LAPORAN PRAKTIKUM 6

Oleh :

Aditya Putra Pratama

21091397043

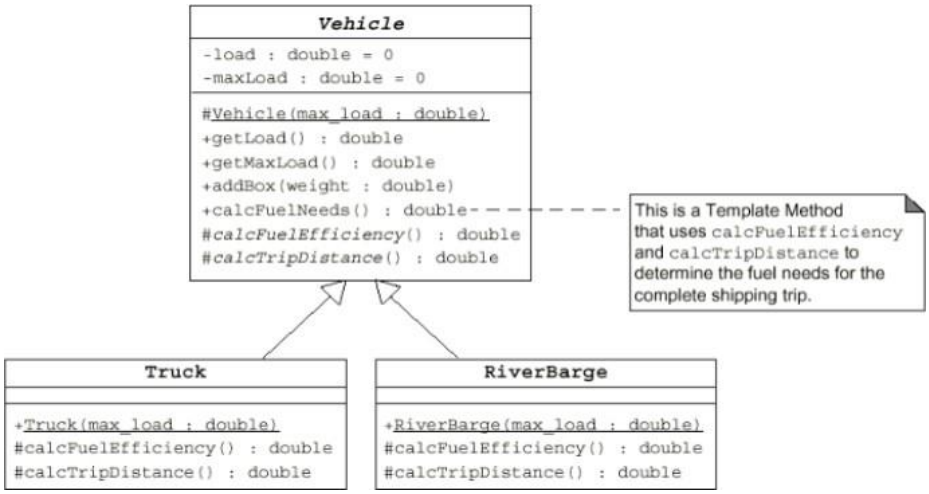
D4 MANAJEMEN INFORMATIKA

FAKULTAS VOKASI

UNIVERSITAS NEGERI SURABAYA

TAHUN 2022

1. Buat program berdasarkan UML berikut



- **Source Code**

1_index.php | 1a.php 6 | 1b.php

@UNESA > @TUGAS KULIAH UNESA > SEMESTER 3 > PEMROGRAMAN BERORIENTASI OBJEK > Praktikum_6 > Nomor 1 > 1_index.php > ...

```php
<?php
require_once "1a.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="stylesheet" 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>Nomor 1</strong></h4>
                <br><br>
                <b><?=$truck->getMaxLoad() . " kg" ?> <br></b>
                <br>
                <?=$truck->addBox(3000) . " kg" ?> <br>
                <?=$truck->addBox(1000) . " kg" ?> <br>
                <?=$truck->addBox(6000) . " kg" ?> <br>

                    <?php echo "Jadi, membutuhkan bahan bakar sebanyak " . $truck->calcFuelNeeds() . " Liter" . "<br>"; ?>
                <br>
                <br>
                <b><?=$riverBarge->getMaxLoad() . " kg" ?> <br></b>
                <br>
                <?=$riverBarge->addBox(1000) . " kg" ?> <br>
                <?=$riverBarge->addBox(6000) . " kg" ?> <br>
                <?=$riverBarge->addBox(8000) . " kg" ?> <br>
```

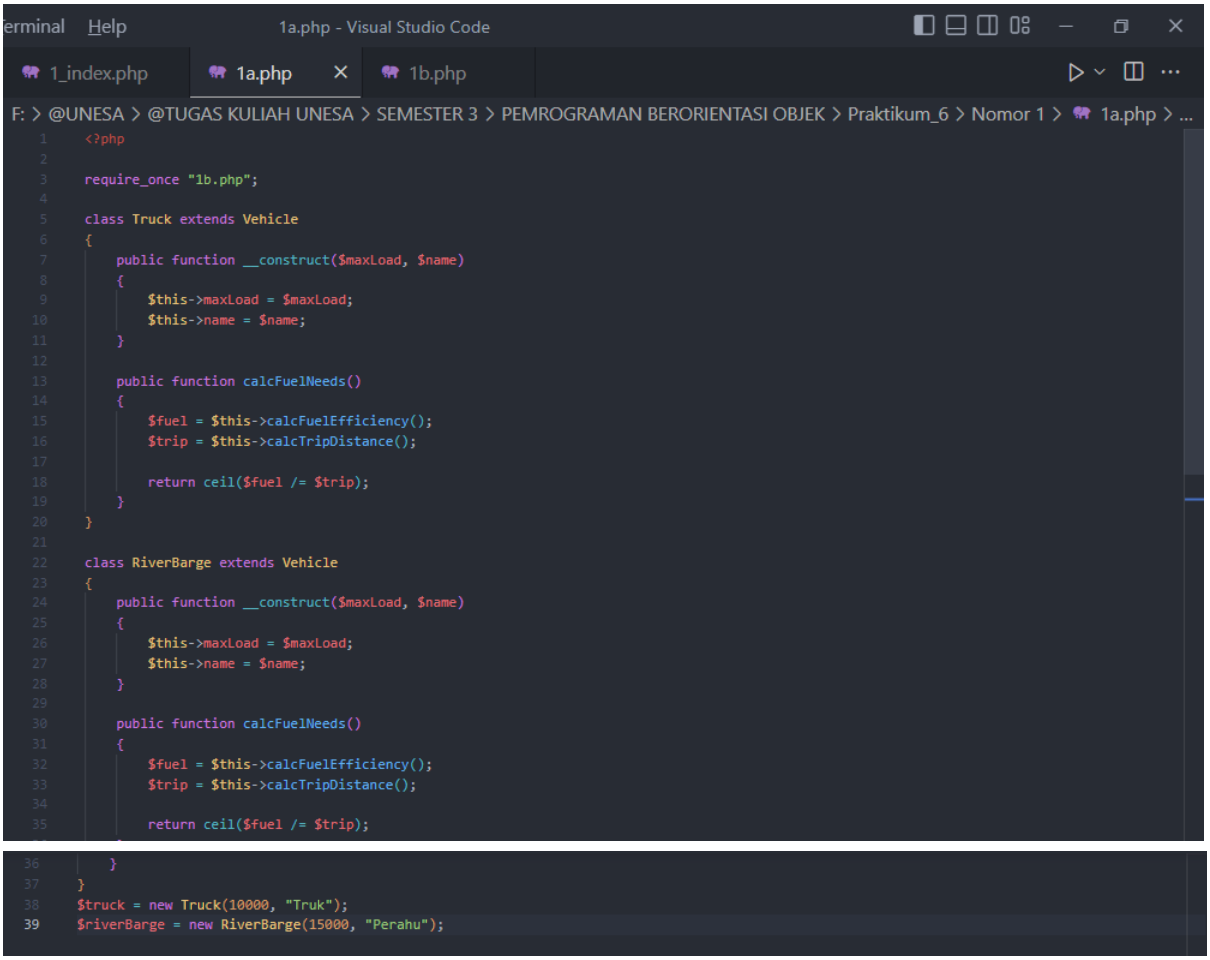
Ln 44, Col 20 Spaces: 4 UTF-8 CRLF PHP

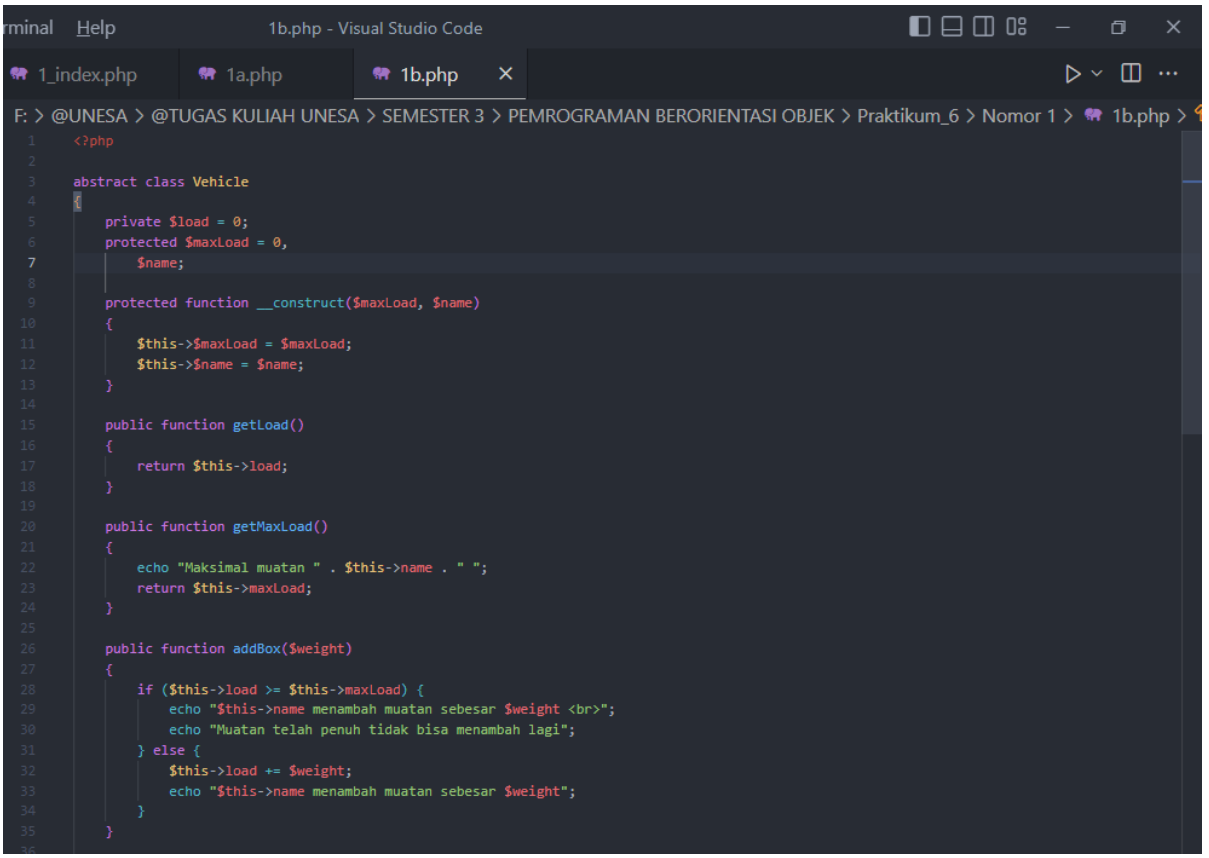
Terminal Help 1_index.php - Visual Studio Code

1_index.php | 1a.php | 1b.php

@UNESA > @TUGAS KULIAH UNESA > SEMESTER 3 > PEMROGRAMAN BERORIENTASI OBJEK > Praktikum_6 > Nomor 1 > 1_index.php > ...

```php
                    <?php echo "Jadi, membutuhkan bahan bakar sebanyak " . $riverBarge->calcFuelNeeds() . " Liter"; ?>
                </div>
            </div>
        </div>
    </body>
    </html>
```


erminal Help
1a.php - Visual Studio Code
1_index.php
1a.php
1b.php
F: > @UNESA > @TUGAS KULIAH UNESA > SEMESTER 3 > PEMROGRAMAN BERORIENTASI OBJEK > Praktikum_6 > Nomor 1 > 1a.php > ...
<?php
require_once "1b.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()
{
$fuel = $this->calcFuelEfficiency();
$trip = $this->calcTripDistance();
return ceil($fuel /= $trip);
}
}
$truck = new Truck(10000, "Truk");
$riverBarge = new RiverBarge(15000, "Perahu");


rminal Help
1b.php - Visual Studio Code
1_index.php
1a.php
1b.php
F: > @UNESA > @TUGAS KULIAH UNESA > SEMESTER 3 > PEMROGRAMAN BERORIENTASI OBJEK > Praktikum_6 > Nomor 1 > 1b.php
<?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
";
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;
    }
}
```

- **Output**

Praktikum 6

Nomor 1

Maksimal muatan Truk 10000 kg

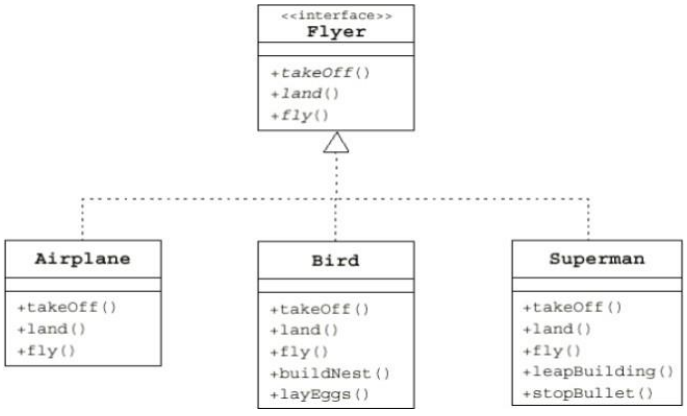
Truk menambah muatan sebesar 3000 kg
Truk menambah muatan sebesar 1000 kg
Truk menambah muatan sebesar 6000 kg
Jadi, membutuhkan bahan bakar sebanyak 10 Liter

Maksimal muatan Perahu 15000 kg

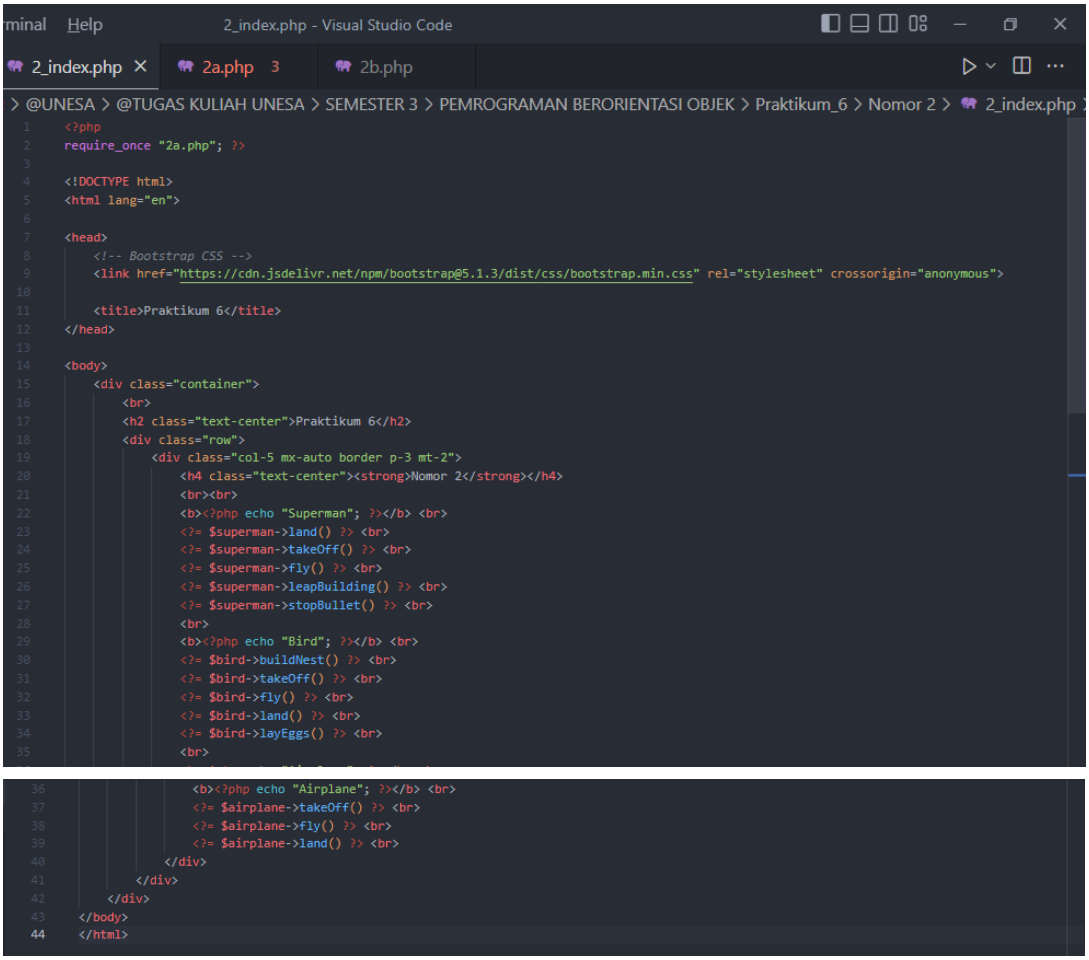
Perahu menambah muatan sebesar 1000 kg
Perahu menambah muatan sebesar 6000 kg
Perahu menambah muatan sebesar 8000 kg
Jadi, membutuhkan bahan bakar sebanyak 7 Liter

- **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 RiverBarge yang akan mengembalikan nilai yang dihasilkan dari pembagian 2 method yaitu calcFuelEfficiency dancalcTripDistance.

2. Buat program berdasarkan UML berikut



- **Source Code**


rminal Help
2_index.php - Visual Studio Code
2_index.php
2a.php 3
2b.php
> @UNESA > @TUGAS KULIAH UNESA > SEMESTER 3 > PEMROGRAMAN BERORIENTASI OBJEK > Praktikum_6 > Nomor 2 > 2_index.php
<?php
require_once "2a.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="stylesheet" crossorigin="anonymous">
<title>Praktikum 6</title>
</head>
<body>

<h2 class="text-center">Praktikum 6</h2>

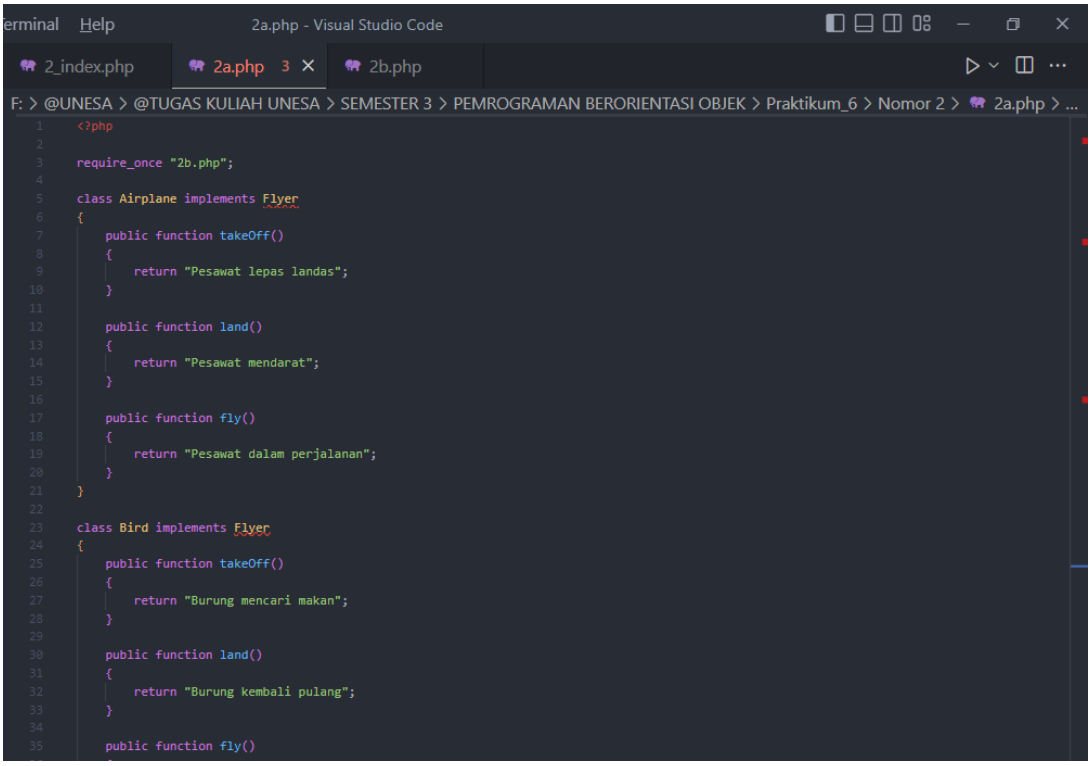
<h4 class="text-center">Nomor 2</h4>

<?php echo "Superman"; ?>
<?= $superman->land() ?>
<?= $superman->takeOff() ?>
<?= $superman->fly() ?>
<?= $superman->leapBuilding() ?>
<?= $superman->stopBullet() ?>

<?php echo "Bird"; ?>
<?= $bird->buildNest() ?>
<?= $bird->takeOff() ?>
<?= $bird->fly() ?>
<?= $bird->land() ?>
<?= $bird->layEggs() ?>

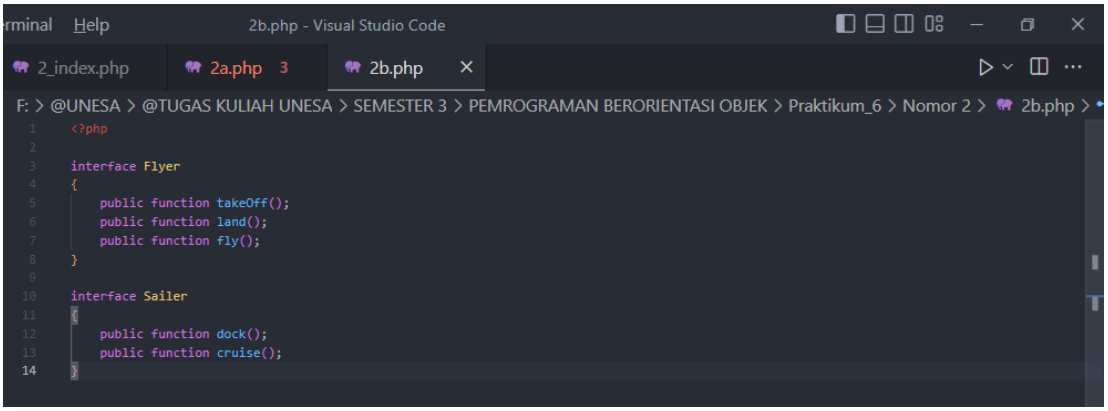
<?php echo "Airplane"; ?>
<?= $airplane->takeOff() ?>
<?= $airplane->fly() ?>
<?= $airplane->land() ?>

</body>
</html>

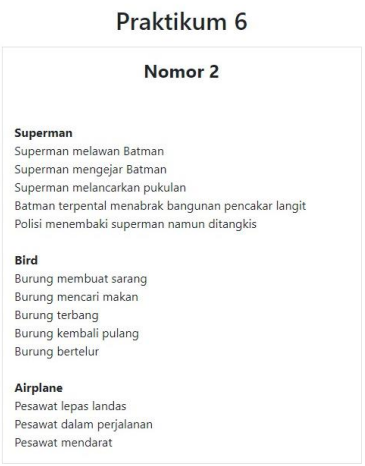

erminal Help
2a.php - Visual Studio Code
2_index.php
2a.php 3
2b.php
F: > @UNESA > @TUGAS KULIAH UNESA > SEMESTER 3 > PEMROGRAMAN BERORIENTASI OBJEK > Praktikum_6 > Nomor 2 > 2a.php > ...
<?php
require_once "2b.php";
class Airplane implements Flyer
{
public function takeOff()
{
return "Pesawat lepas 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()

```php
    {
        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";
    }

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

```php
    }

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

$airplane = new Airplane();
$bird = new Bird();
$superman = new Superman();
```

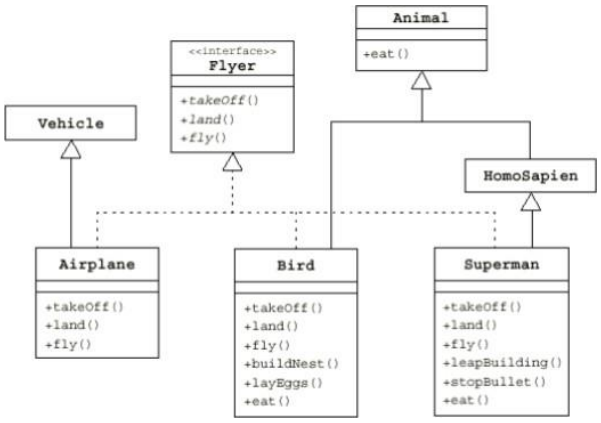


- **Output**

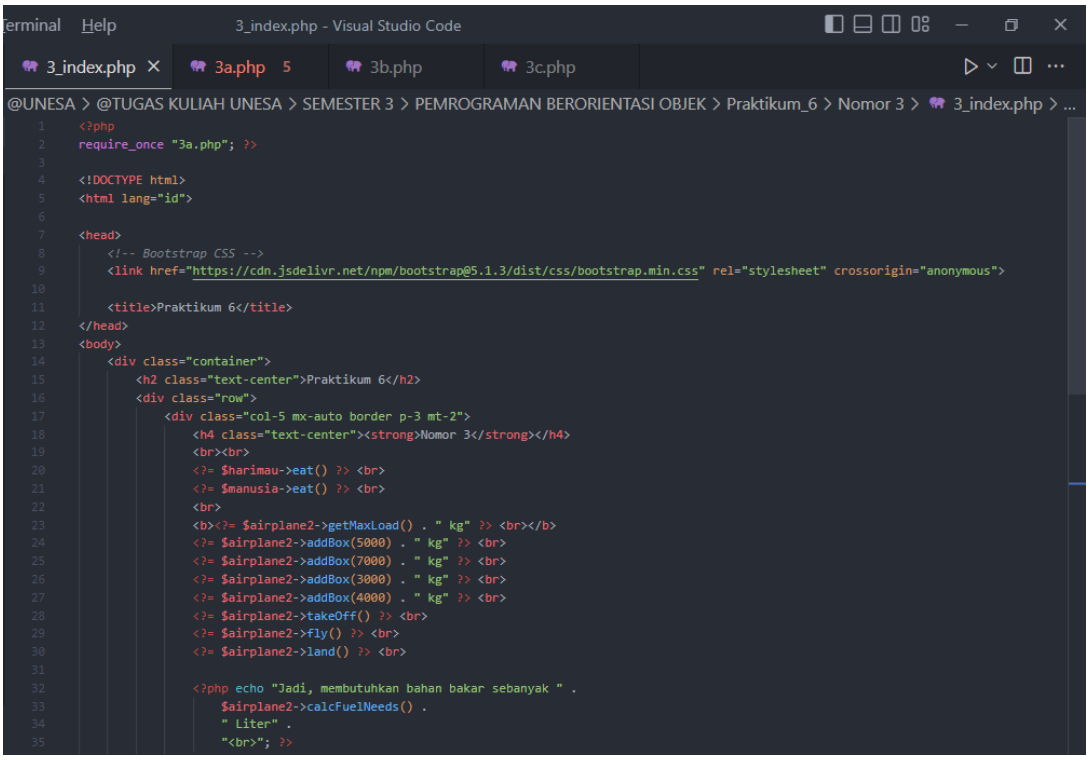


- **Analisis**
  Implementasi Polymhorpism dengan penggunaan Interface Flyer sehingga semua classyang Implements dari interface Fyler harus memiliki method takeoff, land, dan fly.

3. Buat program berdasarkan UML berikut



- **Source Code**



```
                    <br>
                    <?= $superman2->eat() ?> <br>
                    <?= $superman2->land() ?> <br>
                    <?= $superman2->takeOff() ?> <br>
                    <?= $superman2->fly() ?> <br>
                    <?= $superman2->leapBuilding() ?> <br>
                    <?= $superman2->stopBullet() ?> <br>
                </div>
            </div>
        </div>
    </body>
    </html>
```

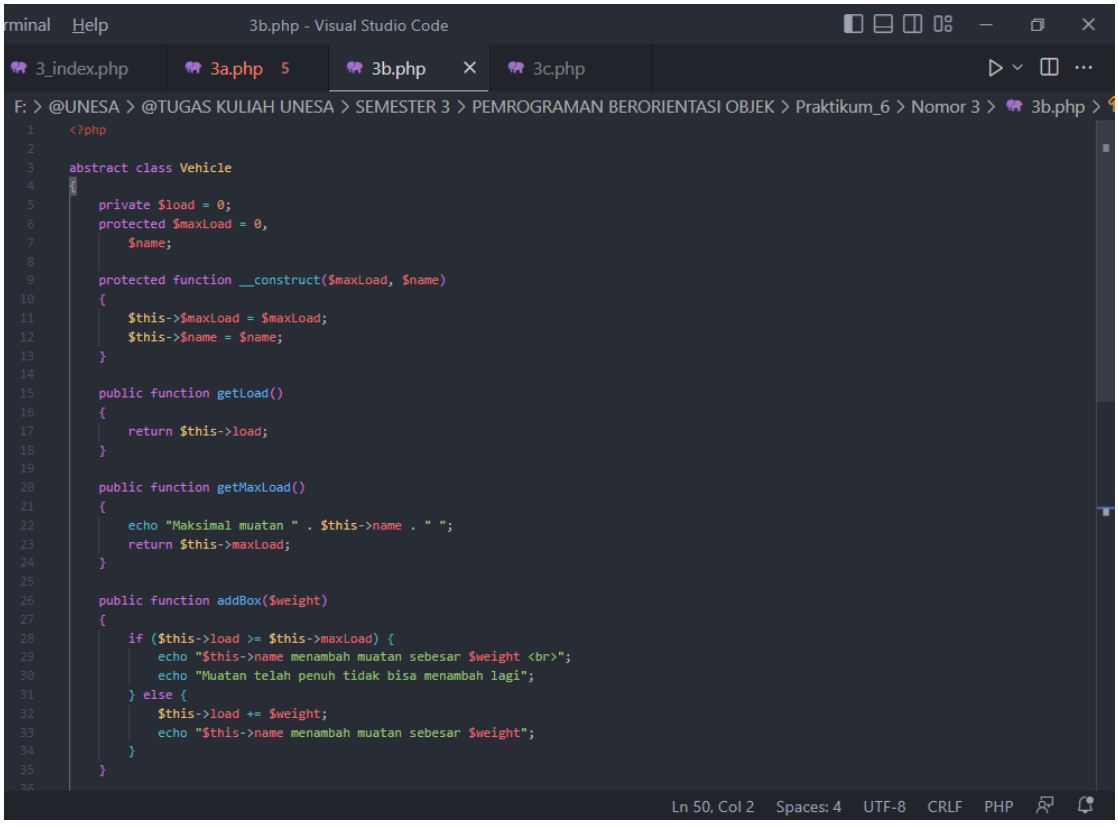
erminal Help 3a.php - Visual Studio Code

3_index.php | 3a.php 5 | 3b.php | 3c.php

F: > @UNESA > @TUGAS KULIAH UNESA > SEMESTER 3 > PEMROGRAMAN BERORIENTASI OBJEK > Praktikum_6 > Nomor 3 > 3a.php > ...

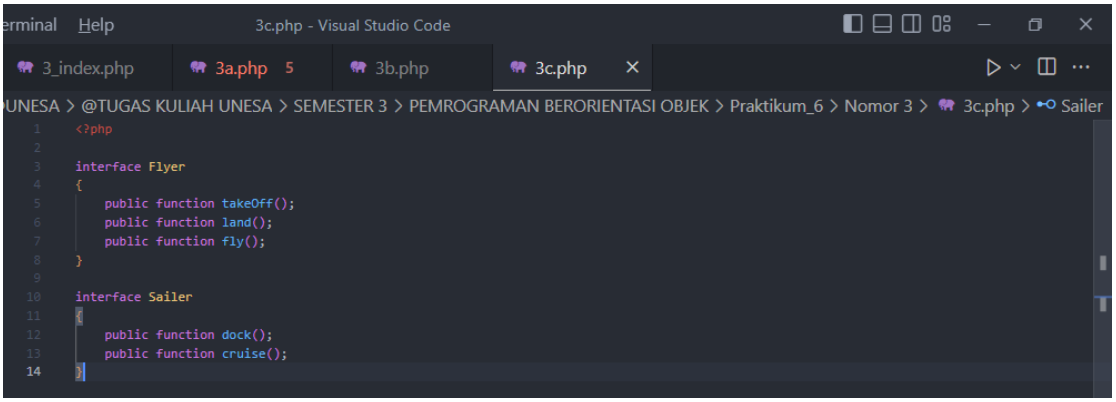
```php
<?php

require_once "3b.php";
require_once "3c.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;
    }

    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";
    }

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

    public function fly()
    {
```

```php
        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";
    }
}

$harimau = new Animal("Harimau");
$manusia = new Homosapiens("Núñez");
$airplane2 = new Airplane2(20000, "Batik Air");
$superman2 = new Superman2("Superman");
```

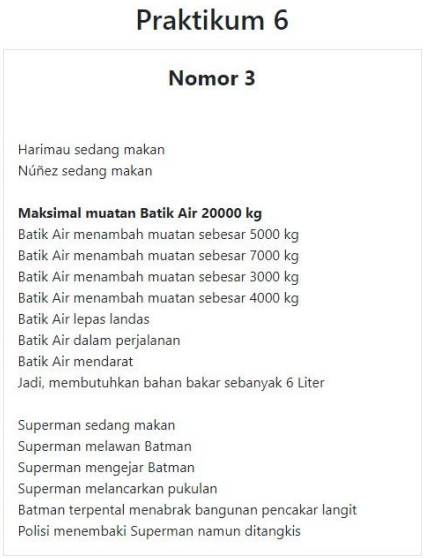



```php
<?php

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;
    }
}
```



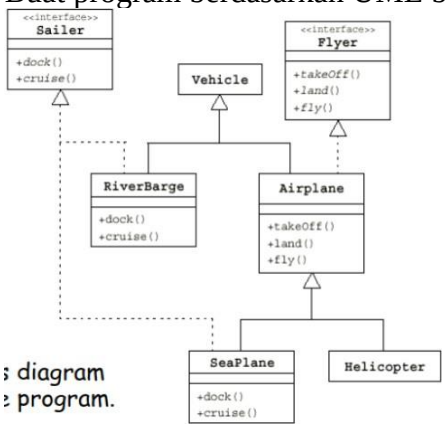
- **Output**



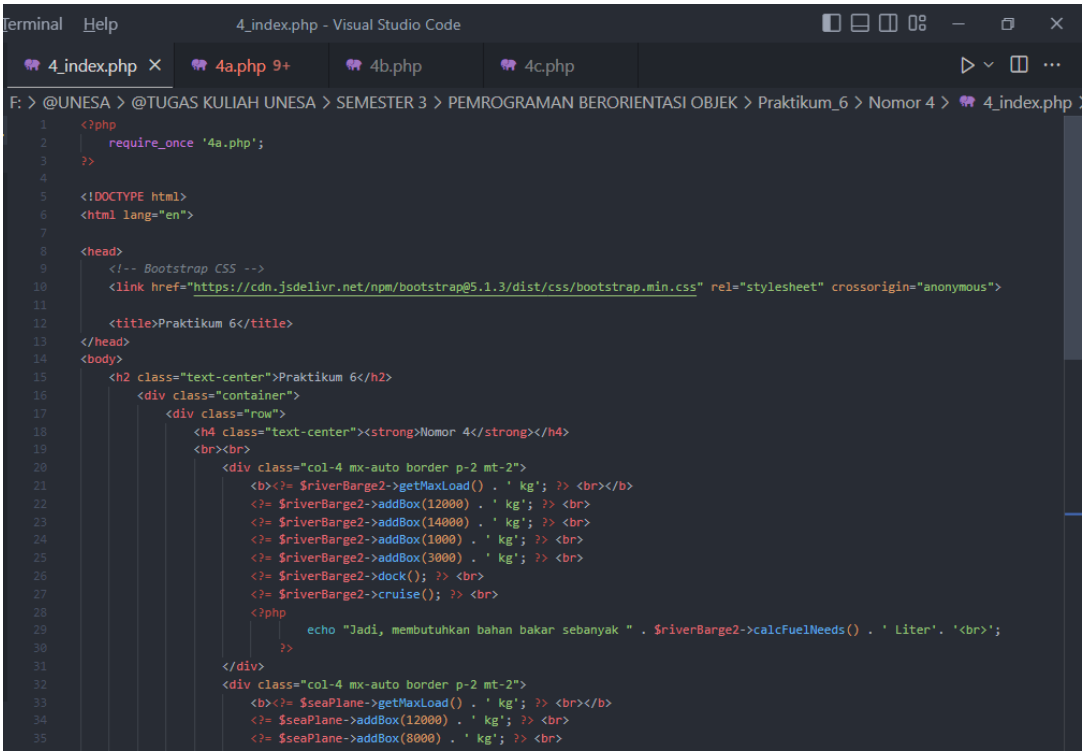
- **Analisis**

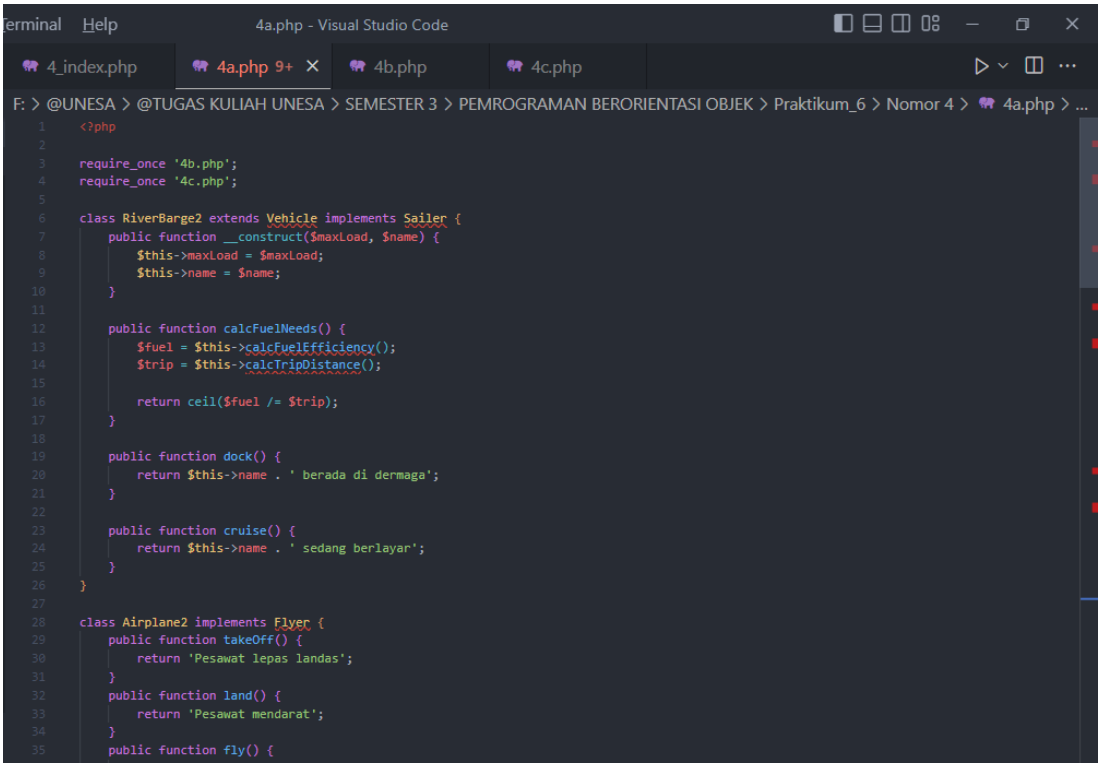
  Terdapat interface Flyer dan abstract class Vehicle. Class airplane merupakan implementasi dari interface Flyer dan turunan dari Vehicle sehingga class Airplane harus memiliki method calcFuelNeeds, takeoff, land, dan fly. Class Bird adalah implementasi dari Flyer dan turunan dari Animal sehingga memiliki 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



- **Source Code**




Terminal Help
4a.php - Visual Studio Code
4_index.php
4a.php 9+
4b.php
4c.php
F: > @UNESA > @TUGAS KULIAH UNESA > SEMESTER 3 > PEMROGRAMAN BERORIENTASI OBJEK > Praktikum_6 > Nomor 4 > 4a.php > ...
<?php
require_once '4b.php';
require_once '4c.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 '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';
}
public function takeOff() {
return $this->name . ' lepas landas';
}
public function land() {
return $this->name . ' mendarat';
}
public function fly() {
return $this->name . ' dalam perjalanan';

```php
    }
}

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';
    }

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

$riverBarge2 = new RiverBarge2(30000, 'Storeum');
$seaPlane = new SeaPlane(20000, 'Falcon');
$helicopter = new Helicopter(10000, 'Hurricane');
```

Terminal Help 4b.php - Visual Studio Code

4_index.php | 4a.php 9+ | 4b.php | 4c.php

F: > @UNESA > @TUGAS KULIAH UNESA > SEMESTER 3 > PEMROGRAMAN BERORIENTASI OBJEK > Praktikum_6 > Nomor 4 > 4b.php >

```php
<?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;
    }
}
```

Terminal Help 4c.php - Visual Studio Code

4_index.php | 4a.php 9+ | 4b.php | 4c.php

UNESA > @TUGAS KULIAH UNESA > SEMESTER 3 > PEMROGRAMAN BERORIENTASI OBJEK > Praktikum_6 > Nomor 4 > 4c.php > Sailer

```php
<?php

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

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

- **Output**

Praktikum 6

Nomor 4

Maksimal muatan Storeum 30000 kg	**Maksimal muatan Falcon 20000 kg**	**Maksimal muatan Hurricane 10000 kg**
Storeum menambah muatan sebesar 12000 kg	Falcon menambah muatan sebesar 12000 kg	Hurricane menambah muatan sebesar 8000 kg
Storeum menambah muatan sebesar 14000 kg	Falcon menambah muatan sebesar 8000 kg	Hurricane menambah muatan sebesar 2000 kg
Storeum menambah muatan sebesar 1000 kg	Falcon berada di dermaga	Hurricane lepas landas
Storeum menambah muatan sebesar 3000 kg	Falcon sedang berlayar	Hurricane dalam perjalanan
Storeum berada di dermaga	Falcon lepas landas	Hurricane mendarat
Storeum sedang berlayar	Falcon dalam perjalanan	Jadi, membutuhkan bahan bakar sebanyak 10 Liter
Jadi, membutuhkan bahan bakar sebanyak 4 Liter	Falcon mendarat	
	Jadi, membutuhkan bahan bakar sebanyak 5 Liter	

- **Analisis**

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