

## RESPONSI PBO

NAMA : FIKRY MUMTAZ PRATAMA  
NIM : H1D024106  
SHIFT : SHIFT B  
TANGGAL : 07/12/2025

### JAWAB

#### 1. Class Utama: KendaraanGalaksi (Abstract Class)

Ini adalah **template dasar** untuk semua kendaraan.

Isi di dalamnya:

- Nama kendaraan
- Level energi (0–100)
- Kapasitas penumpang
- Method **tampilStatus()** → menampilkan energi + kapasitas
- Method abstract:
  - **aktifkanMesin()**
  - **jelajah(jarak)**
  - **isiEnergi(jumlah)**

Karena abstract:

- Tidak bisa dibuat objeknya.
- Hanya bisa diwarisi oleh class lain.

Tujuannya:

Semua jenis kendaraan galaksi **punya aturan dasar yang sama**, tetapi cara kerjanya beda-beda.

#### 2. KapalEksplorasi (extends KendaraanGalaksi)

Tambahan atribut:

- modulScan

Perilaku khusus:

- Mesin bisa aktif jika energi  $\geq 15$
- Menjelajah: konsumsi energi = jarak  $\times 2$
- Bisa scan planet

Ini adalah kendaraan yang fokus untuk survei dan eksplorasi.

### 3. PesawatTempur (extends KendaraanGalaksi)

Tambahan atribut:

- jumlahRudal

Perilaku khusus:

- Mesin aktif jika energi  $\geq 20$
- Menjelajah: energi habis 3% per km
- Bisa menembak rudal
- Cocok untuk operasi tempur

### 4. Alur Program di UjiGalaksi

Program dimulai di main().

#### A. Menampilkan header

=== UJI SISTEM KENDARAAN GALAKSI ===

#### B. Membuat objek PesawatTempur

PesawatTempur pesawat = new PesawatTempur("Astra-Fury", 2, 8);

Artinya:

- Nama: Astra-Fury
- Kapasitas: 2 orang
- Rudal: 8 buah
- Energi awal: 100%

Jalannya aksi:

1. pesawat.aktifkanMesin()  
→ Kalau energi di atas 20, mesin aktif.
  2. pesawat.jelajah(10)  
→ energi berkurang  $10 \times 3 = 30\%$
  3. pesawat.jelajah(30)  
→ energi berkurang  $30 \times 3 = 90\%$   
→ Jika energi tersisa tidak cukup, muncul pesan energi tidak mencukupi.
  4. pesawat.tembakRudal(3)  
→ mengurangi 3 rudal
  5. pesawat.tampilStatus()  
→ tampilkan energi yang tersisa + kapasitas penumpang
-

### C. Membuat objek KapalEksplorasi

KapalEksplorasi kapal = new KapalEksplorasi("Voyager X", 10, 4);

Atribut:

- Nama: Voyager X
- Kapasitas: 10
- Modul scan: 4
- Energi awal: 100%

Jalannya aksi:

1. kapal.aktifkanMesin()  
→ jika energi di atas 15, mesin hidup.
2. kapal.jelajah(15)  
→ energi berkurang  $15 \times 2 = 30\%$
3. kapal.scanPlanet("Kepler-442b")  
→ melakukan scanning planet
4. kapal.tampilStatus()  
→ tampilkan energi tersisa dan kapasitas

### KESIMPULAN ALUR

Secara singkat:

1. Ada **class dasar** KendaraanGalaksi berisi aturan umum.
2. Ada **dua turunan**:
  - KapalEksplorasi → fokus scan & eksplorasi
  - PesawatTempur → fokus tempur & rudal
3. Program membuat objek kedua kendaraan tersebut.
4. Setiap kendaraan dipanggil method-nya (aktifkanMesin, jelajah, dll).
5. Energi kendaraan berkurang sesuai aturan masing-masing class.
6. Di akhir, program menampilkan status energi dan kapasitas.

```

29
30     @Override    no usages
31     public void isiEnergi(int jumlah) {
32         setEnergi(getLevelEnergi() + jumlah);
33         System.out.println("Energi kapal bertambah.");
34     }
35
36     public void scanPlanet(String namaPlanet) { 1 usage
37         System.out.println("Melakukan scan pada planet " + namaPlanet +
38             " dengan modul level " + modulScan + ".");
39     }
40 }
41

```

KendaraanGalaksi.java
 KapalEksplorasi.java
 PesawatTempur.java
 UjiGalaksi.java

```

1  public class KapalEksplorasi extends KendaraanGalaksi { 3 usages
2
3      private int modulScan; 2 usages
4
5      public KapalEksplorasi(String nama, int kapasitas, int modulScan) { 2 usages
6          super(nama, kapasitas);
7          this.modulScan = modulScan;
8      }
9
10     @Override 2 usages
11     public void aktifkanMesin() {
12         if (getLevelEnergi() < 15) {
13             System.out.println("Energi tidak mencukupi untuk memulai ekspedisi!");
14         } else {
15             System.out.println("Kapal eksplorasi siap berangkat!");
16         }
17     }
18
19     @Override 3 usages
20     public void jelajah(int jarak) {
21         int konsumsi = jarak * 2; // 2% per km
22         if (getLevelEnergi() < konsumsi) {
23             System.out.println("Energi tidak cukup untuk menjelajah sejauh " + jarak + " km.");
24         } else {
25             setEnergi(getLevelEnergi() - konsumsi);
26             System.out.println("Kapal eksplorasi menjelajah sejauh " + jarak + " km.");
27         }
28     }
29

```

```

    }

    protected void setEnergi(int energiBaru) { 4 usages
        if (energiBaru < 0) energiBaru = 0;
        if (energiBaru > 100) energiBaru = 100;
        this.levelEnergi = energiBaru;
    }

    public final void tampilStatus() { 2 usages
        System.out.println(
            getNamaKendaraan() +
                " | Energi: " + getLevelEnergi() + "% | Kapasitas: " +
                getKapasitasPenumpang() + " orang"
        );
    }

    public abstract void aktifkanMesin(); 2 usages 2 implementations
    public abstract void jelajah(int jarak); 3 usages 2 implementations
    public abstract void isiEnergi(int jumlah); no usages 2 implementations
}

```

```

KendaraanGalaksi.java x PesawatTempur.java UjiGalaksi.java
1 public abstract class KendaraanGalaksi { 3 usages 2 inheritors
2
3     private String namaKendaraan; 2 usages
4     private int levelEnergi; // 0-100 3 usages
5     private int kapasitasPenumpang; 2 usages
6
7     public KendaraanGalaksi(String namaKendaraan, int kapasitasPenumpang) { 3 usages
8         this.namaKendaraan = namaKendaraan;
9         this.kapasitasPenumpang = kapasitasPenumpang;
10        this.levelEnergi = 100; // default 100%
11    }
12
13    public String getNamaKendaraan() { 1 usage
14        return namaKendaraan;
15    }
16
17    public int getLevelEnergi() { 9 usages
18        return levelEnergi;
19    }
20
21    public int getKapasitasPenumpang() { 1 usage
22        return kapasitasPenumpang;
23    }
24
25    protected void setEnergi(int energiBaru) { 4 usages
26        if (energiBaru < 0) energiBaru = 0;
27        if (energiBaru > 100) energiBaru = 100;
28        this.levelEnergi = energiBaru;
29    }

```

```
KendaraanGalaksi.java PesawatTempur.java x UjiGalaksi.java
1 public class PesawatTempur extends KendaraanGalaksi { 3 usages
20     public void jelajah(int jarak) {
24         } else {
25             setEnergi(getLevelEnergi() - konsumsi);
26             System.out.println("Pesawat tempur menjelajah sejauh " + jarak + " km.");
27         }
28     }
29
30     @Override no usages
31     public void isiEnergi(int jumlah) {
32         setEnergi(getLevelEnergi() + jumlah);
33         System.out.println("Energi pesawat bertambah.");
34     }
35
36     public void tembakRudal(int jumlah) { 1 usage
37         if (jumlahRudal >= jumlah) {
38             jumlahRudal -= jumlah;
39             System.out.println("Menembakkan " + jumlah + " rudal!");
40         } else {
41             System.out.println("Rudal tidak mencukupi!");
42         }
43     }
44 }
45
```

```
KendaraanGalaksi.java PesawatTempur.java x UjiGalaksi.java
1 public class PesawatTempur extends KendaraanGalaksi { 3 usages
2
3     private int jumlahRudal; 3 usages
4
5     public PesawatTempur(String nama, int kapasitas, int jumlahRudal) { 2 usages
6         super(nama, kapasitas);
7         this.jumlahRudal = jumlahRudal;
8     }
9
10    @Override 2 usages
11    public void aktifkanMesin() {
12        if (getLevelEnergi() < 0) {
13            System.out.println("Energi pesawat tidak dapat diaktifkan.");
14        } else {
15            System.out.println("Mesin pesawat diaktifkan.");
16        }
17    }
18
19    @Override 3 usages
20    public void jelajah(int jarak) {
21        int konsumsi = jarak * 3; // 3% per km
22        if (getLevelEnergi() < konsumsi) {
23            System.out.println("Energi tidak mencukupi untuk menjelajah sejauh " + jarak + " km.");
24        } else {
25            setEnergi(getLevelEnergi() - konsumsi);
26            System.out.println("Pesawat tempur menjelajah sejauh " + jarak + " km.");
27        }
28    }
29
```

Ⓢ PesawatTempur

public void aktifkanMesin()

Specified by: aktifkanMesin in class KendaraanGalaksi

📄 pertemuan 6 kendaraan 🔗 ⋮

```
KendaraanGalaksi.java  PesawatTempur.java  UjiGalaksi.java x
1 public class UjiGalaksi {
2     public static void main(String[] args) {
3
4         System.out.println("=== UJI SISTEM KENDARAAN GALAKSI ===\n");
5
6         // PESAWAT TEMPUR
7         PesawatTempur pesawat = new PesawatTempur( nama: "Astra-Fury", kapasitas: 2, jumlahRudal: 8);
8
9         System.out.println("--- PESAWAT TEMPUR ---");
10        pesawat.aktifkanMesin();
11        pesawat.jelajah( jarak: 10);
12        pesawat.jelajah( jarak: 30);
13        pesawat.tembakRudal( jumlah: 3);
14        pesawat.tampilStatus();
15
16        // KAPAL EKSPLORASI
17        System.out.println("\n--- KAPAL EKSPLORASI ---");
18        KapalEksplorasi kapal = new KapalEksplorasi( nama: "Voyager X", kapasitas: 10, modulScan: 4);
19
20        kapal.aktifkanMesin();
21        kapal.jelajah( jarak: 15);
22        kapal.scanPlanet( namaPlanet: "Kepler-442b");
23        kapal.tampilStatus();
24    }
25 }
26
```

"C:\Program Files\Eclipse Adoptium\jdk-21.0.7-hotspot\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2025.2.1\lib\idea\_rt.jar=62659" -Dfile.encoding=UTF-8

```
=== UJI SISTEM KENDARAAN GALAKSI ===

--- PESAWAT TEMPUR ---
Mesin pesawat tempur diaktifkan.
Pesawat tempur menjelajah sejauh 10 km.
Energi tidak mencukupi untuk menjelajah sejauh 30 km.
Menembakkan 3 rudal!
Astra-Fury | Energi: 70% | Kapasitas: 2 orang

--- KAPAL EKSPLORASI ---
Kapal eksplorasi siap berangkat!
Kapal eksplorasi menjelajah sejauh 15 km.
Melakukan scan pada planet Kepler-442b dengan modul level 4.
Voyager X | Energi: 70% | Kapasitas: 10 orang

Process finished with exit code 0
|
```