

TUGAS PRAKTIKUM 6
PEMROGRAMAN BERORIENTASI OBJEK



Oleh :

RAHMADITYA PUTRI LAILATUL ISMI
21091397036
2021B

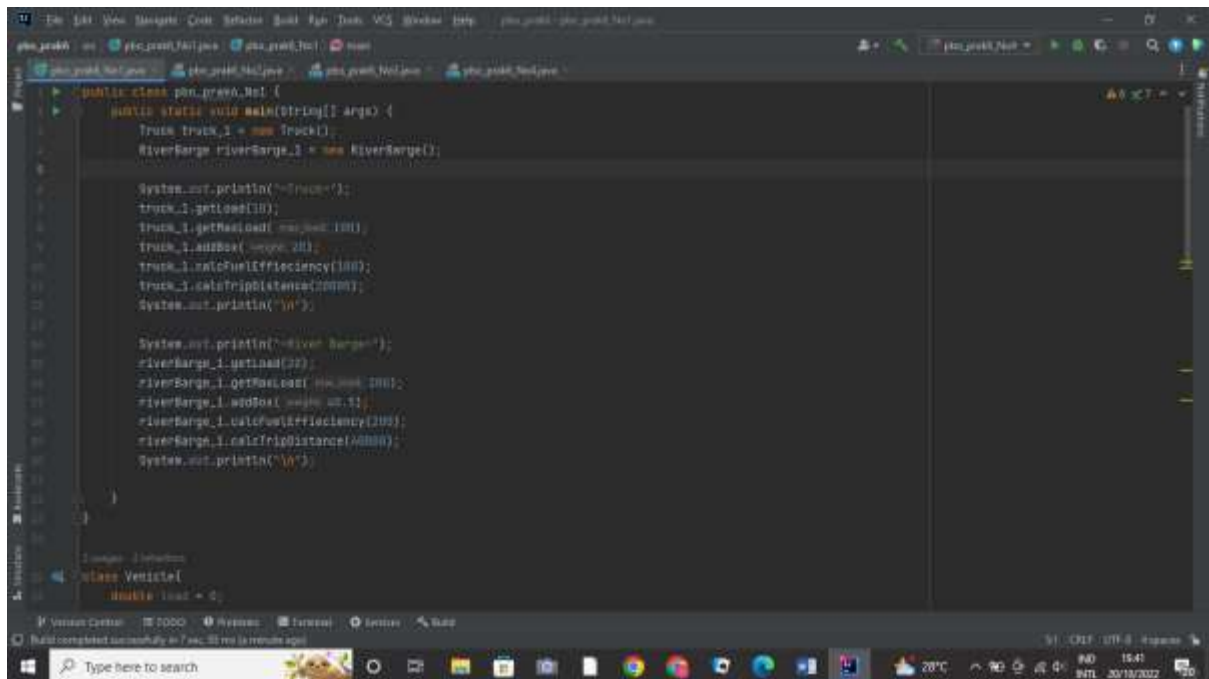
PROGRAM STUDI MANAJEMEN INFORMATIKA

FAKULTAS VOKASI

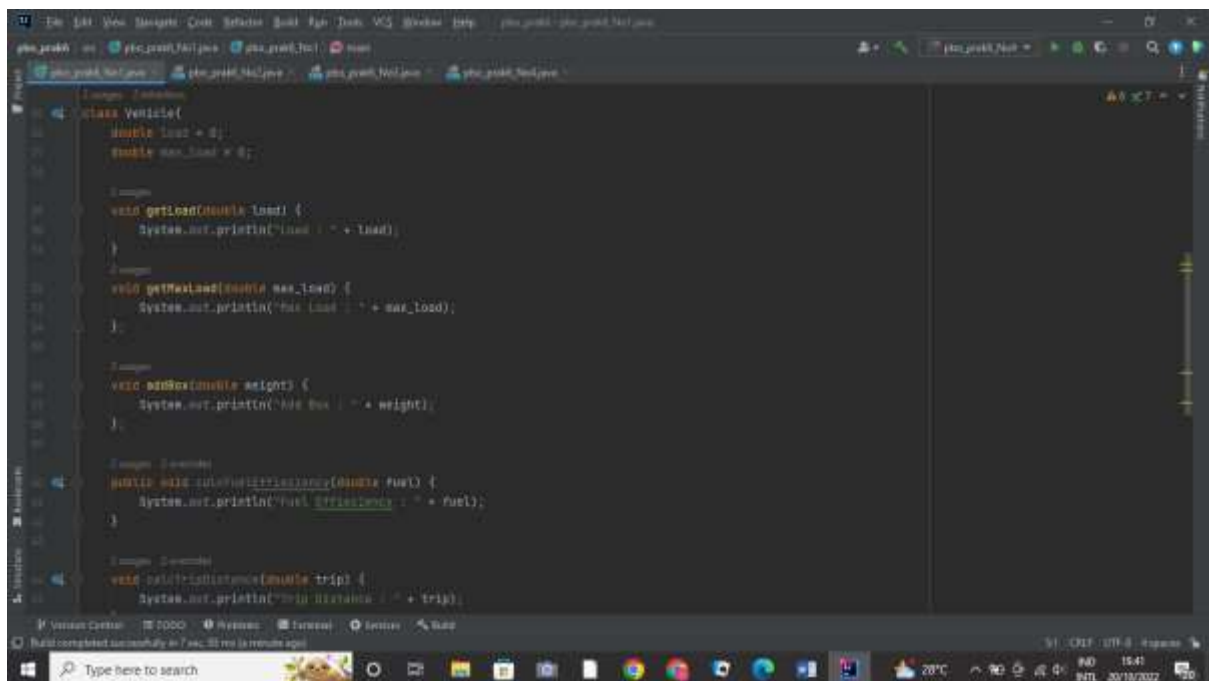
UNIVERSITAS NEGERI SURABAYA

2022

1.



```
public class p01_p01_Net {  
    public static void main(String[] args) {  
        Truck truck_1 = new Truck();  
        RiverBarge riverBarge_1 = new RiverBarge();  
  
        System.out.println("Truck");  
        truck_1.getLoad();  
        truck_1.getMaxLoad( maxLoad: 100);  
        truck_1.addBox( weight: 20);  
        truck_1.calcFuelEfficiency(100);  
        truck_1.calcTripDistance(2000);  
        System.out.println("\n");  
  
        System.out.println("River Barge");  
        riverBarge_1.getLoad();  
        riverBarge_1.getMaxLoad( maxLoad: 100);  
        riverBarge_1.addBox( weight: 40.1);  
        riverBarge_1.calcFuelEfficiency(200);  
        riverBarge_1.calcTripDistance(4000);  
        System.out.println("\n");  
    }  
}
```



```
class Vehicle {  
    double load = 0;  
    double max_load = 0;  
  
    //getter  
    void getLoad(double load) {  
        System.out.println("load : " + load);  
    }  
  
    //getter  
    void getMaxLoad(double max_load) {  
        System.out.println("max_load : " + max_load);  
    }  
  
    //setter  
    void addBox(double weight) {  
        System.out.println("add Box : " + weight);  
    }  
  
    //method  
    public void calcFuelEfficiency(double fuel) {  
        System.out.println("fuel efficiency : " + fuel);  
    }  
  
    //method  
    void calcTripDistance(double trip) {  
        System.out.println("trip distance : " + trip);  
    }  
}
```

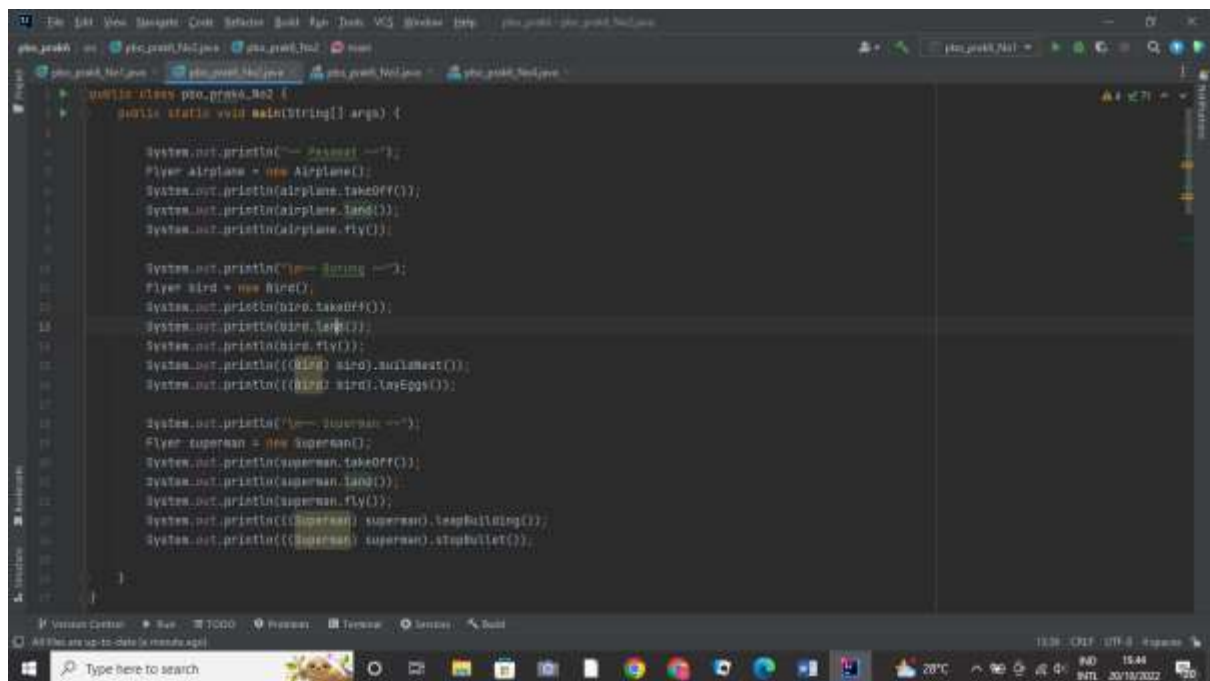
```
1 // vehicle.java
2
3 package pbo.pasik.Nel;
4
5 public class vehicle {
6     // example
7     public void calcFuelEfficiency(double fuel) {
8         System.out.println("Fuel Efficiency : " + fuel);
9     }
10
11     // example
12     void calcTripDistance(double trip) {
13         System.out.println("Trip Distance : " + trip);
14     }
15 }
16
17 // Truck.java
18 package pbo.pasik.Nel;
19
20 public class Truck extends vehicle {
21     // example
22     public void calcFuelEfficiency(double fuel) {
23         System.out.println("Fuel Efficiency : " + fuel);
24     }
25
26     // example
27     void calcTripDistance(double trip) {
28         System.out.println("Trip Distance : " + trip);
29     }
30 }
31
32 // RiverBarge.java
33 package pbo.pasik.Nel;
34
35 public class RiverBarge extends vehicle {
36     // example
37     public void calcFuelEfficiency(double fuel) {
38         System.out.println("Fuel Efficiency : " + fuel);
39     }
40
41     // example
42     void calcTripDistance(double trip) {
43         System.out.println("Trip Distance : " + trip);
44     }
45 }
```

HASIL

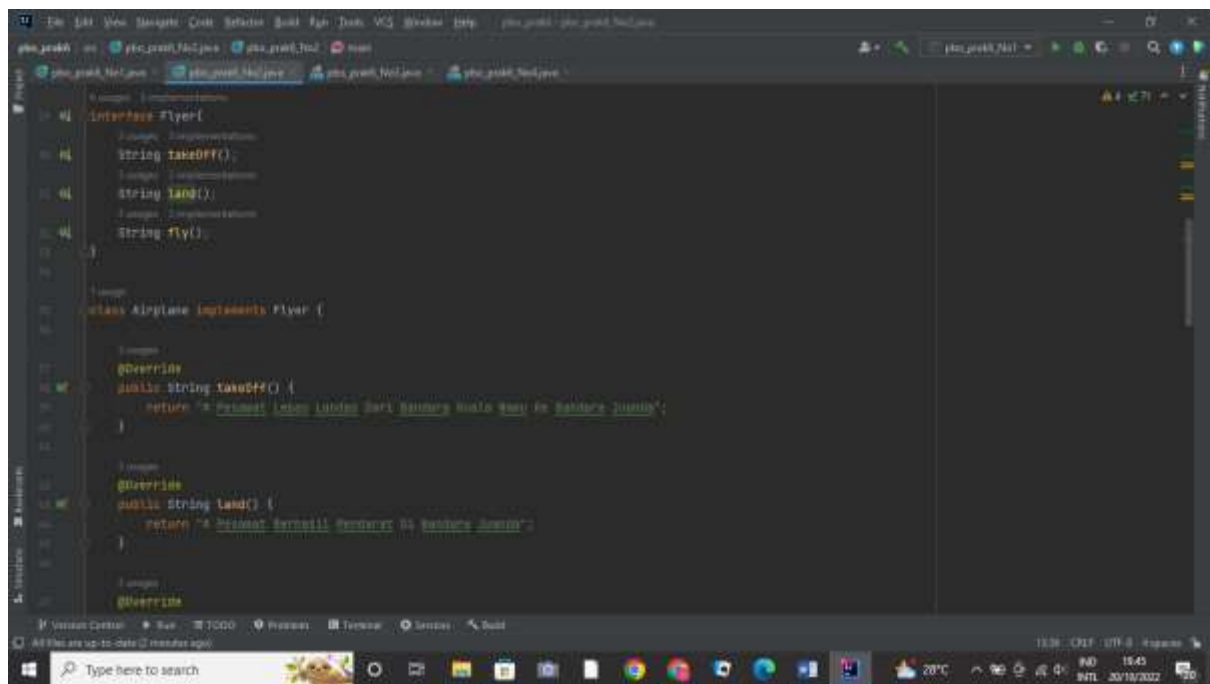
```
1 C:\Users\user\j2mispj2m-19\bin\java.exe ~\javaagent:~\Program Files\JetBrains\IntelliJ IDEA Community Edition 2022.2.2\lib\idea_rt.jar=90658-C:\Program
2 Files\Java\jdk-19\bin\java.exe
3
4 -Truck-
5 Load : 10.0
6 Max Load : 100.0
7 Add Box : 10.0
8 Fuel Efficiency : 100.0
9 Trip Distance : 10000.0
10
11 -River Barge-
12 Load : 20.0
13 Max Load : 200.0
14 Add Box : 20.0
15 Fuel Efficiency : 200.0
16 Trip Distance : 20000.0
17
18 Process finished with exit code 0
```

Dari kode program diatas terdapat 3 class yaitu vehicle, truck, riverbarge dimana class vehicle nantinya mewariskan methodnya ke class childnya. Didalam class truck dan riverbarge terdapat 2 method yang sama yaitu method bensin dan jarak tempuh. Kemudian, method truck dan riverbarge dipanggil difungsi main.

2.



```
1 public class pto_prak_02 {
2     public static void main(String[] args) {
3
4         System.out.println("\n-- Pesawat --");
5         Flyer airplane = new Airplane();
6         System.out.println(airplane.takeOff());
7         System.out.println(airplane.land());
8         System.out.println(airplane.fly());
9
10        System.out.println("\n-- Burung --");
11        Flyer bird = new Bird();
12        System.out.println(bird.takeOff());
13        System.out.println(bird.land());
14        System.out.println(bird.fly());
15        System.out.println(((Bird) bird).swim());
16        System.out.println(((Bird) bird).layEggs());
17
18        System.out.println("\n-- Superman --");
19        Flyer superman = new Superman();
20        System.out.println(superman.takeOff());
21        System.out.println(superman.land());
22        System.out.println(superman.fly());
23        System.out.println(((Superman) superman).throwBullet());
24        System.out.println(((Superman) superman).stopBullet());
25
26    }
27 }
```



```
1 interface Flyer {
2     String takeOff();
3     String land();
4     String fly();
5 }
6
7 class Airplane implements Flyer {
8     @Override
9     public String takeOff() {
10        return "Pesawat lepas landas dari Bandara Kuala Lumpur ke Bandara Jember";
11    }
12
13     @Override
14     public String land() {
15        return "Pesawat berlandar di Bandara Jember";
16    }
17
18     @Override
19 }
```

```
File Edit View Settings Code Refactor Build Run Tools VCS Window Help yhs.prakt1 - yhs.prakt1.Noel.java
yhs.prakt1.Noel.java yhs.prakt1.Noel.java yhs.prakt1.Noel.java yhs.prakt1.Noel.java
17 @Override
18 public String fly() {
19     return "Penguin terbang jelajah 5 jam";
20 }
21
22
23 // Penguin
24 class Bird implements Flyer {
25
26     // Penguin
27     @Override
28     public String takeOff() {
29         return "Burung Pingu terbang dari Antartika dan tidak bisa berenang laut";
30     }
31
32     // Penguin
33     @Override
34     public String land() {
35         return "Burung Pingu beresat turun dari atas permukaan laut";
36     }
37
38     // Penguin
39     @Override
40     public String fly() {
41         return "Burung Pingu terbang jelajah 5 jam";
42     }
43 }
```

```
File Edit View Settings Code Refactor Build Run Tools VCS Window Help yhs.prakt1 - yhs.prakt1.Noel.java
yhs.prakt1.Noel.java yhs.prakt1.Noel.java yhs.prakt1.Noel.java yhs.prakt1.Noel.java
17
18 public String swallowEgg() {
19     return "Burung Pingu menelamkan telur-telurnya di dalam bejana";
20 }
21
22 // Penguin
23 public String layEgg() {
24     return "Burung Pingu menelamkan telur-telurnya";
25 }
26
27 // Penguin
28 class Superman implements Flyer {
29
30     // Penguin
31     @Override
32     public String takeOff() {
33         return "Superman terbang dengan sangat cepat tanpa perlu melompat pertama";
34     }
35
36     // Penguin
37     @Override
38     public String land() {
39         return "Superman mendarat di tempat melompat pertama";
40     }
41
42     // Penguin
43     @Override
44 }
```


3.

```

1  public class pto_prog4_905 {
2      public static void main(String[] args) {
3
4          System.out.println("==> Program ==>");
5          Vehicle airplane_1 = new Vehicle();
6          airplane_1.getFuel(10);
7          airplane_1.getMaxFuel(100);
8          airplane_1.setFuel(100);
9          airplane_1.calculateFuelConsumption(100);
10         airplane_1.calculateTripDistance(2000);
11         System.out.println("V");
12
13         Flyer airplane = new Airplane();
14         System.out.println(airplane.takeOff());
15         System.out.println(airplane.land());
16         System.out.println(airplane.fly());
17
18         System.out.println("A==> Summary ==>");
19         Bird bird = new Bird();
20         System.out.println(bird.takeOff());
21         System.out.println(bird.land());
22         System.out.println(bird.fly());
23         System.out.println(bird.birdTest());
24         System.out.println(bird.layEggs());
25         System.out.println(bird.get("bird")+"==> Summary ==>");
26
27         System.out.println("==> Program ==>");
28     }
29 }

```

Build completed successfully in 15 ms (13 ms in 15 ms ago)

```

1  System.out.println("I'm Superman!");
2  Superman superman = new Superman();
3  System.out.println(superman.takeOff());
4  System.out.println(superman.land());
5  System.out.println(superman.fly());
6  System.out.println(superman.landBuilding());
7  System.out.println(superman.stopBullet());
8  System.out.println(superman.tampl("name", "Tyr", "age", 19));
9  }
10 }
11
12 // Superhero
13
14 interface Flyer {
15     // Superhero
16     String takeOff();
17     // Superhero
18     String land();
19     // Superhero
20     String fly();
21 }
22
23 // Superhero
24
25 class Animal {
26     // Superhero
27     String eat(String food) {
28         return "Hahaha! Eating " + food;
29     }
30 }

```

```
File Edit View Settings Code Refactor Build Run Tools Windows Help yhs.praktika - yhs.praktika.Net.java
yhs.praktika.Net.java yhs.praktika.Net.java yhs.praktika.Net.java yhs.praktika.Net.java

// Example 1: Interface
10 class Kuman {
11     // Example
12     string takoff(String name, int age) {
13         return "Nama : " + name + " Umur : " + age;
14     }
15 }

// Example
16 class Airplane extends Vehicle implements Flyer {
17     // Example
18     @Override
19     public String takoff() {
20         return "Pesawat lepas landas dari Bandara Kuala Selat ke Bandara Jember";
21     }
22     // Example
23     @Override
24     public String land() {
25         return "Pesawat mendarat di Bandara Jember";
26     }
27     // Example
28     @Override
29     public String fly() {
30         return "Pesawat Terbang Terbang ke Jember";
31     }
32 }

// Version Control Run 1000 Properties Terminal Run Build
Build completed successfully in 15 sec, 173 ms (2 minutes ago)
Type here to search 28°C 100% 15:43 30/10/2022
```

```
File Edit View Settings Code Refactor Build Run Tools Windows Help yhs.praktika - yhs.praktika.Net.java
yhs.praktika.Net.java yhs.praktika.Net.java yhs.praktika.Net.java yhs.praktika.Net.java

// Example
10 @Override
11 public String fly() {
12     return "Pesawat Terbang Terbang ke Jember";
13 }
14 // Example
15 class Bird extends Animal implements Flyer {
16     // Example
17     @Override
18     public String takoff() {
19         return "Burung lepas landas dari Bandara ROR Ponorogo (Situs Perumahan Lusi)";
20     }
21     // Example
22     @Override
23     public String land() {
24         return "Burung mendarat di Bandara ROR Ponorogo (Situs Perumahan Lusi)";
25     }
26     // Example
27     @Override
28     public String fly() {
29         return "Burung Terbang Terbang ke Jember";
30 }

// Version Control Run 1000 Properties Terminal Run Build
Build completed successfully in 15 sec, 173 ms (2 minutes ago)
Type here to search 28°C 100% 15:43 30/10/2022
```



```

C:\Users\User1> java -cp src\bin\java.exe -javaagent:C:\Program Files\JetBrains\IntelliJ IDEA (Community Edition 2022.2.2)\lib\idea_rt.jar=50701:C:\Program Files\JetBrains\IntelliJ IDEA (Community Edition 2022.2.2)\bin\java.exe
-- Pesawat --
Load : 10.0
Max Load : 100.0
Add Box : 10.0
Fuel Efficiency : 100.0
Trip Distance : 10000.0

# Pesawat lepas landas dari Bandara Kuala Namu ke Bandara Juanda
# Pesawat berhasil mendarat di Bandara Juanda
# Pesawat terbang selama 5 jam

-- Burung --
# Burung Elang terbang dari ketinggian 500 meter diatas permukaan laut
# Burung Puyuh mendarat tepat pada batang pohon mangga
# Burung Puyuh terbang selama 1 jam
# Burung Puyuh terbang dengan sarangnya di pohon mangga
# Burung Puyuh mengerami telurnya
Makanan Burung : Jagung

-- Superman --
# Superman terbang dengan sangat cepat menuju tempat kejadian perkara
# Superman mendarat di tempat kejadian perkara
# Superman terbang dengan kecepatan 110 km/jam
# Superman terbang melewati banyak gedung tinggi

Process finished with exit code 0

```

```

# Pesawat lepas landas dari Bandara Kuala Namu ke Bandara Juanda
# Pesawat berhasil mendarat di Bandara Juanda
# Pesawat terbang selama 5 jam

-- Burung --
# Burung Elang terbang dari ketinggian 500 meter diatas permukaan laut
# Burung Puyuh mendarat tepat pada batang pohon mangga
# Burung Puyuh terbang selama 1 jam
# Burung Puyuh terbang dengan sarangnya di pohon mangga
# Burung Puyuh mengerami telurnya
Makanan Burung : Jagung

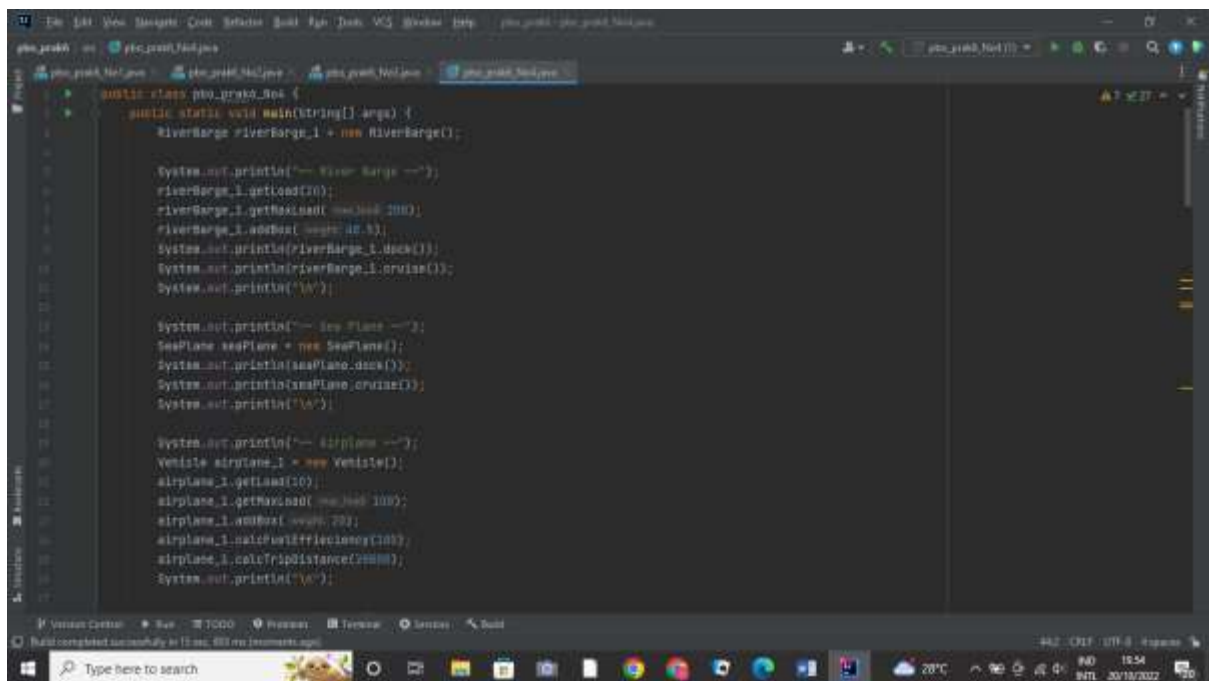
-- Superman --
# Superman terbang dengan sangat cepat menuju tempat kejadian perkara
# Superman mendarat di tempat kejadian perkara
# Superman terbang dengan kecepatan 110 km/jam
# Superman terbang melewati banyak gedung tinggi
# Superman tidak bisa ditembak peluru
Nama : Tya
Umur : 19

Process finished with exit code 0

```

Sama seperti kode program sebelumnya, kode diatas memiliki interface flyer yang memiliki 3 method yang sama. Kode program diatas terdapat 2 class tambahan yaitu class animal dan human. Berbeda dengan kode sebelumnya, class airplane, bird, dan superman mewarisi dari class induknya. Class airplane mewarisi class vehicle yang terdapat pada soal no.1. class bird mewarisi class animal dan class superman mewarisi class human. Inisialisasi dan hasil ditampilkan pada fungsi main.

4.



```

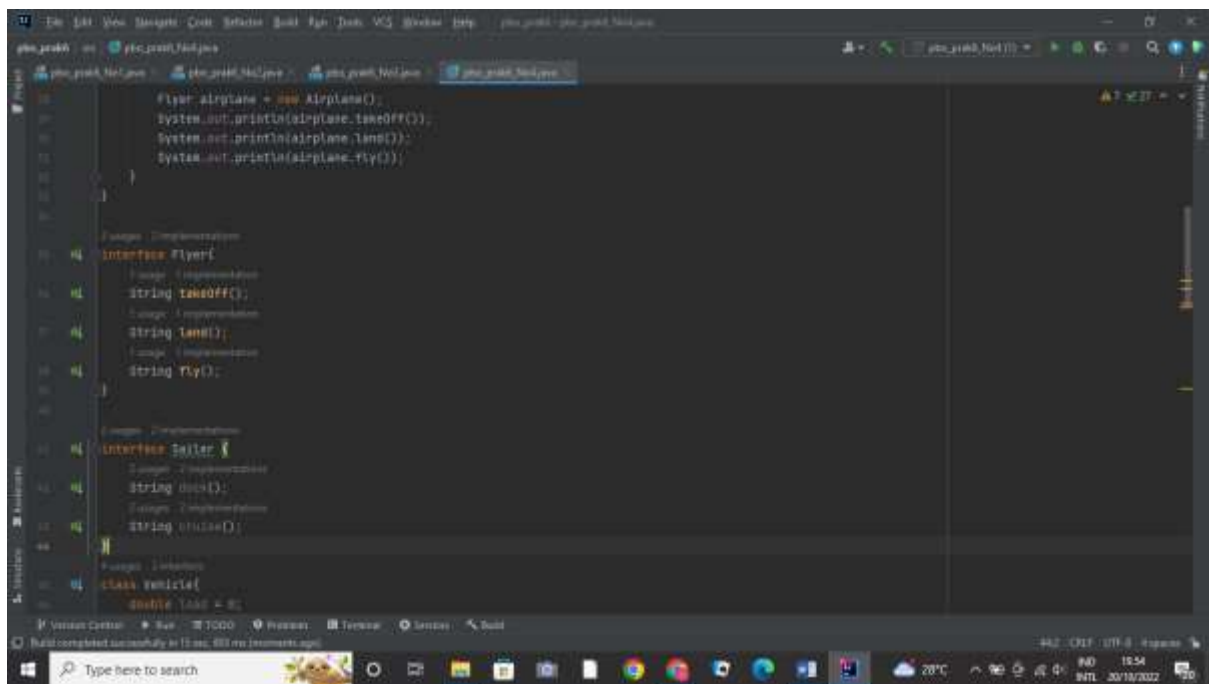
public class pto_prak0_Net {
    public static void main(String[] args) {
        RiverBarge riverBarge_1 = new RiverBarge();

        System.out.println("== River Barge ==");
        riverBarge_1.getLoad();
        riverBarge_1.getMaxLoad(1000000);
        riverBarge_1.addBox(1000000);
        System.out.println(riverBarge_1.docx());
        System.out.println(riverBarge_1.cruise());
        System.out.println("\n");

        System.out.println("== Sea Plane ==");
        SeaPlane seaPlane = new SeaPlane();
        System.out.println(seaPlane.docx());
        System.out.println(seaPlane.cruise());
        System.out.println("\n");

        System.out.println("== Airplane ==");
        Vehicle airplane_1 = new Vehicle();
        airplane_1.getLoad();
        airplane_1.getMaxLoad(1000000);
        airplane_1.addBox(1000000);
        airplane_1.calculateEfficiency(100);
        airplane_1.calculateDistance(1000000);
        System.out.println("\n");
    }
}

```



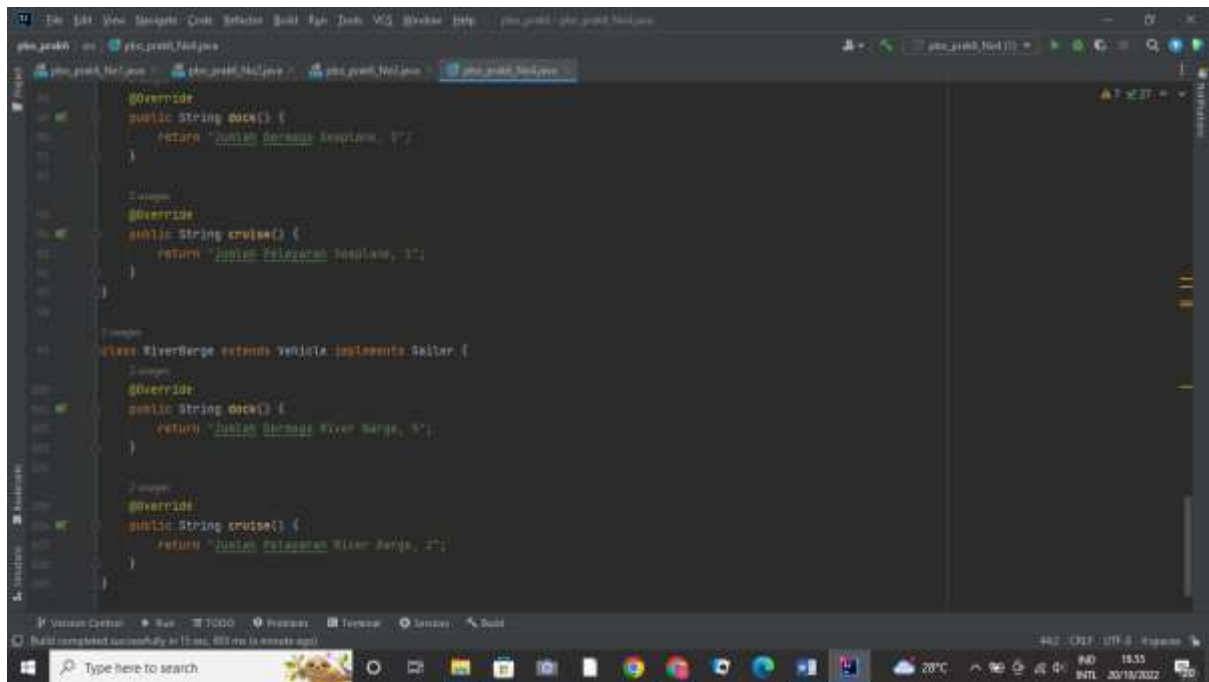
```

// Flyer - Implementation
public interface Flyer {
    // Flyer - Implementation
    String takeoff();
    // Flyer - Implementation
    String land();
    // Flyer - Implementation
    String fly();
}

// Seller - Implementation
public interface Seller {
    // Seller - Implementation
    String docx();
    // Seller - Implementation
    String cruise();
}

// Vehicle - Implementation
public class Vehicle {
    public void addBox() {
    }
}

```

```

@Override
public String dock() {
    return "Jumlah Dermaga Seaplane, 3";
}

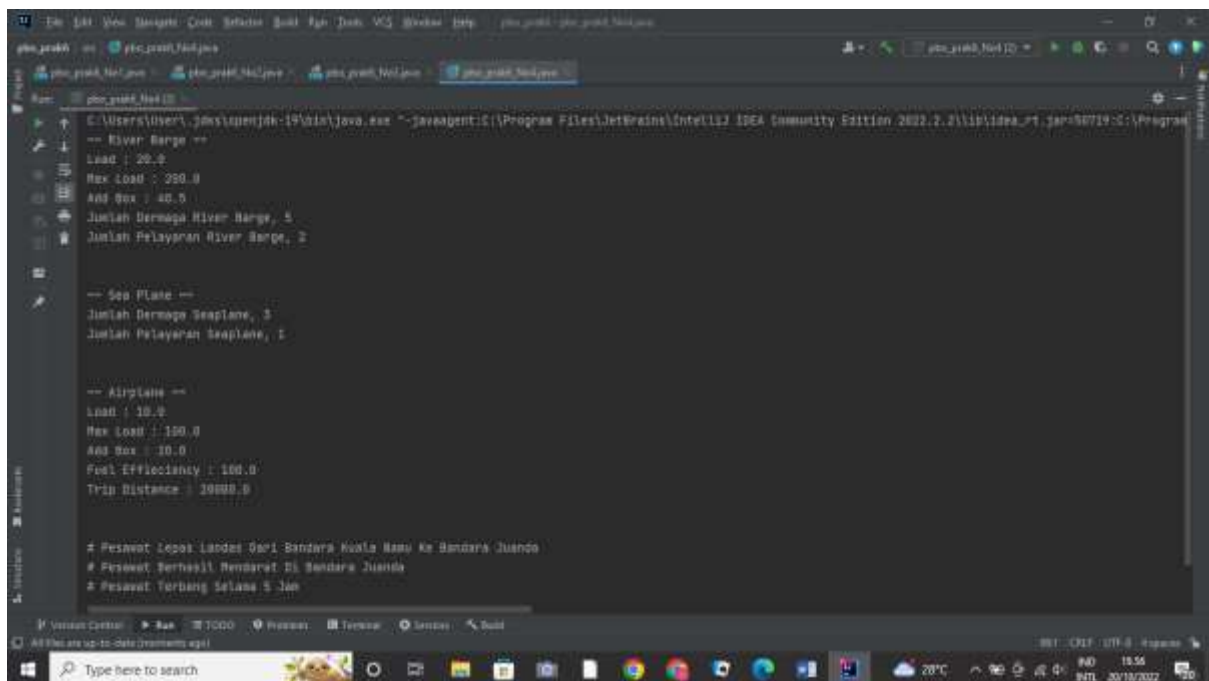
// Seaplane
@Override
public String cruise() {
    return "Jumlah Pelayaran Seaplane, 1";
}

// RiverBarge extends Vehicle implements Sailer
@Override
public String dock() {
    return "Jumlah Dermaga River Barge, 5";
}

@Override
public String cruise() {
    return "Jumlah Pelayaran River Barge, 2";
}

```

HASIL



```

C:\Users\user1\Documents\19\bin\java.exe -javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2022.2.2\lib\idea_rt.jar=50719-C:\Program
-- River Barge --
Load : 20.0
Max Load : 250.0
Add Box : 40.0
Jumlah Dermaga River Barge, 5
Jumlah Pelayaran River Barge, 2

-- Sea Plane --
Jumlah Dermaga Seaplane, 3
Jumlah Pelayaran Seaplane, 1

-- Airplane --
Load : 10.0
Max Load : 150.0
Add Box : 10.0
Fuel Efficiency : 100.0
Trip Distance : 10000.0

* Pesawat Lepas Landas Dari Bandara Kuala Namu Ke Bandara Juanda
* Pesawat Berhasil Mendarat Di Bandara Juanda
* Pesawat Terbang Selama 5 Jam

```

Kode program diatas memiliki 2 interface yaitu flyer dan sailer. Class vehicle memiliki 2 variabel dan 5 method yang nantinya akan digunakan pada class turunannya. Class airplane, seaplane, riverbarge sama-sama mewarisi class induknya dan mengimplementasi class interface yang telah dibuat. Inisialisasi class dan pengiriman data pada method terdapat di fungsi main.