

**Laporan Pemrograman Berorientasi Objek**

**Praktikum 6**



Disusun Oleh :

Nadia Alifiani Raissa Pansera (21091397014)

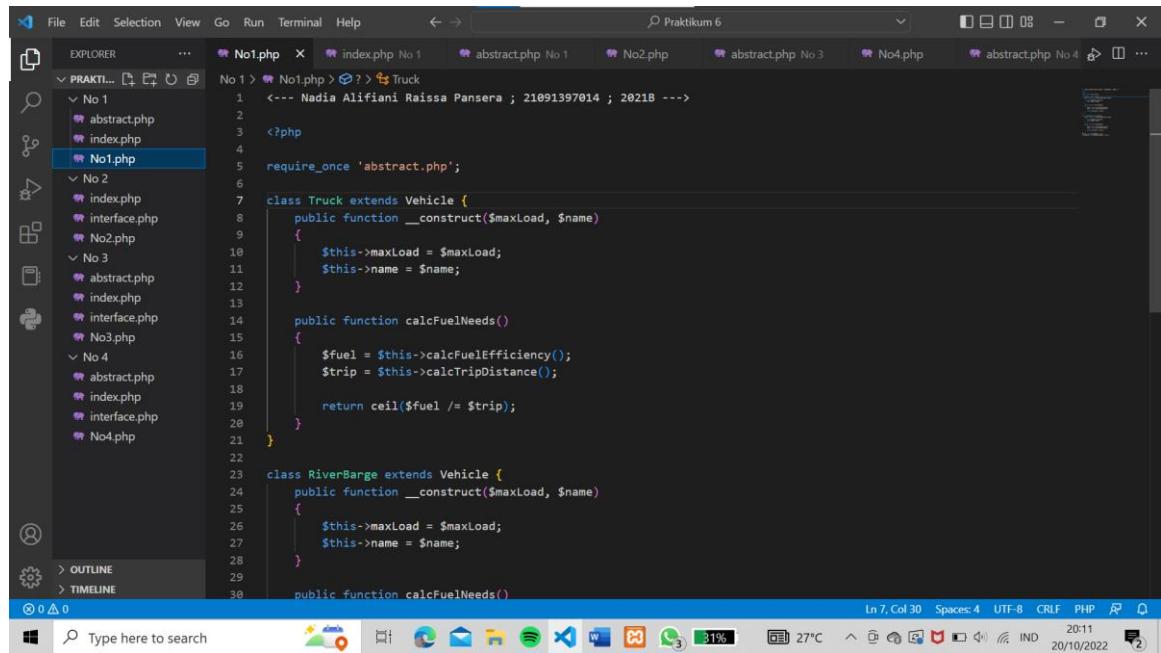
**Program Studi D4 Manajemen Informatika**

**Fakultas Vokasi**

**Universitas Negeri Surabaya**

**2022**

## 1. Source Code

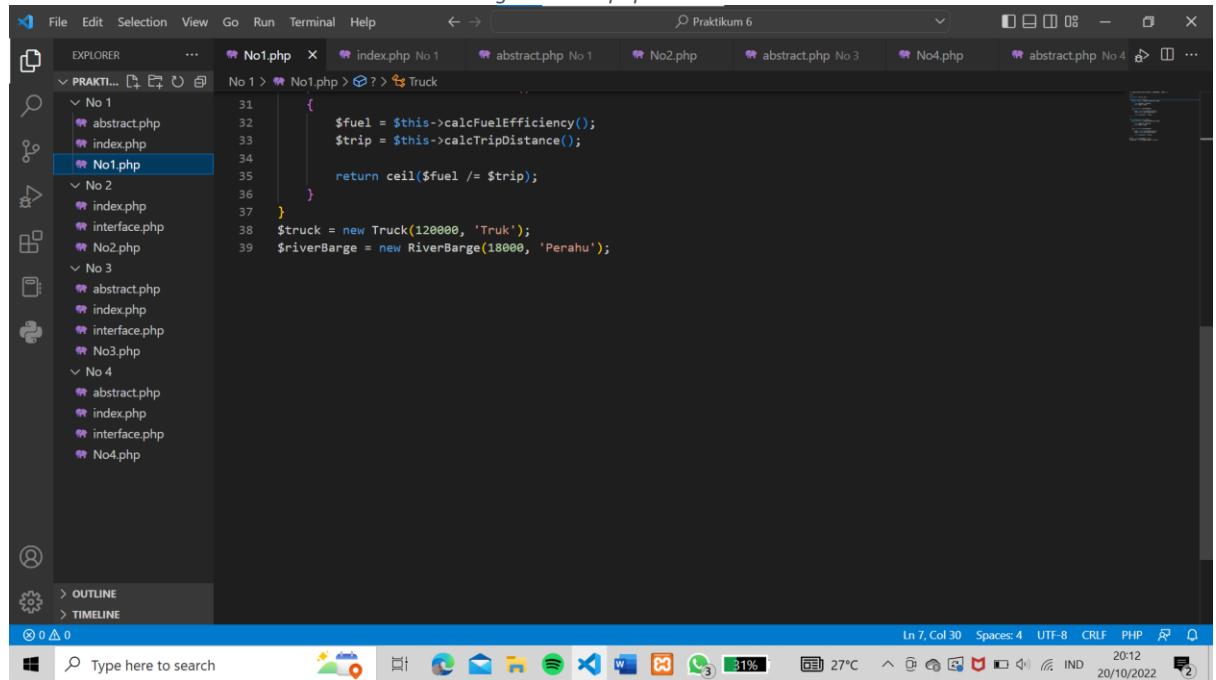


The screenshot shows a code editor interface with a dark theme. The title bar reads "No 1 > No1.php > ? > Truck". The left sidebar shows a project structure under "EXPLORER" with files like abstract.php, index.php, and No1.php. The main editor area contains PHP code:

```
No 1 > No1.php > ? > Truck
1 <-- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B --->
2
3 <?php
4
5 require_once 'abstract.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()
```

The status bar at the bottom indicates "Ln 7, Col 30 Spaces: 4 UTF-8 CRLF PHP" and the date "20/10/2022".

Figure 1 No1.php



This screenshot shows the same code editor interface as Figure 1, but with more code added to the "calcFuelNeeds" method:

```
No 1 > No1.php > ? > Truck
1
2
3     {
4         $fuel = $this->calcFuelEfficiency();
5         $trip = $this->calcTripDistance();
6
7         return ceil($fuel / $trip);
8     }
9
10    $truck = new Truck(120000, 'Truk');
11    $riverBarge = new RiverBarge(18000, 'Perahu');
```

The status bar at the bottom indicates "Ln 7, Col 30 Spaces: 4 UTF-8 CRLF PHP" and the date "20/10/2022".

Figure 2 No1.php

The screenshot shows a code editor interface with the title bar "Praktikum 6". The left sidebar is labeled "EXPLORER" and shows a project structure under "PRAKTIKUM 6". The "index.php" file is selected in the Explorer. The main pane displays the PHP code for "index.php". The code includes Bootstrap CSS imports, a title, and a row of content. The status bar at the bottom shows "Ln 43, Col 42" and other system information.

```
No 1 > index.php > ?> html > body > div.container > div.row > div.col-5.mx-auto.border.p-3.mt-2
1   <!-- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B --->
2
3   <?php
4   |   require_once 'no1.php';
5   ?
6
7   <!DOCTYPE html>
8   <html lang="id">
9
10  <head>
11    <!-- Bootstrap CSS -->
12    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
13        integrity="sha384-E7wDfH6ZDdZTbZgk/lzKuagT0FjwVdEfxnguL6Orn1wvYaM8ZPjG+uJ8v8J10Q==">
14
15    <title>PBO - Praktikum 6</title>
16  </head>
17
18  <body>
19    <div class="container">
20      <br>
21      <h2 class="text-center">PBO - Praktikum 6</h2>
22      <div class="row">
23        <div class="col-5 mx-auto border p-3 mt-2">
24          <h4 class="text-center"><strong>Soal 1</strong></h4>
25          <br><br>
26          <>?= $truck->addBox(3500) . ' kg'; ?> <br></b>
27          <br>
28          <>?= $truck->addBox(3000) . ' kg'; ?> <br>
29          <>?= $truck->addBox(5000) . ' kg'; ?> <br>
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
```

Figure 3 index.php

The screenshot shows the same code editor interface as Figure 3, but with additional PHP code added to the "index.php" file. The new code includes echo statements for truck and river barge calculations. The status bar at the bottom shows "Ln 43, Col 42" and other system information.

```
No 1 > index.php > ?> html > body > div.container > div.row > div.col-5.mx-auto.border.p-3.mt-2
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
```

Figure 4 index.php

The screenshot shows a code editor interface with multiple tabs open. The active tab is 'abstract.php' (No 1). The code in the editor is as follows:

```
<--- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B --->
<?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";
        }
    }
}
```

Figure 5 abstract.php

The screenshot shows a code editor interface with multiple tabs open. The active tab is 'abstract.php' (No 1). The code in the editor is as follows:

```
31 }
32
33 abstract public function calcFuelNeeds();
34
35 protected function calcFuelEfficiency() {
36     $range = 50000000;
37     $range /= $this->load;
38     return $range;
39 }
40
41 protected function calcTripDistance() {
42     return 500;
43 }
44 }
```

Figure 6 abstract.php

## Output

PBO - Praktikum 6

localhost/pbo/Praktikum%206/No%201/

<--- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B ---> <--- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B ---> <--- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B --->

### PBO - Praktikum 6

#### Soal 1

**Maksimal muatan Truk 120000 kg**

Truk menambah muatan sebesar 3500 kg  
Truk menambah muatan sebesar 3000 kg  
Truk menambah muatan sebesar 5000 kg  
Jadi, Butuh Bahan Bakar sebanyak 9 Liter

-----

**Maksimal muatan Perahu 18000 kg**

Perahu menambah muatan sebesar 6000 kg  
Perahu menambah muatan sebesar 1000 kg  
Perahu menambah muatan sebesar 9000 kg  
Jadi, Butuh Bahan Bakar sebanyak 7 Liter

Type here to search

27°C 20-24 20/10/2022

## 2. Source code

The screenshot shows a code editor window with the title "Praktikum 6". The left sidebar is labeled "EXPLORER" and shows a file structure under "PRAKTIKUM 6" with files like abstract.php, index.php, No1.php, No2.php, No3.php, and No4.php. The main pane displays the contents of "No2.php". The code defines two classes, Airplane and Bird, both implementing the Flyer interface. The Airplane class has methods for takeOff, land, fly, and buildNest. The Bird class has methods for takeOff, land, fly, and layEggs. The code is written in PHP, using English comments and Indonesian strings.

```
<-- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B -->
<?php
require_once 'interface.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';
    }
    public function buildNest() {
        return 'Burung membuat sarang';
    }
}

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

Figure 7 No2.php

This screenshot is identical to Figure 7, showing the same code editor window with the title "Praktikum 6". The left sidebar and main code pane are the same, displaying the PHP code for the Airplane and Bird classes. The code defines the Airplane and Bird classes with their respective methods and the Superman class.

```
<-- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B -->
<?php
require_once 'interface.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';
    }
    public function buildNest() {
        return 'Burung membuat sarang';
    }
}

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';
    }
}

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

Figure 8 No2.php

The screenshot shows a code editor window titled "Praktikum 6". The left sidebar is an "EXPLORER" view showing a directory structure for "PRAKTIKUM 6" containing files like No1.php, No2.php, No3.php, and No4.php, each with "interface.php" and "index.php" sub-files. The right pane displays the content of "interface.php" (No 2). The code defines two interfaces: Flyer and Sailer, both with methods takeOff(), land(), and fly(). A comment at the top indicates the author is Nadia Alifiani Raissa Pansera, ID 21091397014, from 2021B.

```
1  <-- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B -->
2
3  <?php
4
5  interface Flyer {
6      public function takeOff();
7      public function land();
8      public function fly();
9  }
10
11 interface Sailer {
12     public function dock();
13     public function cruise();
14 }
```

Figure 9 interface.php

The screenshot shows a code editor window titled "Praktikum 6". The left sidebar is an "EXPLORER" view showing a directory structure for "PRAKTIKUM 6" containing files like No1.php, No2.php, No3.php, and No4.php, each with "index.php" and "interface.php" sub-files. The right pane displays the content of "index.php" (No 2). The code includes a require\_once statement for "no2.php", sets up an HTML document with Bootstrap CSS, and contains PHP logic to echo "Superman" and call takeOff() and land() methods on a \$superman variable.

```
1  <-- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B -->
2
3  <?php
4      require_once 'no2.php';
5  ?>
6
7  <!DOCTYPE html>
8  <html lang="en">
9
10 <head>
11     <!-- Bootstrap CSS -->
12     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
13          integrity="sha384-E4kkBq78iYhF1dvKuhfTAU6auU8tT94WrHftDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
14
15     <title>PBO - Praktikum 6</title>
16 </head>
17
18 <body>
19     <div class="container">
20         <br>
21         <h2 class="text-center">PBO - Praktikum 6</h2>
22         <div class="row">
23             <div class="col-5 mx-auto border p-3 mt-2">
24                 <h4 class="text-center"><strong>Soal 2</strong></h4>
25                 <br><br>
26                 <b><?php
27                     echo "Superman";
28                 ?></b> <br>
29                 <?= $superman->land(); ?> <br>
30                 <?= $superman->takeOff(); ?> <br>
31         </div>
32     </div>
33 </body>
34 </html>
```

Figure 10 index.php

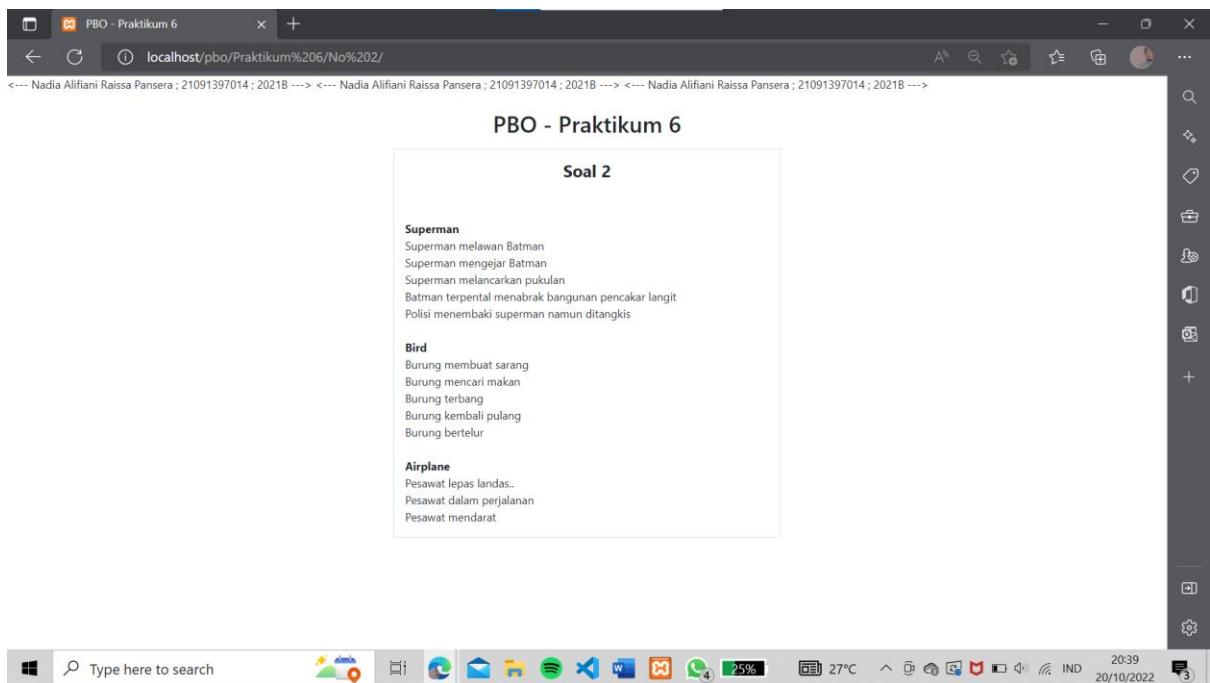
The screenshot shows a code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Toolbar:** Standard icons for file operations like Open, Save, Print, etc.
- Search Bar:** Praktikum 6
- Explorer:** Shows a tree view of files and folders under "PRAKTIKUM 6". The "index.php" file in the "No 2" folder is selected.
- Code Editor:** The content of the "index.php" file is displayed, showing PHP code that outputs text about Superman, Bird, and Airplane.
- Status Bar:** Lines 6, Col 1, Spaces: 4, UTF-8, CRLF, PHP, 2034, 20/10/2022.
- Taskbar:** Shows the Windows Start button, a search bar, and pinned application icons for File Explorer, Edge, Mail, Spotify, and others.

```
<?= $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>
53 </body>
54
55 </html>
```

Figure 11 index.php

## Output



### 3. Source code

The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows a project structure under "PRAKTIKUM 6" with files No1.php, No2.php, No3.php, and No4.php.
- Code Editor:** Displays the contents of No3.php. The code defines a class `Animal` with a constructor and an `eat()` method. It also defines a class `Homosapiens` that extends `Animal`. The code is written in PHP.
- Bottom Bar:** Includes a search bar, system tray icons, and status information like "27°C", "2040", and "20/10/2022".

```
<-- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B -->
<?php
require_once 'abstract.php';
require_once 'interface.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
```

Figure 12 No3.php

The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows a project structure under "PRAKTIKUM 6" with files No1.php, No2.php, No3.php, and No4.php.
- Code Editor:** Displays the contents of No3.php. The code defines a class `Animal` with methods `takeOff()`, `land()`, and `fly()`. It also defines a class `Superman2` that extends `Homosapiens` and implements `Flyer`. The code is written in PHP.
- Bottom Bar:** Includes a search bar, system tray icons, and status information like "27°C", "2041", and "20/10/2022".

```
<?php
require_once 'abstract.php';
require_once 'interface.php';

class Animal
{
    protected $name;

    public function __construct($name)
    {
        $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
```

Figure 13 No3.php

The screenshot shows a code editor window titled "Praktikum 6". The left sidebar displays a file tree under "PRAKTIKUM 6" with files No1.php, No2.php, No3.php, and No4.php. The main editor area contains the following PHP code:

```
No 3 > No3.php > Animal
61 public function takeOff()
62 {
63     return "$this->name mengejar Batman";
64 }
65
66 public function land()
67 {
68     return "$this->name melawan Batman";
69 }
70
71 public function fly()
72 {
73     return "$this->name melancarkan pukulan";
74 }
75
76 public function leapBuilding()
77 {
78     return "Batman terpental menabrak bangunan pencakar langit";
79 }
80
81 public function stopBullet()
82 {
83     return "Polisi menembaki $this->name namun ditangkis";
84 }
85 }
86
87 $singa = new Animal('Macam');
88 $manusia = new Homosapiens('Irwan');
89 $airplane2 = new Airplane2(20000, 'Garuda Air');
90 $superman2 = new Superman2('Superman');
```

The status bar at the bottom indicates "Ln 10, Col 11" and "2043" lines of code.

Figure 14 No3.php

The screenshot shows a code editor window titled "Praktikum 6". The left sidebar displays a file tree under "PRAKTIKUM 6" with files No1.php, No2.php, No3.php, and No4.php. The main editor area contains the following PHP code:

```
No 3 > interface.php > Flyer
1 <-- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B -->
2
3 <?php
4
5 interface Flyer {
6     public function takeOff();
7     public function land();
8     public function fly();
9 }
10
11 interface Sailer [
12     public function dock();
13     public function cruise();
14 ]
```

The status bar at the bottom indicates "Ln 14, Col 2" and "2045" lines of code.

Figure 15 interface.php

The screenshot shows a code editor interface with the title bar "Praktikum 6". The left sidebar is labeled "EXPLORER" and shows a tree view of files under "PRAKTIKUM 6", including "No 1", "No 2", "No 3", and "No 4". The main editor area displays the content of "index.php" (No 4). The code includes Bootstrap CSS imports, a title, and several PHP blocks. The status bar at the bottom shows "Ln 7, Col 16" and "2045".

```
No 3 > index.php > ↻ ?  
1  <--- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B --->  
2  
3  <?php  
4  |     require_once 'no3.php';  
5  ?>  
6  
7  <!DOCTYPE html>  
8  <html lang="id">  
9  
10 <head>  
11   <!-- Bootstrap CSS -->  
12   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"  
13   integrity="sha384-1BmE4kkBq78iYhFldvKuhfTAU6auU8tT94WrHftDbrCEXSUloBoqyl2QvZ6jIW3" crossorigin="anonymous">  
14  
15   <title>PBO - Praktikum 6</title>  
16 </head>  
17 <body>  
18   <div class="container">  
19     <h2 class="text-center">PBO - Praktikum 6</h2>  
20     <div class="row">  
21       <div class="col-5 mx-auto border p-3 mt-2">  
22         <h4 class="text-center"><strong>Soal 3</strong></h4>  
23         <br><br>  
24         <?= $singa->eat(); ?> <br>  
25         <?= $manusia->eat(); ?> <br>  
26         <br>  
27  
28         <b><?= $airplane2->getMaxLoad() . ' kg'; ?> <br></b>  
29         <?= $airplane2->addBox(5000) . ' kg'; ?> <br>  
30         <?= $airplane2->addBox(7000) . ' kg'; ?> <br>  
31  
32         <?= $airplane2->addBox(3000) . ' kg'; ?> <br>  
33         <?= $airplane2->addBox(4000) . ' kg'; ?> <br>  
34         <?= $airplane2->takeOff(); ?> <br>  
35         <?= $airplane2->fly(); ?> <br>  
36         <?= $airplane2->land(); ?> <br>  
37  
38         <?php  
39           echo "Jadi, Butuh Bahan Bakar sebanyak " . $airplane2->calcFuelNeeds() . ' Liter'. '<br>';  
40         </?php  
41         <br>  
42         <?= $superman2->eat(); ?> <br>  
43         <?= $superman2->land(); ?> <br>  
44         <?= $superman2->takeOff(); ?> <br>  
45         <?= $superman2->fly(); ?> <br>  
46         <?= $superman2->leapBuilding(); ?> <br>  
47         <?= $superman2->stopBullet(); ?> <br>  
48       </div>  
49     </div>  
50   </body>  
51 </html>
```

Figure 16 index.php

The screenshot shows a code editor interface with the title bar "Praktikum 6". The left sidebar is labeled "EXPLORER" and shows a tree view of files under "PRAKTIKUM 6", including "No 1", "No 2", "No 3", and "No 4". The main editor area displays the content of "index.php" (No 4). The code includes PHP blocks and HTML output. The status bar at the bottom shows "Ln 7, Col 16" and "2046".

```
No 3 > index.php > ↻ ?  
1  <?= $airplane2->addBox(3000) . ' kg'; ?> <br>  
2  <?= $airplane2->addBox(4000) . ' kg'; ?> <br>  
3  <?= $airplane2->takeOff(); ?> <br>  
4  <?= $airplane2->fly(); ?> <br>  
5  <?= $airplane2->land(); ?> <br>  
6  
7  <?php  
8    echo "Jadi, Butuh Bahan Bakar sebanyak " . $airplane2->calcFuelNeeds() . ' Liter'. '<br>';  
9  </?php  
10 <br>  
11  <?= $superman2->eat(); ?> <br>  
12  <?= $superman2->land(); ?> <br>  
13  <?= $superman2->takeOff(); ?> <br>  
14  <?= $superman2->fly(); ?> <br>  
15  <?= $superman2->leapBuilding(); ?> <br>  
16  <?= $superman2->stopBullet(); ?> <br>  
17  </div>  
18  </div>  
19  </body>  
20 </html>
```

Figure 17 index.php

The screenshot shows a code editor window with the title "Praktikum 6". The left sidebar is labeled "EXPLORER" and shows a file structure under "PRAKTIKUM 6". The main pane displays the content of the "abstract.php" file (No 3). The code defines an abstract class "Vehicle" with methods for load, max load, and adding weight. The status bar at the bottom indicates "Ln 11, Col 6" and "2048" lines of code.

```
1 <--- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B --->
2 <?php
3
4 abstract class Vehicle {
5     private $load = 0;
6     protected $maxLoad = 0, $name;
7
8     protected function __construct($maxLoad, $name) {
9         $this->$maxLoad = $maxLoad;
10        $this->$name = $name;
11    }
12
13    public function getLoad() {
14        return $this->load;
15    }
16
17    public function getMaxLoad() {
18        echo "Maksimal muatan' . $this->name . '";
19        return $this->maxLoad;
20    }
21
22    public function addBox($weight) {
23        if ($this->load >= $this->maxLoad) {
24            echo "$this->name menambah muatan sebesar $weight <br>";
25            echo 'Muatan telah penuh tidak bisa menambah lagi';
26        }else {
27            $this->load += $weight;
28            echo "$this->name menambah muatan sebesar $weight";
29        }
30    }

```

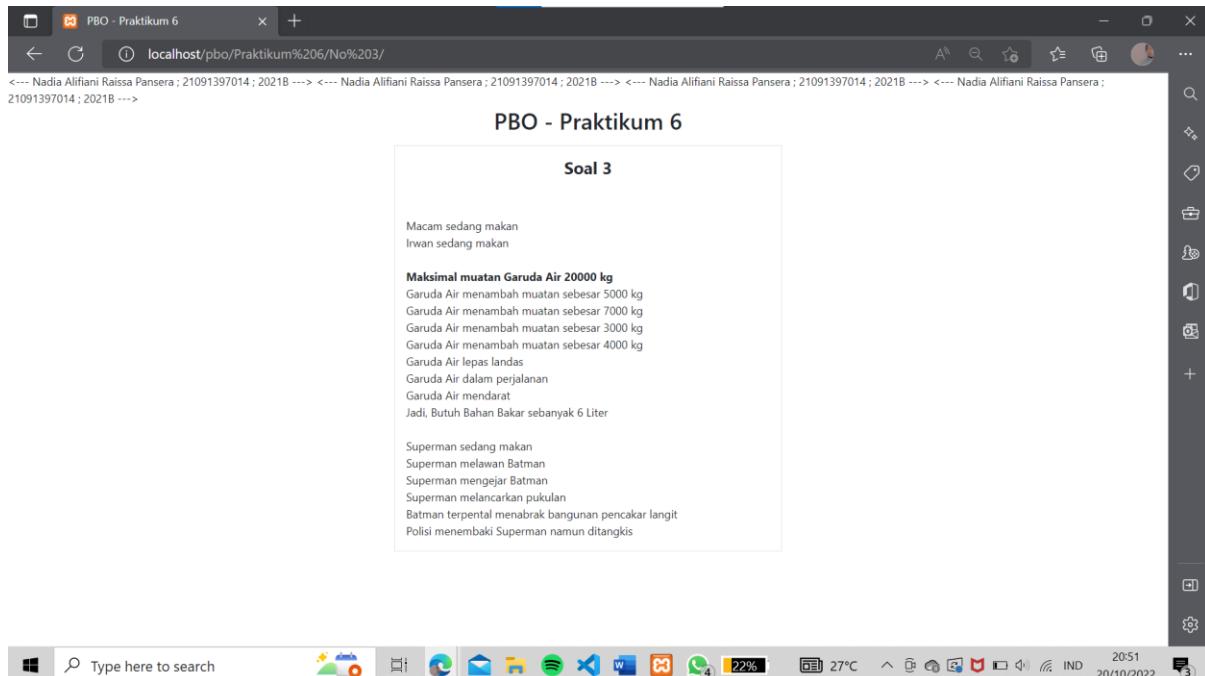
Figure 18 abstract.php

The screenshot shows a code editor window with the title "Praktikum 6". The left sidebar is labeled "EXPLORER" and shows a file structure under "PRAKTIKUM 6". The main pane displays the content of the "abstract.php" file (No 3). The code adds methods for fuel needs, efficiency, and trip distance. The status bar at the bottom indicates "Ln 11, Col 6" and "2049" lines of code.

```
31
32     abstract public function calcFuelNeeds();
33
34     protected function calcFuelEfficiency() {
35         $range = 50000000;
36         $range /= $this->load;
37         return $range;
38     }
39
40     protected function calcTripDistance() {
41         return 500;
42     }
43 }
```

Figure 19 abstract.php

## Output :



## 4. Source code

```
No 4 > No4.php > ?  
1 <--- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B --->  
2  
3 <?php  
4  
5 require_once 'abstract.php';  
6 require_once 'interface.php';  
7  
8 class RiverBarge2 extends Vehicle implements Sailer {  
9     public function __construct($maxLoad, $name) {  
10         $this->maxLoad = $maxLoad;  
11         $this->name = $name;  
12     }  
13  
14     public function calcFuelNeeds() {  
15         $fuel = $this->calcFuelEfficiency();  
16         $trip = $this->calcTripDistance();  
17  
18         return ceil($fuel / $trip);  
19     }  
20  
21     public function dock() {  
22         return $this->name . ' berada di dermaga';  
23     }  
24  
25     public function cruise() {  
26         return $this->name . ' sedang berlayar';  
27     }  
28 }  
29  
30 class Airplane2 implements Flyer {
```

abstract.php No 3 No4.php abstract.php No 4 interface.php No 4 index.php No 4 No3.php interface.php

27°C 22% 2052 20/10/2022

Figure 20 No4.php

The screenshot shows a code editor interface with two tabs open: 'abstract.php No 3' and 'No4.php'. The 'No4.php' tab is active, displaying the following PHP code:

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

The code editor has a dark theme, and the file structure on the left shows several other PHP files under 'PRAKTIKUM 6'.

Figure 21 No4.php

The screenshot shows a code editor interface with two tabs open: 'abstract.php No 3' and 'No4.php'. The 'No4.php' tab is active, displaying the following PHP code:

```
61 }
62
63     public function takeOff() {
64         return $this->name . ' lepas landas';
65     }
66
67     public function land() {
68         return $this->name . ' mendarat';
69     }
70
71     public function fly() {
72         return $this->name . ' dalam perjalanan';
73     }
74 }
75
76 class Helicopter extends Vehicle {
77     public function __construct($maxLoad, $name) {
78         $this->maxLoad = $maxLoad;
79         $this->name = $name;
80     }
81
82     public function calcFuelNeeds() {
83         $fuel = $this->calcFuelEfficiency();
84         $trip = $this->calcTripDistance();
85
86         return ceil($fuel / $trip);
87     }
88     public function takeOff() {
89         return $this->name . ' lepas landas';
90     }
}
```

The code editor has a dark theme, and the file structure on the left shows several other PHP files under 'PRAKTIKUM 6'.

Figure 22 No4.php

The screenshot shows a code editor window titled "Praktikum 6". The left sidebar displays a file tree under "PRAKTIKUM 6" with several PHP files: abstract.php, index.php, No1.php, No2.php, No3.php, and No4.php. The main editor area shows the content of the "No4.php" file:

```
91     public function land() {
92         return $this->name . ' mendarat';
93     }
94
95     public function fly() {
96         return $this->name . ' dalam perjalanan';
97     }
98 }
99
100 $riverBarge2 = new RiverBarge2(80000, 'Atomic');
101 $seaPlane = new Seaplane(20000, 'Maldives');
102 $helicopter = new Helicopter(120000, 'Ultralight');
```

The status bar at the bottom indicates "Ln 4, Col 1" and "20:55 20/10/2022".

Figure 23 No4.php

The screenshot shows a code editor window titled "Praktikum 6". The left sidebar displays a file tree under "PRAKTIKUM 6" with several PHP files: abstract.php, index.php, No1.php, No2.php, No3.php, and No4.php. The main editor area shows the content of the "interface.php" file:

```
1     <-- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B -->
2
3     <?php
4
5     interface Flyer {
6         public function takeOff();
7         public function land();
8         public function fly();
9     }
10
11    interface Sailer {
12        public function dock();
13        public function cruise();
14    }
```

The status bar at the bottom indicates "Ln 14, Col 2" and "20:56 20/10/2022".

Figure 24 interface.php

The screenshot shows a code editor window titled "Praktikum 6". The left sidebar displays a file tree under "PRAKTIKUM 6" with files like abstract.php, index.php, interface.php, and No1.php through No4.php. The main editor area shows the content of the "index.php" file. The code includes PHP logic for calculating fuel needs based on vehicle types (RiverBarge, SeaPlane, Helicopter) and their actions (cruise, dock, takeoff, fly, land). The code uses conditional statements and loops to process multiple vehicles.

```
No 4 > index.php > ? > html
1  <--- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B --->
2
3  <?php
4      require_once 'no4.php';
5  ?>
6
7  <!DOCTYPE html>
8  <html lang="en">
9
10 <head>
11     <!-- Bootstrap CSS -->
12     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
13          integrity="sha384-E7yJcaIv1yWVnS+G0+Iq8H5I6jYfKUyDZQf0XQ+S6JHj0lXoOoY6LWz6jIW3" crossorigin="anonymous">
14
15     <title>PBO - Praktikum 6</title>
16 </head>
17 <body>
18     <h2 class="text-center">PBO - Praktikum 6</h2>
19     <div class="container">
20         <div class="row">
21             <h4 class="text-center"><strong>Soal 4</strong></h4>
22             <br><br>
23             <div class="col-4 mx-auto border p-2 mt-2">
24                 <b><?= $riverBarge2->getMaxLoad() . ' kg'; ?> <br></b>
25                 <?= $riverBarge2->addBox(12000) . ' kg'; ?> <br>
26                 <?= $riverBarge2->addBox(14000) . ' kg'; ?> <br>
27                 <?= $riverBarge2->addBox(1000) . ' kg'; ?> <br>
28                 <?= $riverBarge2->addBox(3000) . ' kg'; ?> <br>
29                 <?= $riverBarge2->dock(); ?> <br>
30                 <?= $riverBarge2->cruise(); ?> <br>
```

Figure 25 index.php

The screenshot shows a code editor window titled "Praktikum 6". The left sidebar displays a file tree under "PRAKTIKUM 6" with files like abstract.php, index.php, interface.php, and No1.php through No4.php. The main editor area shows the content of the "index.php" file. The code includes PHP logic for calculating fuel needs based on vehicle types (RiverBarge, SeaPlane, Helicopter) and their actions (cruise, dock, takeoff, fly, land). The code uses conditional statements and loops to process multiple vehicles.

```
No 4 > index.php > ? > html
30     <?= $riverBarge2->cruise(); ?> <br>
31     <?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     <b><?= $seaPlane->getMaxLoad() . ' kg'; ?> <br></b>
37     <?= $seaPlane->addBox(12000) . ' kg'; ?> <br>
38     <?= $seaPlane->addBox(8000) . ' kg'; ?> <br>
39     <?= $seaPlane->dock(); ?> <br>
40     <?= $seaPlane->cruise(); ?> <br>
41     <?= $seaPlane->takeOff(); ?> <br>
42     <?= $seaPlane->fly(); ?> <br>
43     <?= $seaPlane->land(); ?> <br>
44     <?php
45         echo "Jadi, Butuh Bahan Bakar sebanyak " . $seaPlane->calcFuelNeeds() . ' Liter'. '<br>';
46     ?>
47 </div>
48 <div class="col mx-auto border p-2 mt-2">
49     <b><?= $helicopter->getMaxLoad() . ' kg'; ?> <br></b>
50     <?= $helicopter->addBox(8000) . ' kg'; ?> <br>
51     <?= $helicopter->addBox(2000) . ' kg'; ?> <br>
52     <?= $helicopter->takeOff(); ?> <br>
53     <?= $helicopter->fly(); ?> <br>
54     <?= $helicopter->land(); ?> <br>
55     <?php
56         echo "Jadi, Butuh Bahan Bakar sebanyak " . $helicopter->calcFuelNeeds() . ' Liter'. '<br>';
57     ?>
58 </div>
59 </div>
60 </body>
61 </html>
```

Figure 26 index.php

The screenshot shows a code editor interface with multiple tabs open. The active tab is 'abstract.php' (No 4). The code editor displays the following PHP code:

```
No 4 > abstract.php > Vehicle
1  <-- Nadia Alifiani Raissa Pansera ; 21091397014 ; 2021B ---
2
3  <?php
4
5  abstract class Vehicle {
6      private $load = 0;
7      protected $maxLoad = 0, $name;
8
9      protected function __construct($maxLoad, $name) {
10         $this->$maxLoad = $maxLoad;
11         $this->$name = $name;
12     }
13
14     public function getLoad() {
15         return $this->load;
16     }
17
18     public function getMaxLoad() {
19         echo "Maksimal muatan ' . $this->name . ' ";
20         return $this->maxLoad;
21     }
22
23     public function addBox($weight) {
24         if ($this->load >= $this->maxLoad) {
25             echo "$this->name menambah muatan sebesar $weight <br>";
26             echo "Muatan telah penuh tidak bisa menambah lagi";
27         }else {
28             $this->load += $weight;
29             echo "$this->name menambah muatan sebesar $weight";
30         }
31     }
32
33     abstract public function calcFuelNeeds();
34
35     protected function calcFuelEfficiency() {
36         $range = 5000000;
37         $range /= $this->load;
38         return $range;
39     }
40
41     protected function calcTripDistance() {
42         return 500;
43     }
44 }
```

The code defines an abstract class 'Vehicle' with methods for getting load, maximum load, adding weight, calculating fuel needs, and calculating trip distance. The code is annotated with comments and developer information at the top.

Figure 27 abstract.php

The screenshot shows a code editor interface with multiple tabs open. The active tab is 'abstract.php' (No 4). The code editor displays the following PHP code:

```
No 4 > abstract.php > Vehicle
31
32
33     abstract public function calcFuelNeeds();
34
35     protected function calcFuelEfficiency() {
36         $range = 5000000;
37         $range /= $this->load;
38         return $range;
39     }
40
41     protected function calcTripDistance() {
42         return 500;
43     }
44 }
```

This version of the code is identical to Figure 27 but lacks the initial developer annotations and the constructor definition.

Figure 28 abstract.php

## Output :

The screenshot shows a Windows desktop environment. At the top is a taskbar with various icons for quick access. Below the taskbar is a browser window titled "PBO - Praktikum 6" with the URL "localhost/pbo/Praktikum%206/No%204/". The browser content displays a table with three columns. The first column is titled "Maksimal muatan Atomic 8000 kg" and lists: "Atomic menambah muatan sebesar 12000 kg", "Atomic menambah muatan sebesar 14000 kg", "Atomic menambah muatan sebesar 1000 kg", "Atomic menambah muatan sebesar 3000 kg", and "Jadi, Butuh Bahan Bakar sebanyak 4 Liter". The second column is titled "Maksimal muatan Maldives 20000 kg" and lists: "Maldives menambah muatan sebesar 12000 kg", "Maldives menambah muatan sebesar 8000 kg", "Maldives berada di dermaga", "Maldives sedang berlayar", "Maldives lepas landas", "Maldives dalam perjalanan", and "Maldives mendarat". The third column is titled "Maksimal muatan Ultralight 120000 kg" and lists: "Ultralight menambah muatan sebesar 8000 kg", "Ultralight menambah muatan sebesar 2000 kg", "Ultralight lepas landas", "Ultralight dalam perjalanan", "Ultralight mendarat", and "Jadi, Butuh Bahan Bakar sebanyak 10 Liter". The taskbar also shows system status like battery level (19%), temperature (27°C), and date/time (20/10/2022, 21:04).

Maksimal muatan Atomic 8000 kg	Maksimal muatan Maldives 20000 kg	Maksimal muatan Ultralight 120000 kg
Atomic menambah muatan sebesar 12000 kg	Maldives menambah muatan sebesar 12000 kg	Ultralight menambah muatan sebesar 8000 kg
Atomic menambah muatan sebesar 14000 kg	Maldives menambah muatan sebesar 8000 kg	Ultralight menambah muatan sebesar 2000 kg
Atomic menambah muatan sebesar 1000 kg	Maldives berada di dermaga	Ultralight lepas landas
Atomic menambah muatan sebesar 3000 kg	Maldives sedang berlayar	Ultralight dalam perjalanan
Jadi, Butuh Bahan Bakar sebanyak 4 Liter	Maldives lepas landas	Ultralight mendarat
	Maldives dalam perjalanan	Jadi, Butuh Bahan Bakar sebanyak 10 Liter
	Maldives mendarat	