

Responsi 2 Praktikum PBO - Pertemuan 6 (Abstract Class Kendaraan Galaksi)

A. Alur Kerja

1. Membuat abstract class KendaraanGalaksi dengan atribut: namaKendaraan, levelEnergi, kapasitasPenumpang
2. Membuat constructor KendaraanGalaksi(String, int)
3. Membuat method konkret tampilStatus() dan abstract method: aktifkanMesin(), jelajah(int), isiEnergi(int)
4. Membuat subclass PesawatTempur dan KapalEksplorasi
5. Implementasikan semua abstract method + tambahan method khusus (tembakRudal, scanPlanet)
6. Buat class UjiGalaksi untuk:
 - Instansiasi objek pesawat & kapal
 - Panggil aktifkanMesin(), jelajah(...), tembakRudal / scanPlanet
 - Tampilkan status akhir
7. Jalankan program

B. Fungsi yang Digunakan

1. KendaraanGalaksi(String, int) – constructor parent
2. aktifkanMesin() – cek energi & cetak pesan khusus tiap kendaraan
3. jelajah(int jarak) – kurangi energi sesuai konsumsi tiap kendaraan
4. isiEnergi(int) – tambah energi (maks 100)
5. tampilStatus() – cetak nama, energi, kapasitas (final method)
6. tembakRudal(int) – kurangi rudal (khusus PesawatTempur)
7. scanPlanet(String) – cetak hasil scan (khusus KapalEksplorasi)

C. Output Program

```
--- PESAWAT TEMPUR ---
Mesin pesawat tempur diaktifkan.
Pesawat tempur menjelajah sejauh 10 km.
Pesawat tempur menjelajah sejauh 30 km.
Menembakkan 3 rudal!
Astra-Fury | Energi: 100% | Kapasitas: 2awak

--- 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: 100% | Kapasitas: 10awak

Process finished with exit code 0
```

--- PESAWAT TEMPUR ---

Mesin pesawat tempur diaktifkan.
Pesawat tempur menjelajah sejauh 10 km.
Pesawat tempur menjelajah sejauh 30 km.
Menembakkan 3 rudal!
Astra-Fury | Energi: 100% | Kapasitas: 2awak

--- 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: 100% | Kapasitas: 10awak

D. Kesimpulan

Program ini berhasil menampilkan output Abstract Class Kendaraan Galaksi. Abstract class memaksa subclass untuk mengimplementasikan perilaku khusus. Konsep encapsulation dan override membuat kode terstruktur. Sistem kendaraan menjadi mudah dikembangkan untuk jenis baru.