TUGAS PRAKTIKUM 6
PEMROGRAMAN BERORIENTASI OBJEK

Oleh :

Reiznu Ahmad Tjandrida
21091397018
2021B

PROGRAM STUDI MANAJEMEN INFORMATIKA FAKULTAS VOKASI

UNIVERSITAS NEGERI SURABAYA

2022

No 1

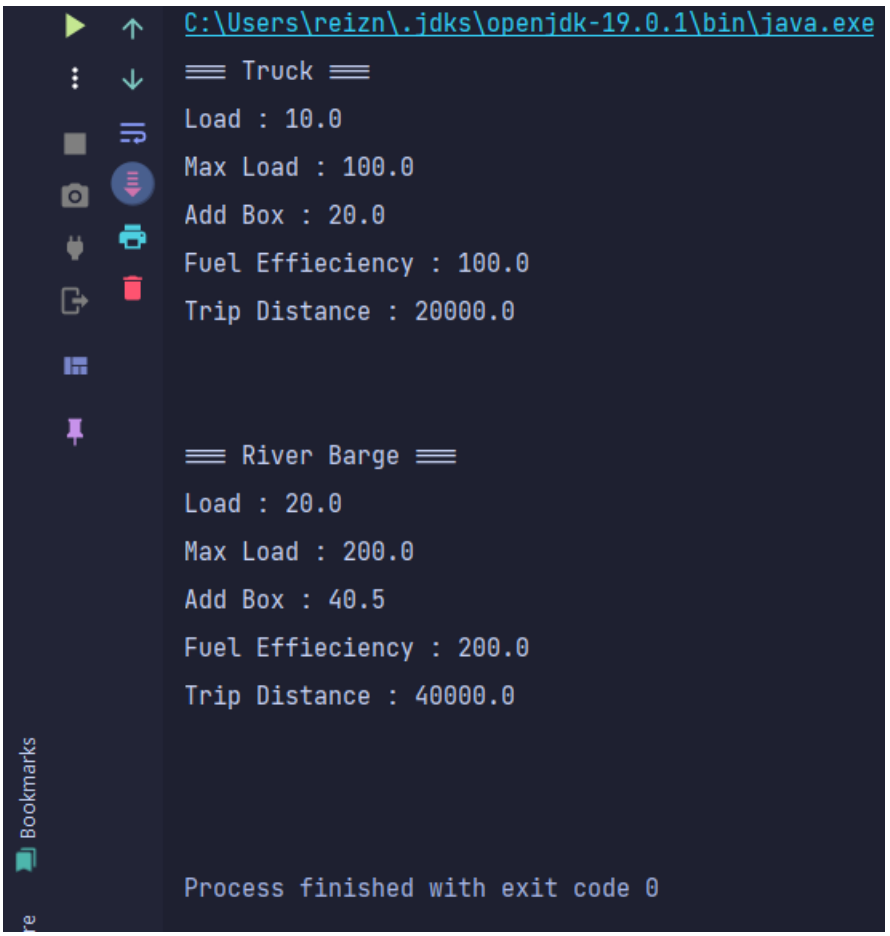
Kode Program

```
public class praktikum_6_no_1 {
    public static void main(String[] args) {
        Truck truck_1 = new Truck();
        RiverBarge riverBarge_1 = new RiverBarge();

        System.out.println("=== Truck ===");
        truck_1.getLoad(10);
        truck_1.getMaxLoad(100);
        truck_1.addBox(20);
        truck_1.calcFuelEffieciency(100);
        truck_1.calcTripDistance(20000);
        System.out.println("\n");

        System.out.println("=== River Barge ===");
        riverBarge_1.getLoad(20);
        riverBarge_1.getMaxLoad(200);
        riverBarge_1.addBox(40.5);
        riverBarge_1.calcFuelEffieciency(200);
        riverBarge_1.calcTripDistance(40000);
        System.out.println("\n");

    }
}

class Vehicle{
    double load = 0;
    double max_load = 0;

    void getLoad(double load) {
        System.out.println("Load : " + load);
    }
    void getMaxLoad(double max_load) {
        System.out.println("Max Load : " + max_load);
    };

    void addBox(double weight) {
        System.out.println("Add Box : " + weight);
    };

    public void calcFuelEffieciency(double fuel) {
        System.out.println("Fuel Effieciency : " + fuel);
    }

    void calcTripDistance(double trip) {
        System.out.println("Trip Distance : " + trip);
    }
}

class Truck extends Vehicle {
    public void calcFuelEffieciency(double fuel) {
        System.out.println("Fuel Effieciency : " + fuel);
    }

    void calcTripDistance(double trip) {
        System.out.println("Trip Distance : " + trip);
    }
}

class RiverBarge extends Vehicle {
    public void calcFuelEffieciency(double fuel) {
```

```
        System.out.println("Fuel Effieciency : " + fuel);
    }

    void calcTripDistance(double trip) {
        System.out.println("Trip Distance : " + trip);
    }
}
```

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 ***calcFuelEffieciency*** dan ***calcTripDistance***. Kemudian, method milik class truck dan riverBarge dipanggil difungsi main.

Hasil



No 2

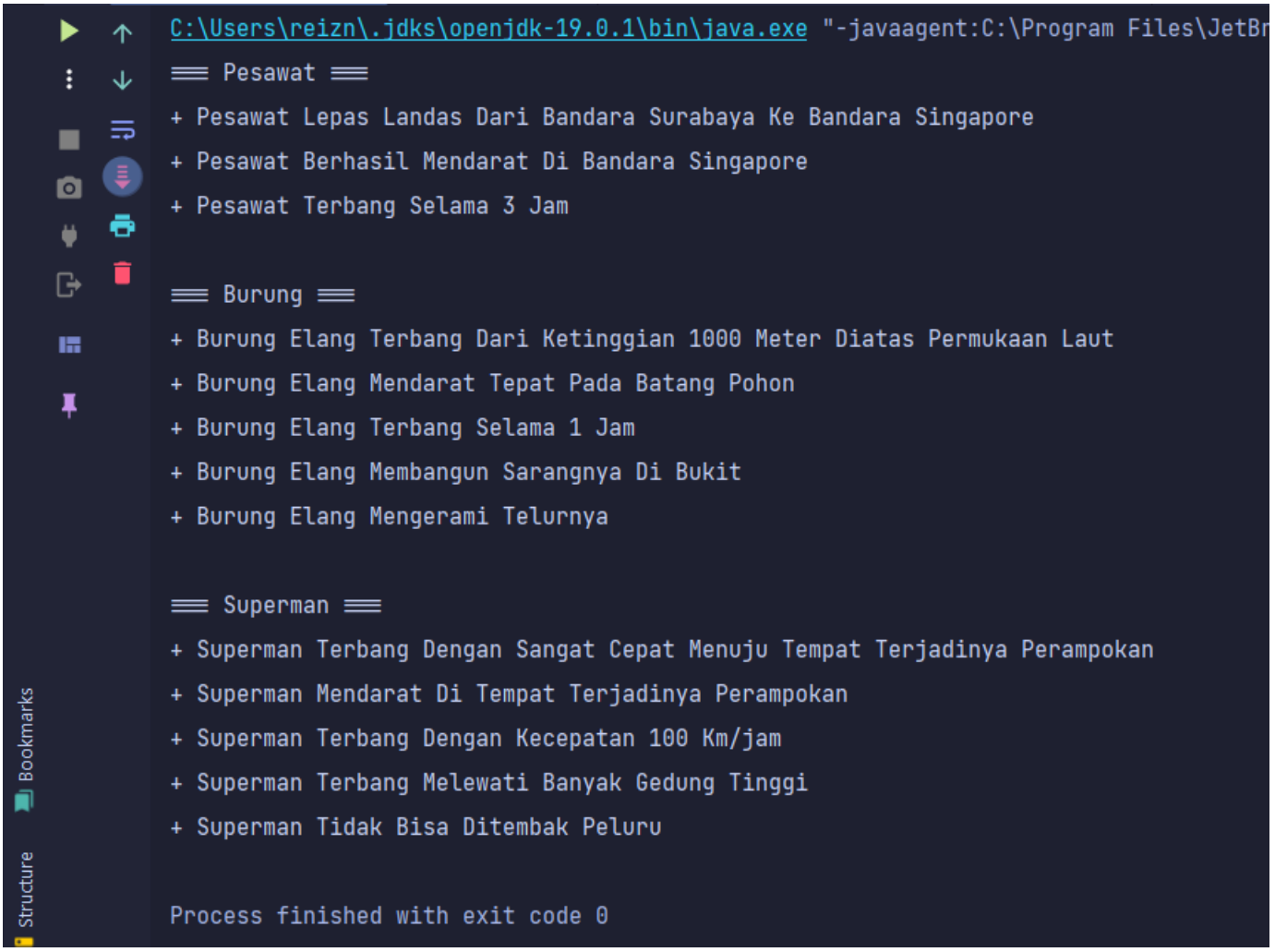
Kode Program

```
public class praktikum_6_no_2 {
    public static void main(String[] args) {

        System.out.println("=== Pesawat ===");
        Flyer airplane = new Airplane();
        System.out.println(airplane.takeOff());
        System.out.println(airplane.land());
        System.out.println(airplane.fly());

        System.out.println("\n=== Burung ===");
        Flyer bird = new Bird();
        System.out.println(bird.takeOff());
        System.out.println(bird.land());
        System.out.println(bird.fly());
        System.out.println(((Bird) bird).buildNest());
        System.out.println(((Bird) bird).layEggs());

        System.out.println("\n=== Superman ===");
        Flyer superman = new Superman();
        System.out.println(superman.takeOff());
        System.out.println(superman.land());
        System.out.println(superman.fly());
        System.out.println(((Superman) superman).leapBuilding());
        System.out.println(((Superman) superman).stopBullet());

    }
}

interface Flyer{
    String takeOff();
    String land();
    String fly();
}

class Airplane implements Flyer {

    @Override
    public String takeOff() {
        return "+ Pesawat Lepas Landas Dari Bandara Surabaya Ke Bandara Singapore";
    }

    @Override
    public String land() {
        return "+ Pesawat Berhasil Mendarat Di Bandara Singapore";
    }

    @Override
    public String fly() {
        return "+ Pesawat Terbang Selama 3 Jam";
    }
}

class Bird implements Flyer{

    @Override
    public String takeOff() {
        return "+ Burung Elang Terbang Dari Ketinggian 1000 Meter Diatas Permukaan Laut";
    }
```

```
    @Override
    public String land() {
        return "+ Burung Elang Mendarat Tepat Pada Batang Pohon";
    }

    @Override
    public String fly() {
        return "+ Burung Elang Terbang Selama 1 Jam";
    }

    public String buildNest() {
        return "+ Burung Elang Membangun Sarangnya Di Bukit";
    }

    public String layEggs() {
        return "+ Burung Elang Mengerami Telurnya";
    }
}

class Superman implements Flyer {

    @Override
    public String takeOff() {
        return "+ Superman Terbang Dengan Sangat Cepat Menuju Tempat Terjadinya Perampokan";
    }

    @Override
    public String land() {
        return "+ Superman Mendarat Di Tempat Terjadinya Perampokan";
    }

    @Override
    public String fly() {
        return "+ Superman Terbang Dengan Kecepatan 100 Km/jam";
    }

    public String leapBuilding() {
        return "+ Superman Terbang Melewati Banyak Gedung Tinggi";
    }

    public String stopBullet() {
        return "+ Superman Tidak Bisa Ditembak Peluru";
```

Pada kode program diatas terdapat 1 interface yang bernama flyer. Interface flyer memiliki 3 method yang nantinya dapat di implementasikan pada kelas yang membutuhkannya. Terdapat class airplane, bird, superman yang mengimplementasikan interface flyer, masing-masing classnya mengoverride method interfacenya. Fungsi main terdapat sintaks untuk menginisiasi class airplane, bird, supeman dan menampilkan hasilnya.

Hasil


C:\Users\reizn\.jdks\openjdk-19.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBr
Pesawat
+ Pesawat Lepas Landas Dari Bandara Surabaya Ke Bandara Singapore
+ Pesawat Berhasil Mendarat Di Bandara Singapore
+ Pesawat Terbang Selama 3 Jam
Burung
+ Burung Elang Terbang Dari Ketinggian 1000 Meter Diatas Permukaan Laut
+ Burung Elang Mendarat Tepat Pada Batang Pohon
+ Burung Elang Terbang Selama 1 Jam
+ Burung Elang Membangun Sarangnya Di Bukit
+ Burung Elang Mengerami Telurnya
Superman
+ Superman Terbang Dengan Sangat Cepat Menuju Tempat Terjadinya Perampokan
+ Superman Mendarat Di Tempat Terjadinya Perampokan
+ Superman Terbang Dengan Kecepatan 100 Km/jam
+ Superman Terbang Melewati Banyak Gedung Tinggi
+ Superman Tidak Bisa Ditembak Peluru
Process finished with exit code 0
Bookmarks
Structure

No 3

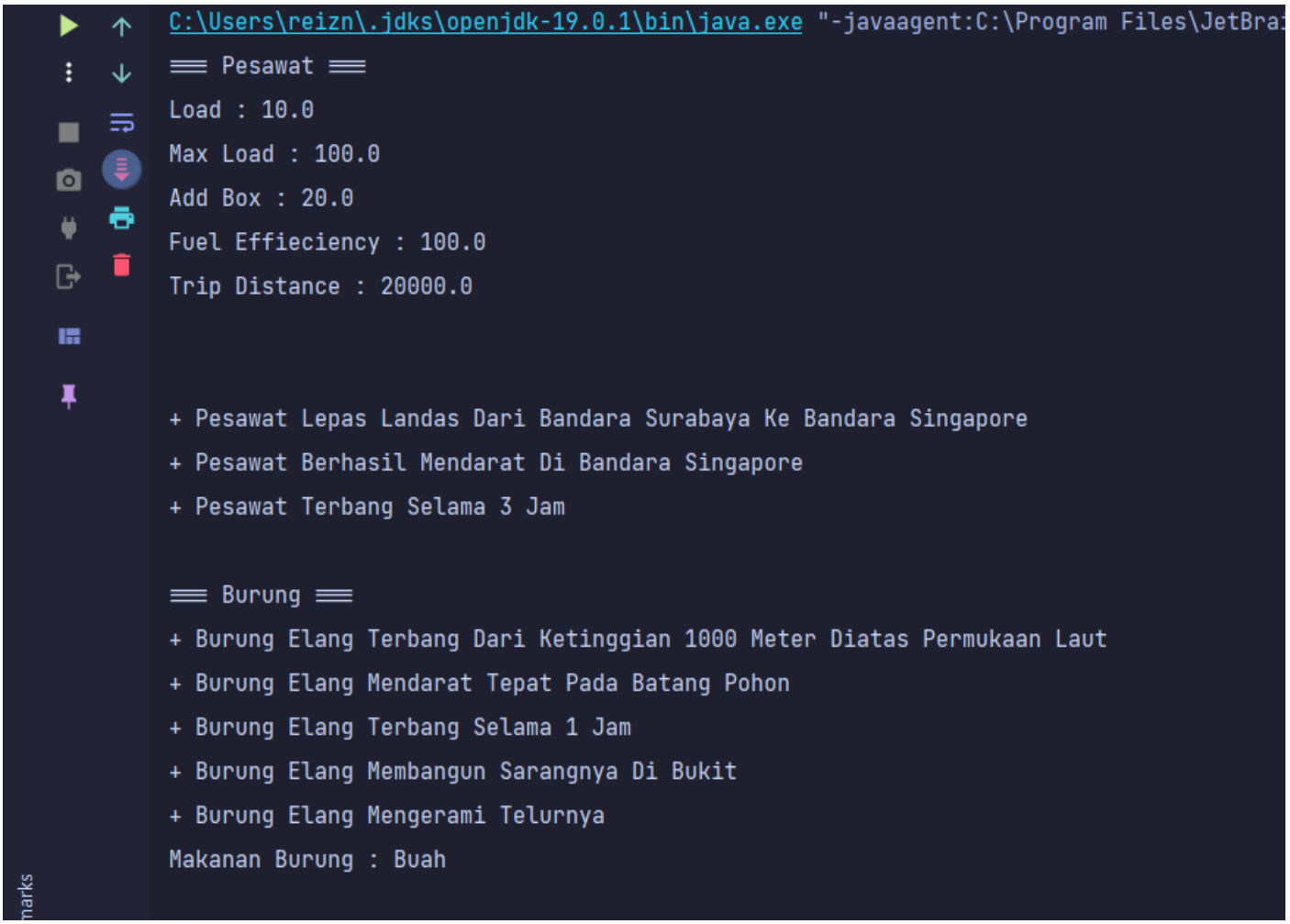
Kode Program

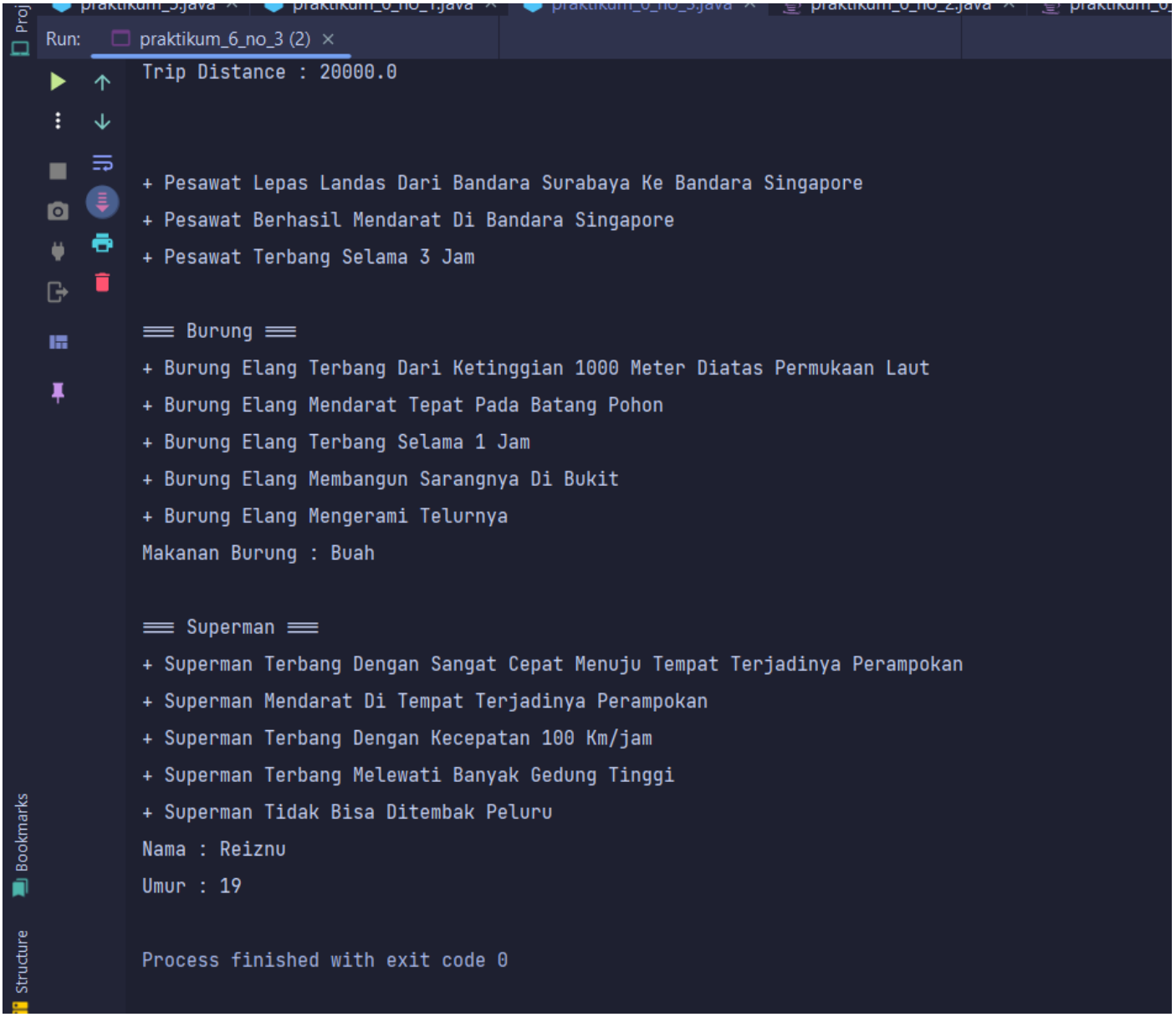
```
public class praktikum_6_no_3 {
    public static void main(String[] args) {

        System.out.println("=== Pesawat ===");
        Vehicle airplane_1 = new Vehicle();
        airplane_1.getLoad(10);
        airplane_1.getMaxLoad(100);
        airplane_1.addBox(20);
        airplane_1.calcFuelEffieciency(100);
        airplane_1.calcTripDistance(20000);
        System.out.println("\n");

        Flyer airplane = new Airplane();
        System.out.println(airplane.takeOff());
        System.out.println(airplane.land());
        System.out.println(airplane.fly());

        System.out.println("\n=== Burung ===");
        Bird bird = new Bird();
        System.out.println(bird.takeOff());
        System.out.println(bird.land());
        System.out.println(bird.fly());
        System.out.println(bird.buildNest());
        System.out.println(bird.layEggs());
        System.out.println(bird.eat("Buah"));

        System.out.println("\n=== Superman ===");
        Superman superman = new Superman();
        System.out.println(superman.takeOff());
        System.out.println(superman.land());
        System.out.println(superman.fly());
        System.out.println(superman.leapBuilding());
        System.out.println(superman.stopBullet());
        System.out.println(superman.tampil("Reiznu", 19));
    }
}

interface Flyer{
    String takeOff();
    String land();
    String fly();
}

class Animal {
    String eat(String food) {
        return "Makanan Burung : " + food;
    }
}

class Human {
    String tampil(String name, int age) {
        return "Nama : " + name + "\nUmur : " + age;
    }
}

class Airplane extends Vehicle implements Flyer {

    @Override
    public String takeOff() {
        return "+ Pesawat Lepas Landas Dari Bandara Surabaya Ke Bandara Singapore";
```

```
    }

    @Override
    public String land() {
        return "+ Pesawat Berhasil Mendarat Di Bandara Singapore";
    }

    @Override
    public String fly() {
        return "+ Pesawat Terbang Selama 3 Jam";
    }
}

class Bird extends Animal implements Flyer{

    @Override
    public String takeOff() {
        return "+ Burung Elang Terbang Dari Ketinggian 1000 Meter Diatas Permukaan Laut";
    }

    @Override
    public String land() {
        return "+ Burung Elang Mendarat Tepat Pada Batang Pohon";
    }

    @Override
    public String fly() {
        return "+ Burung Elang Terbang Selama 1 Jam";
    }

    public String buildNest() {
        return "+ Burung Elang Membangun Sarangnya Di Bukit";
    }

    public String layEggs() {
        return "+ Burung Elang Mengerami Telurnya";
    }
}

class Superman extends Human implements Flyer {

    @Override
    public String takeOff() {
        return "+ Superman Terbang Dengan Sangat Cepat Menuju Tempat Terjadinya Perampokan";
    }

    @Override
    public String land() {
        return "+ Superman Mendarat Di Tempat Terjadinya Perampokan";
    }

    @Override
    public String fly() {
        return "+ Superman Terbang Dengan Kecepatan 100 Km/jam";
    }

    public String leapBuilding() {
        return "+ Superman Terbang Melewati Banyak Gedung Tinggi";
    }

    public String stopBullet() {
        return "+ Superman Tidak Bisa Ditembak Peluru";
    }
}
```

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.

Hasil




praktikum_6_no_3 (2)
Trip Distance : 20000.0
+ Pesawat Lepas Landas Dari Bandara Surabaya Ke Bandara Singapore
+ Pesawat Berhasil Mendarat Di Bandara Singapore
+ Pesawat Terbang Selama 3 Jam
Burung
+ Burung Elang Terbang Dari Ketinggian 1000 Meter Diatas Permukaan Laut
+ Burung Elang Mendarat Tepat Pada Batang Pohon
+ Burung Elang Terbang Selama 1 Jam
+ Burung Elang Membangun Sarangnya Di Bukit
+ Burung Elang Mengerami Telurnya
Makanan Burung : Buah
Superman
+ Superman Terbang Dengan Sangat Cepat Menuju Tempat Terjadinya Perampokan
+ Superman Mendarat Di Tempat Terjadinya Perampokan
+ Superman Terbang Dengan Kecepatan 100 Km/jam
+ Superman Terbang Melewati Banyak Gedung Tinggi
+ Superman Tidak Bisa Ditembak Peluru
Nama : Reiznu
Umur : 19
Process finished with exit code 0
Bookmarks
Structure

No 4

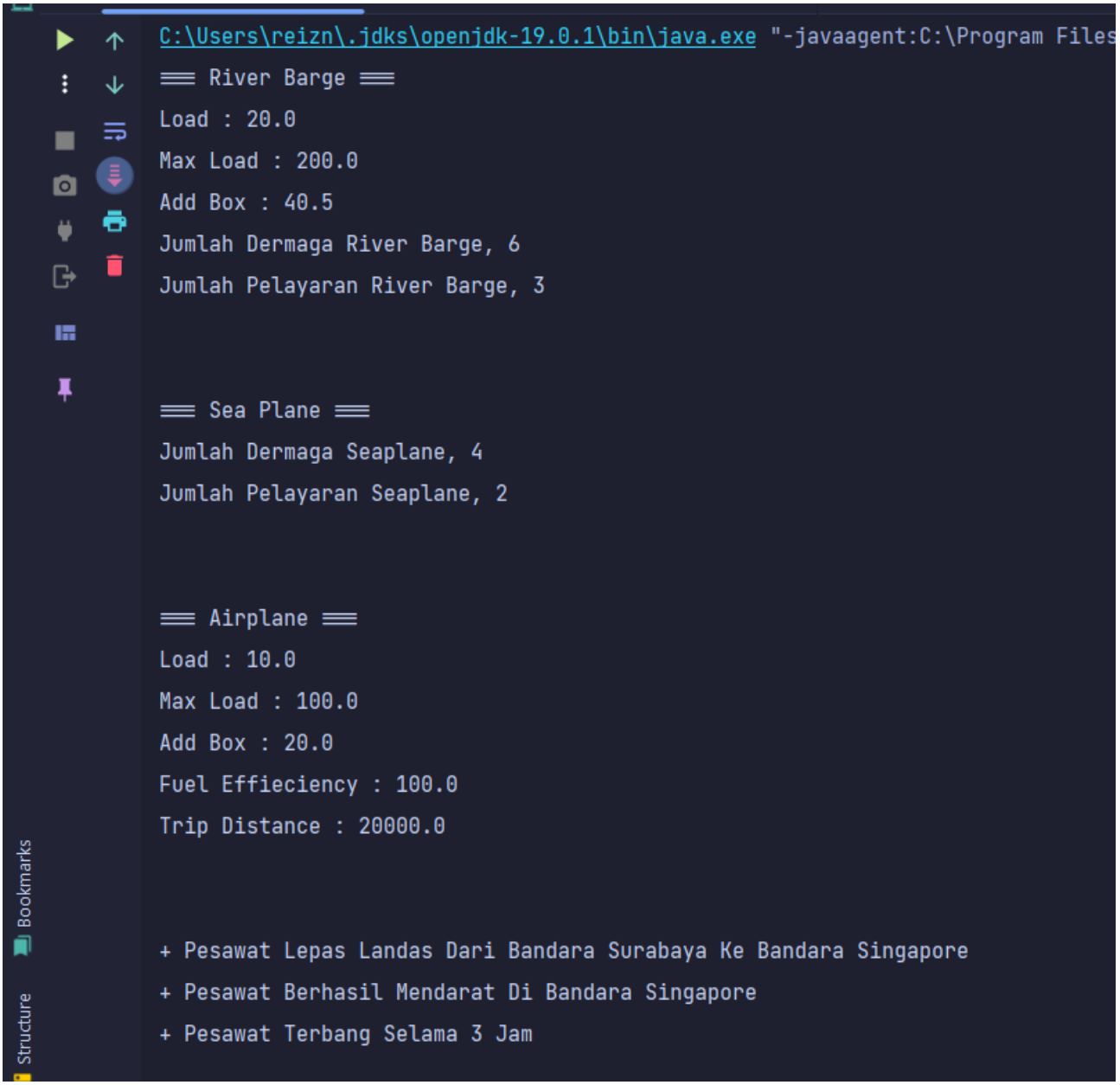
Kode Program

```
public class praktikum_6_no_4 {
    public static void main(String[] args) {
        RiverBarge riverBarge_1 = new RiverBarge();

        System.out.println("=== River Barge ===");
        riverBarge_1.getLoad(20);
        riverBarge_1.getMaxLoad(200);
        riverBarge_1.addBox(40.5);
        System.out.println(riverBarge_1.dock());
        System.out.println(riverBarge_1.cruise());
        System.out.println("\n");

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

        System.out.println("=== Airplane ===");
        Vehicle airplane_1 = new Vehicle();
        airplane_1.getLoad(10);
        airplane_1.getMaxLoad(100);
        airplane_1.addBox(20);
        airplane_1.calcFuelEffieciency(100);
        airplane_1.calcTripDistance(20000);
        System.out.println("\n");

        Flyer airplane = new Airplane();
        System.out.println(airplane.takeOff());
        System.out.println(airplane.land());
        System.out.println(airplane.fly());
    }
}

interface Flyer{
    String takeOff();
    String land();
    String fly();
}

interface Sailer {
    String dock();
    String cruise();
}
class Vehicle{
    double load = 0;
    double max_load = 0;

    void getLoad(double load) {
        System.out.println("Load : " + load);
    }
    void getMaxLoad(double max_load) {
        System.out.println("Max Load : " + max_load);
    };

    void addBox(double weight) {
        System.out.println("Add Box : " + weight);
    };

    public void calcFuelEffieciency(double fuel) {
```

```
        System.out.println("Fuel Effieciency : " + fuel);
    }

    void calcTripDistance(double trip) {
        System.out.println("Trip Distance : " + trip);
    }
}

class Airplane extends Vehicle implements Flyer {

    @Override
    public String takeOff() {
        return "+ Pesawat Lepas Landas Dari Bandara Surabaya Ke Bandara Singapore";
    }

    @Override
    public String land() {
        return "+ Pesawat Berhasil Mendarat Di Bandara Singapore";
    }

    @Override
    public String fly() {
        return "+ Pesawat Terbang Selama 3 Jam";
    }
}

class SeaPlane extends Airplane implements Sailer {
    @Override
    public String dock() {
        return "Jumlah Dermaga Seaplane, 4";
    }

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

class RiverBarge extends Vehicle implements Sailer {
    @Override
    public String dock() {
        return "Jumlah Dermaga River Barge, 6";
    }

    @Override
    public String cruise() {
        return "Jumlah Pelayaran River Barge, 3";
    }
}
```

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.

Hasil


C:\Users\reizn\.jdks\openjdk-19.0.1\bin\java.exe "-javaagent:C:\Program Files
== River Barge ==
Load : 20.0
Max Load : 200.0
Add Box : 40.5
Jumlah Dermaga River Barge, 6
Jumlah Pelayaran River Barge, 3
== Sea Plane ==
Jumlah Dermaga Seaplane, 4
Jumlah Pelayaran Seaplane, 2
== Airplane ==
Load : 10.0
Max Load : 100.0
Add Box : 20.0
Fuel Effieciency : 100.0
Trip Distance : 20000.0
+ Pesawat Lepas Landas Dari Bandara Surabaya Ke Bandara Singapore
+ Pesawat Berhasil Mendarat Di Bandara Singapore
+ Pesawat Terbang Selama 3 Jam
Bookmarks
Structure