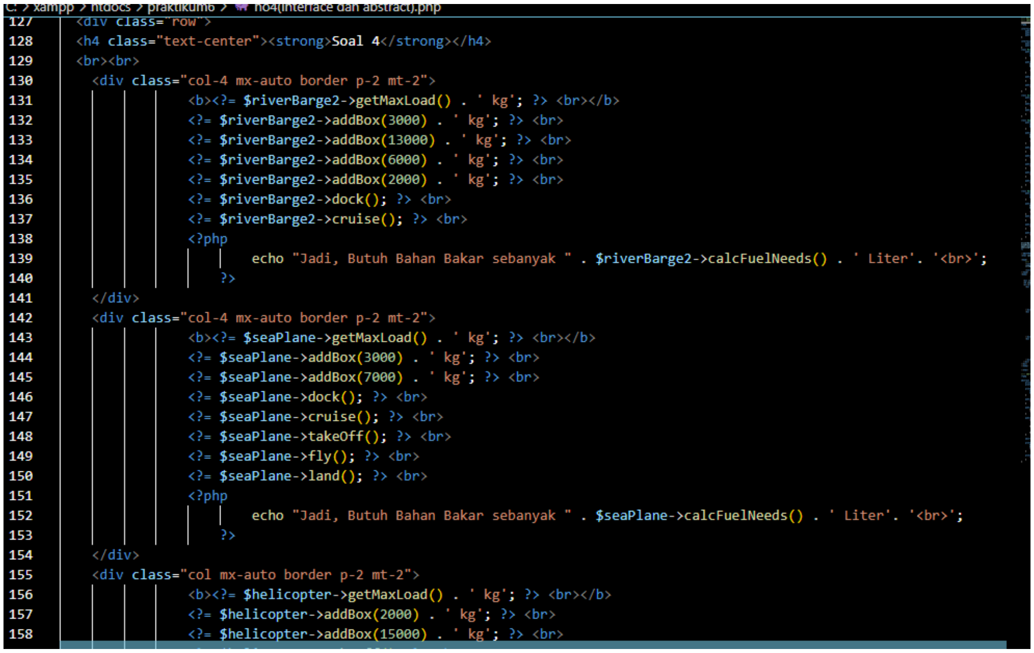


C: > xampp > htdocs > praktikum6 > no4(interface dan abstract).php

```php
        return $this->name . ' berada di dermaga';
    }

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

    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() {
```

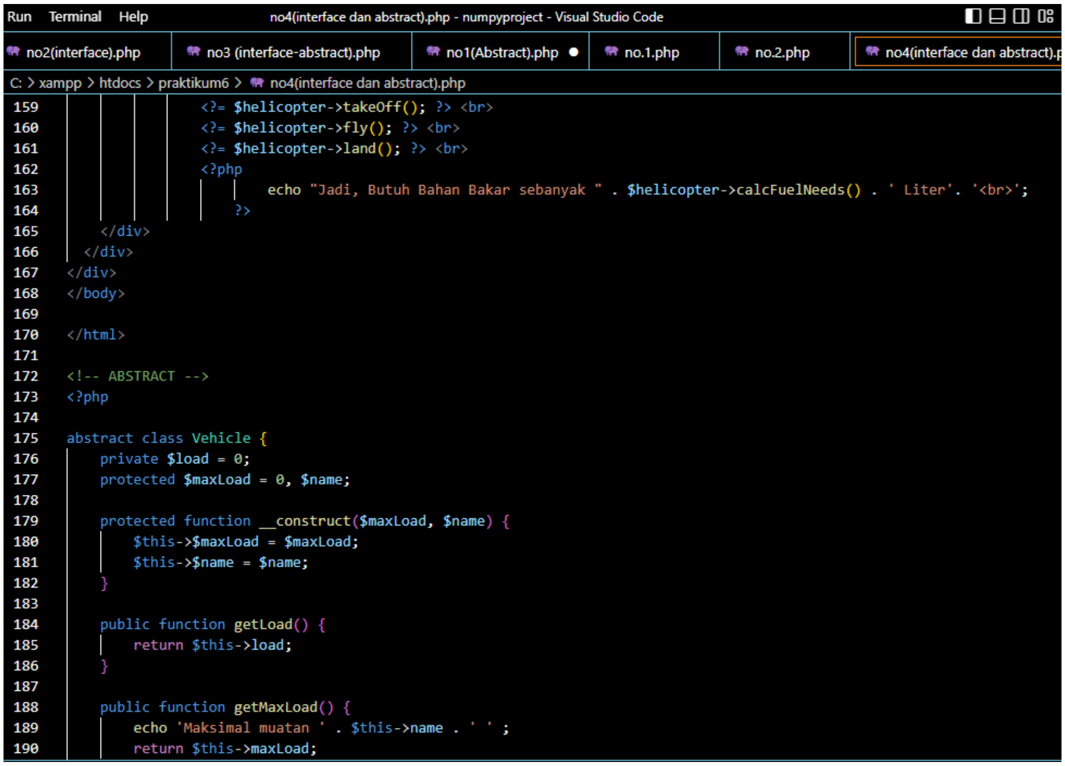


```
    public function takeOff() {
        return $this->name . ' lepas landas';
    }

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

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

$riverBarge2 = new RiverBarge2(12000, 'Revina');
$seaPlane = new SeaPlane(13000, 'Aurigha');
$helicopter = new Helicopter(15000, 'Firdaus');
?>

<!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="stylesheet"
    integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">

    <title>Praktikum 6</title>
</head>
<body>
        <h2 class="text-center">Praktikum 6</h2>
        <div class="container">
  <div class="row">
```

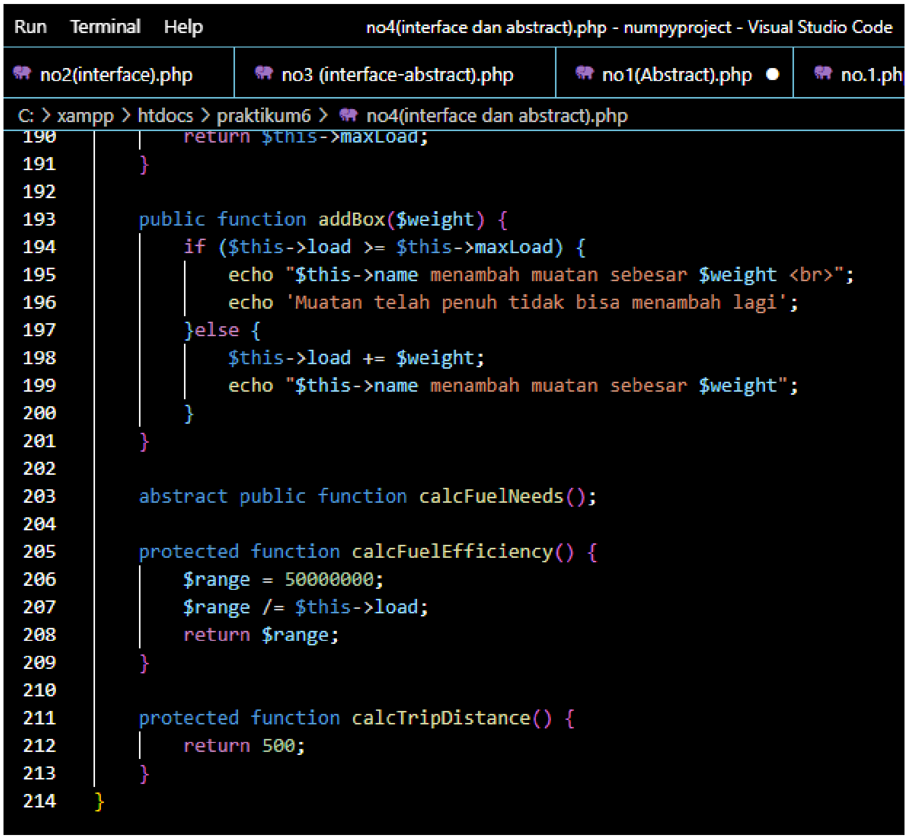


C: > xampp > htdocs > praktikum6 > no4(interface dan abstract).php

```
  <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(3000) . ' kg'; ?> <br>
                <?= $riverBarge2->addBox(13000) . ' kg'; ?> <br>
                <?= $riverBarge2->addBox(6000) . ' kg'; ?> <br>
                <?= $riverBarge2->addBox(2000) . ' kg'; ?> <br>
                <?= $riverBarge2->dock(); ?> <br>
                <?= $riverBarge2->cruise(); ?> <br>
                <?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(3000) . ' kg'; ?> <br>
                <?= $seaPlane->addBox(7000) . ' 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(2000) . ' kg'; ?> <br>
                <?= $helicopter->addBox(15000) . ' kg'; ?> <br>
```

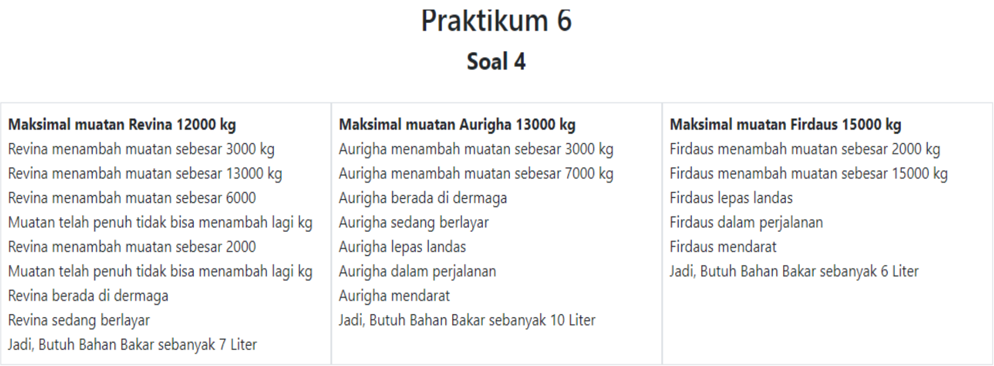

Run
Terminal
Help
no4(interface dan abstract).php - numpyproject - Visual Studio Code
no2(interface).php
no3 (interface-abstract).php
no1(Abstract).php ●
no.1.php
no.2.php
no4(interface dan abstract).p
C: > xampp > htdocs > praktikum6 > no4(interface dan abstract).php

```
                    <?= $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>

    <!-- ABSTRACT -->
    <?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;
```


Run
Terminal
Help
no4(interface dan abstract).php - numpyproject - Visual Studio Code
no2(interface).php
no3 (interface-abstract).php
no1(Abstract).php ●
no.1.ph
C: > xampp > htdocs > praktikum6 > no4(interface dan abstract).php

```
            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;
        }
    }
```

- Output



```
Praktikum 6
Soal 4

Maksimal muatan Revina 12000 kg
Revina menambah muatan sebesar 3000 kg
Revina menambah muatan sebesar 13000 kg
Revina menambah muatan sebesar 6000
Muatan telah penuh tidak bisa menambah lagi kg
Revina menambah muatan sebesar 2000
Muatan telah penuh tidak bisa menambah lagi kg
Revina berada di dermaga
Revina sedang berlayar
Jadi, Butuh Bahan Bakar sebanyak 7 Liter

Maksimal muatan Aurigha 13000 kg
Aurigha menambah muatan sebesar 3000 kg
Aurigha menambah muatan sebesar 7000 kg
Aurigha berada di dermaga
Aurigha sedang berlayar
Aurigha lepas landas
Aurigha dalam perjalanan
Aurigha mendarat
Jadi, Butuh Bahan Bakar sebanyak 10 Liter

Maksimal muatan Firdaus 15000 kg
Firdaus menambah muatan sebesar 2000 kg
Firdaus menambah muatan sebesar 15000 kg
Firdaus lepas landas
Firdaus dalam perjalanan
Firdaus mendarat
Jadi, Butuh Bahan Bakar sebanyak 6 Liter
```

- Analisis

  Implementasi polymhorphism dengan interface dan abstact class yaitu pada class Plane yang turuna dari interface Sailer, class Airplane yang turunan Flyer dan child dari Vehicle sehingga class SeaPlane memiliki method dock, cruise, takeoff, land, fly, dan calcFuelNeeds.