LAPORAN PRAKTIKUM

POSTTEST 5

Pemrograman Berorientasi Objek

Disusun oleh:

Brayen Pranajaya Kesuma

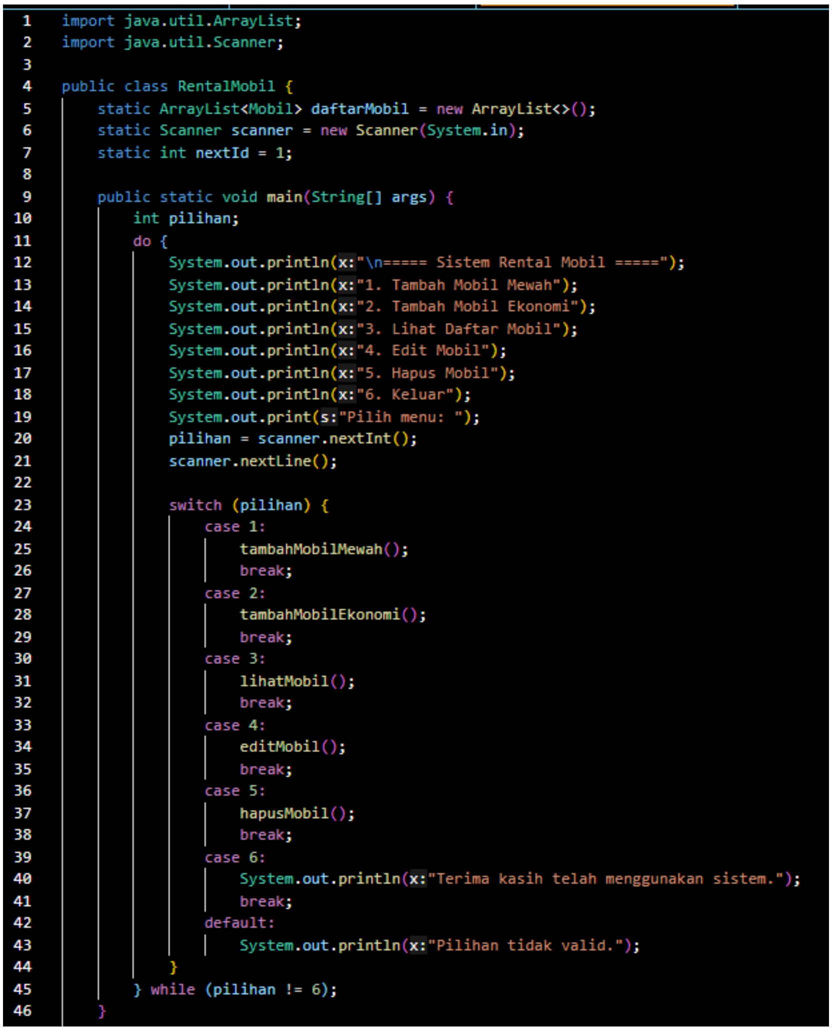
2309106128

C2

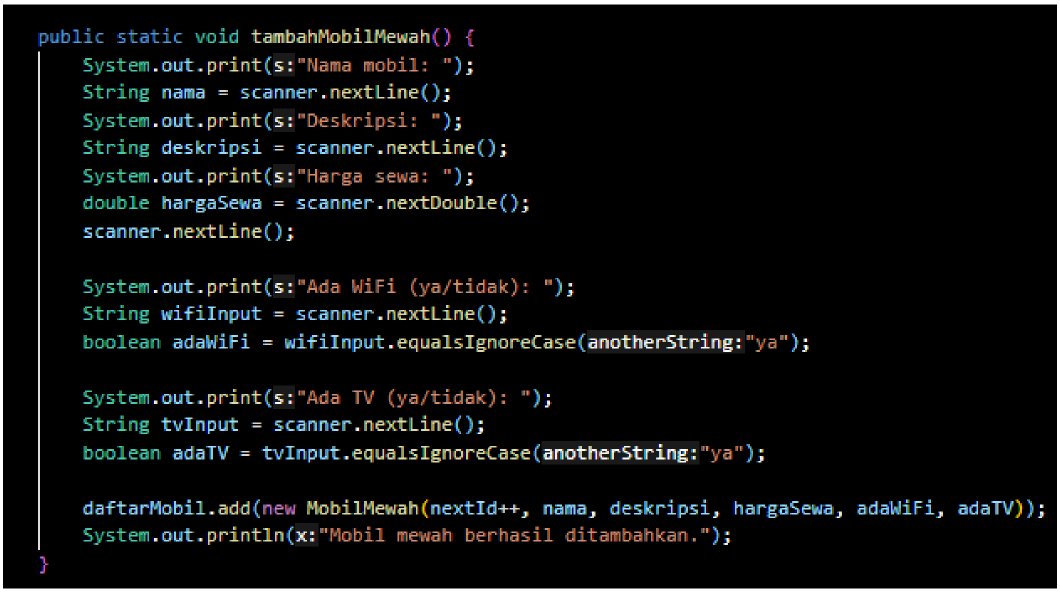
PROGRAM STUDI INFORMATIKA

UNIVERSITAS MULAWARMAN SAMARINDA

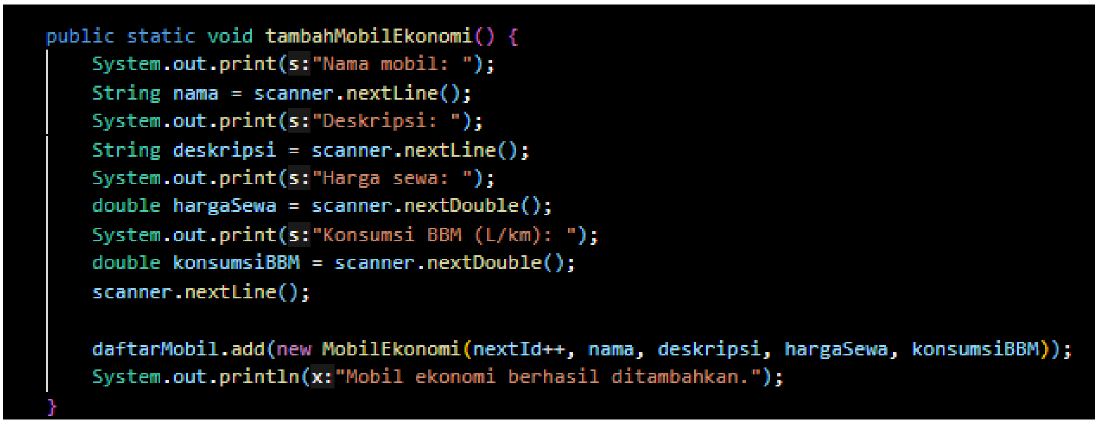
2025



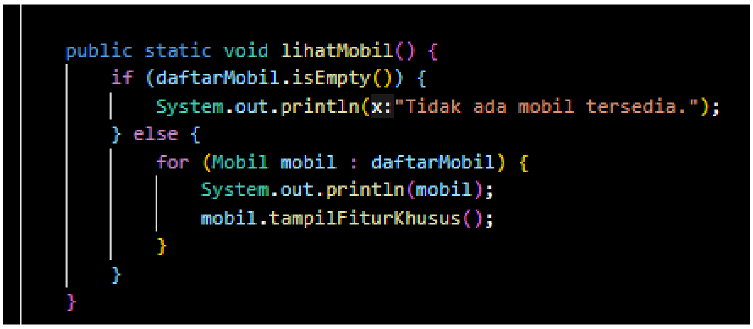
```java
import java.util.ArrayList;
import java.util.Scanner;

public class RentalMobil {
    static ArrayList<Mobil> daftarMobil = new ArrayList<>();
    static Scanner scanner = new Scanner(System.in);
    static int nextId = 1;

    public static void main(String[] args) {
        int pilihan;
        do {
            System.out.println(x:"\n===== Sistem Rental Mobil =====");
            System.out.println(x:"1. Tambah Mobil Mewah");
            System.out.println(x:"2. Tambah Mobil Ekonomi");
            System.out.println(x:"3. Lihat Daftar Mobil");
            System.out.println(x:"4. Edit Mobil");
            System.out.println(x:"5. Hapus Mobil");
            System.out.println(x:"6. Keluar");
            System.out.print(s:"Pilih menu: ");
            pilihan = scanner.nextInt();
            scanner.nextLine();

            switch (pilihan) {
                case 1:
                    tambahMobilMewah();
                    break;
                case 2:
                    tambahMobilEkonomi();
                    break;
                case 3:
                    lihatMobil();
                    break;
                case 4:
                    editMobil();
                    break;
                case 5:
                    hapusMobil();
                    break;
                case 6:
                    System.out.println(x:"Terima kasih telah menggunakan sistem.");
                    break;
                default:
                    System.out.println(x:"Pilihan tidak valid.");
            }
        } while (pilihan != 6);
    }
```



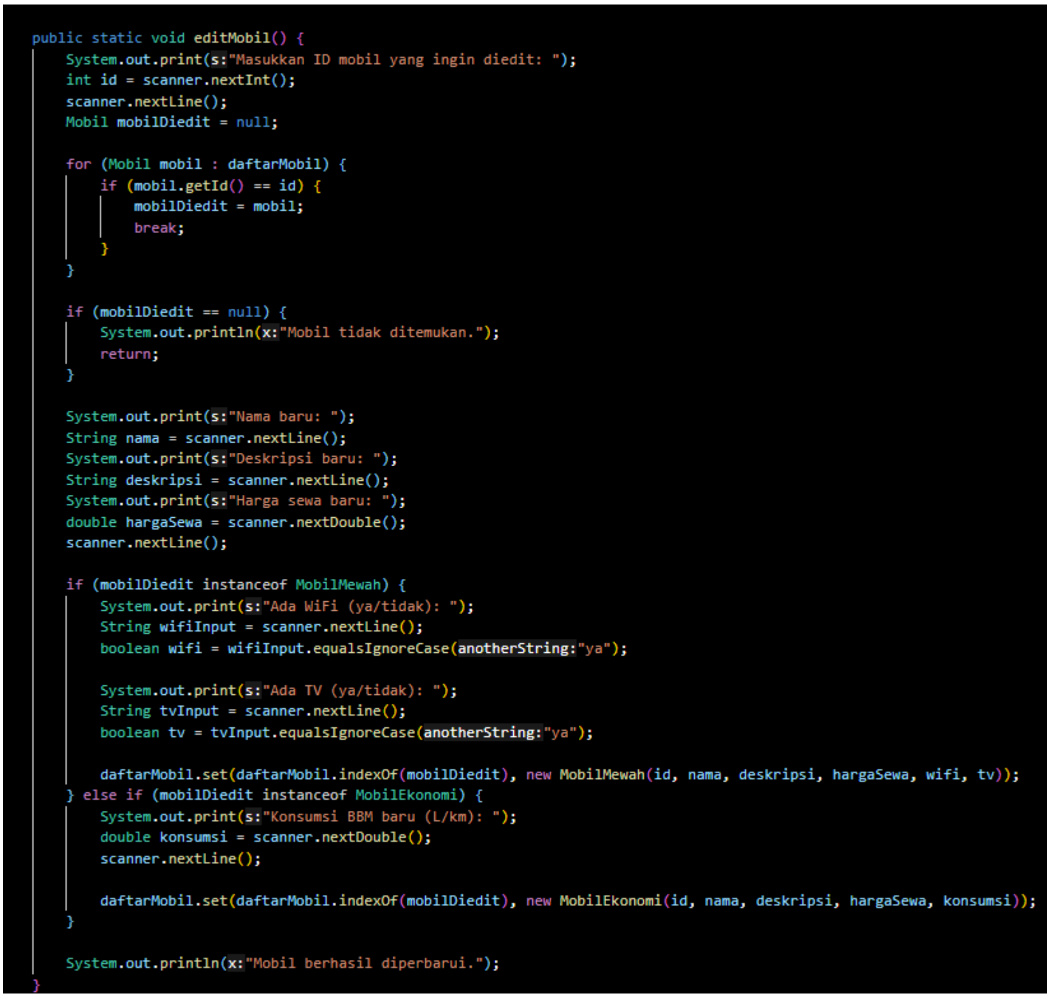
```java
public static void tambahMobilMewah() {
    System.out.print(s:"Nama mobil: ");
    String nama = scanner.nextLine();
    System.out.print(s:"Deskripsi: ");
    String deskripsi = scanner.nextLine();
    System.out.print(s:"Harga sewa: ");
    double hargaSewa = scanner.nextDouble();
    scanner.nextLine();

    System.out.print(s:"Ada WiFi (ya/tidak): ");
    String wifiInput = scanner.nextLine();
    boolean adaWiFi = wifiInput.equalsIgnoreCase(anotherString:"ya");

    System.out.print(s:"Ada TV (ya/tidak): ");
    String tvInput = scanner.nextLine();
    boolean adaTV = tvInput.equalsIgnoreCase(anotherString:"ya");

    daftarMobil.add(new MobilMewah(nextId++, nama, deskripsi, hargaSewa, adaWiFi, adaTV));
    System.out.println(x:"Mobil mewah berhasil ditambahkan.");
}
```



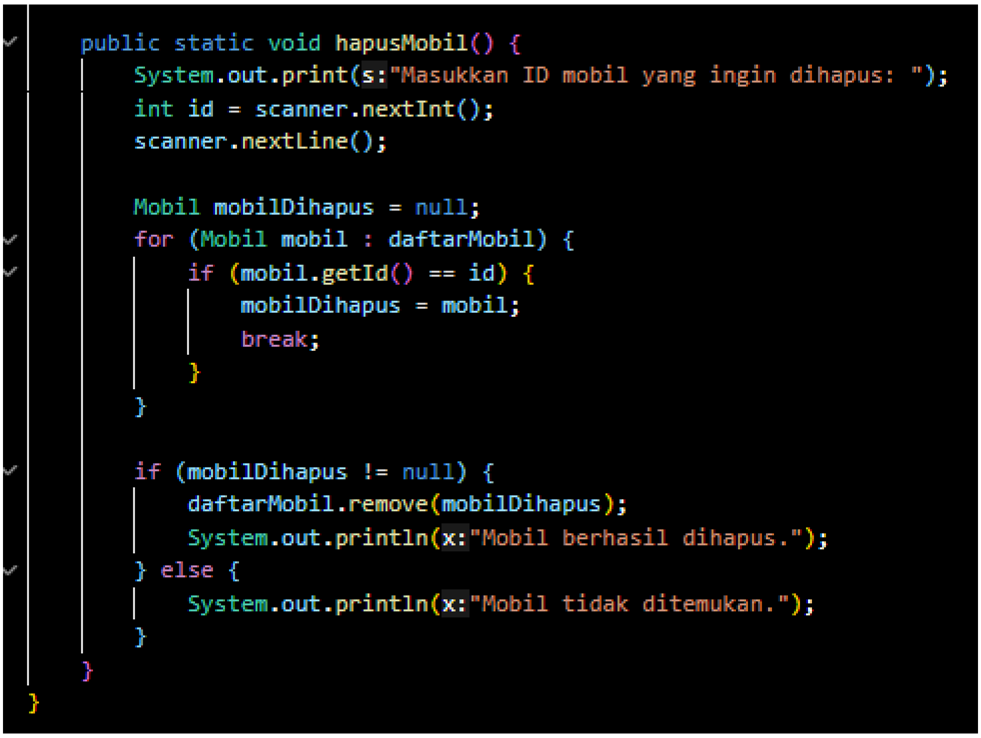
```java
public static void tambahMobilEkonomi() {
    System.out.print(s:"Nama mobil: ");
    String nama = scanner.nextLine();
    System.out.print(s:"Deskripsi: ");
    String deskripsi = scanner.nextLine();
    System.out.print(s:"Harga sewa: ");
    double hargaSewa = scanner.nextDouble();
    System.out.print(s:"Konsumsi BBM (L/km): ");
    double konsumsiBBM = scanner.nextDouble();
    scanner.nextLine();

    daftarMobil.add(new MobilEkonomi(nextId++, nama, deskripsi, hargaSewa, konsumsiBBM));
    System.out.println(x:"Mobil ekonomi berhasil ditambahkan.");
}
```



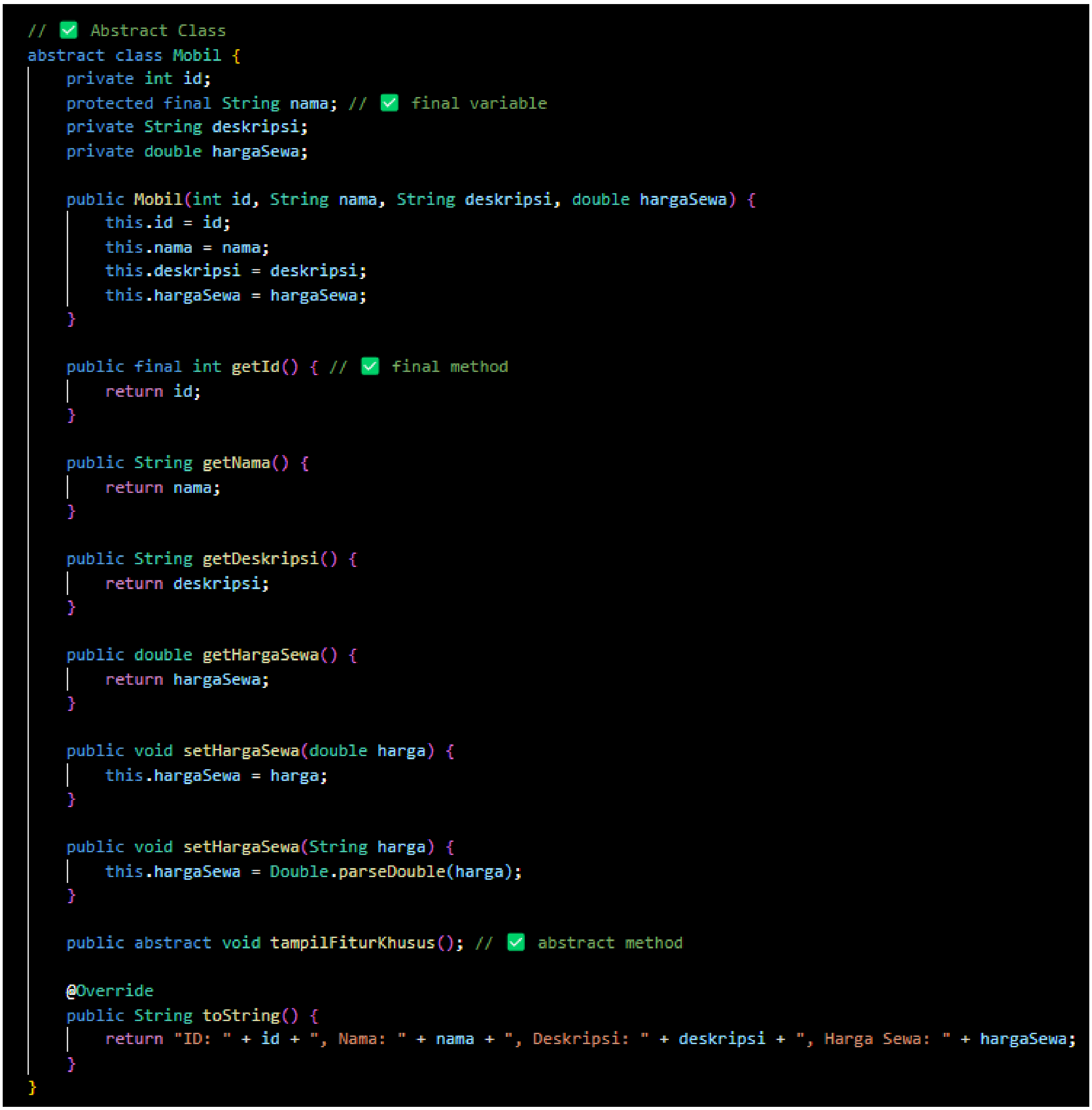
```java
public static void lihatMobil() {
    if (daftarMobil.isEmpty()) {
        System.out.println(x:"Tidak ada mobil tersedia.");
    } else {
        for (Mobil mobil : daftarMobil) {
            System.out.println(mobil);
            mobil.tampilFiturKhusus();
        }
    }
}
```



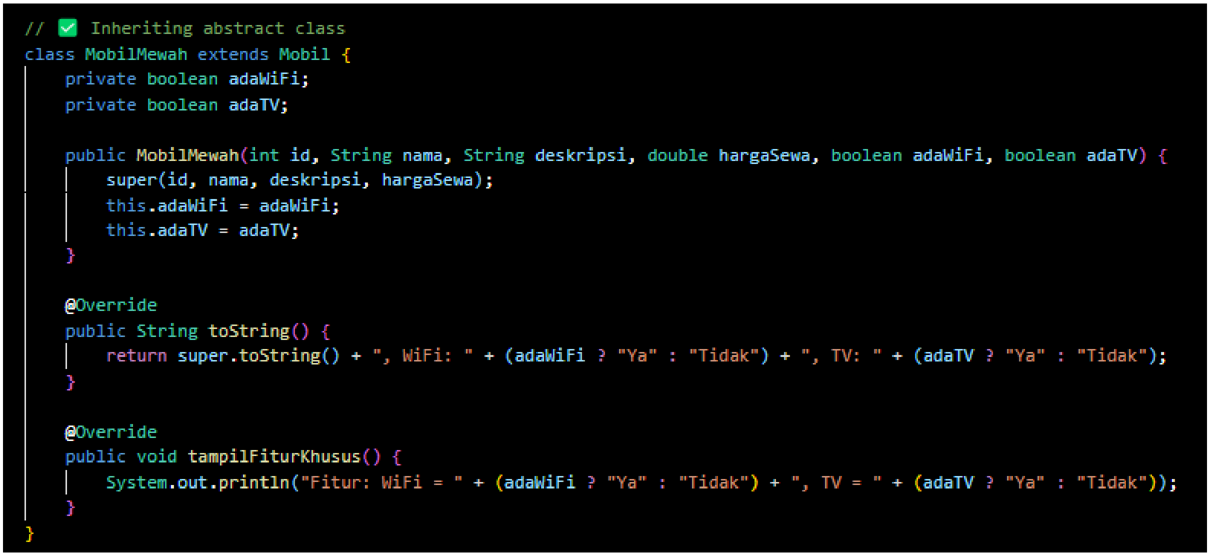
```java
public static void editMobil() {
    System.out.print(s:"Masukkan ID mobil yang ingin diedit: ");
    int id = scanner.nextInt();
    scanner.nextLine();
    Mobil mobilDiedit = null;

    for (Mobil mobil : daftarMobil) {
        if (mobil.getId() == id) {
            mobilDiedit = mobil;
            break;
        }
    }

    if (mobilDiedit == null) {
        System.out.println(x:"Mobil tidak ditemukan.");
        return;
    }

    System.out.print(s:"Nama baru: ");
    String nama = scanner.nextLine();
    System.out.print(s:"Deskripsi baru: ");
    String deskripsi = scanner.nextLine();
    System.out.print(s:"Harga sewa baru: ");
    double hargaSewa = scanner.nextDouble();
    scanner.nextLine();

    if (mobilDiedit instanceof MobilMewah) {
        System.out.print(s:"Ada WiFi (ya/tidak): ");
        String wifiInput = scanner.nextLine();
        boolean wifi = wifiInput.equalsIgnoreCase(anotherString:"ya");

        System.out.print(s:"Ada TV (ya/tidak): ");
        String tvInput = scanner.nextLine();
        boolean tv = tvInput.equalsIgnoreCase(anotherString:"ya");

        daftarMobil.set(daftarMobil.indexOf(mobilDiedit), new MobilMewah(id, nama, deskripsi, hargaSewa, wifi, tv));
    } else if (mobilDiedit instanceof MobilEkonomi) {
        System.out.print(s:"Konsumsi BBM baru (L/km): ");
        double konsumsi = scanner.nextDouble();
        scanner.nextLine();

        daftarMobil.set(daftarMobil.indexOf(mobilDiedit), new MobilEkonomi(id, nama, deskripsi, hargaSewa, konsumsi));
    }

    System.out.println(x:"Mobil berhasil diperbarui.");
}
```



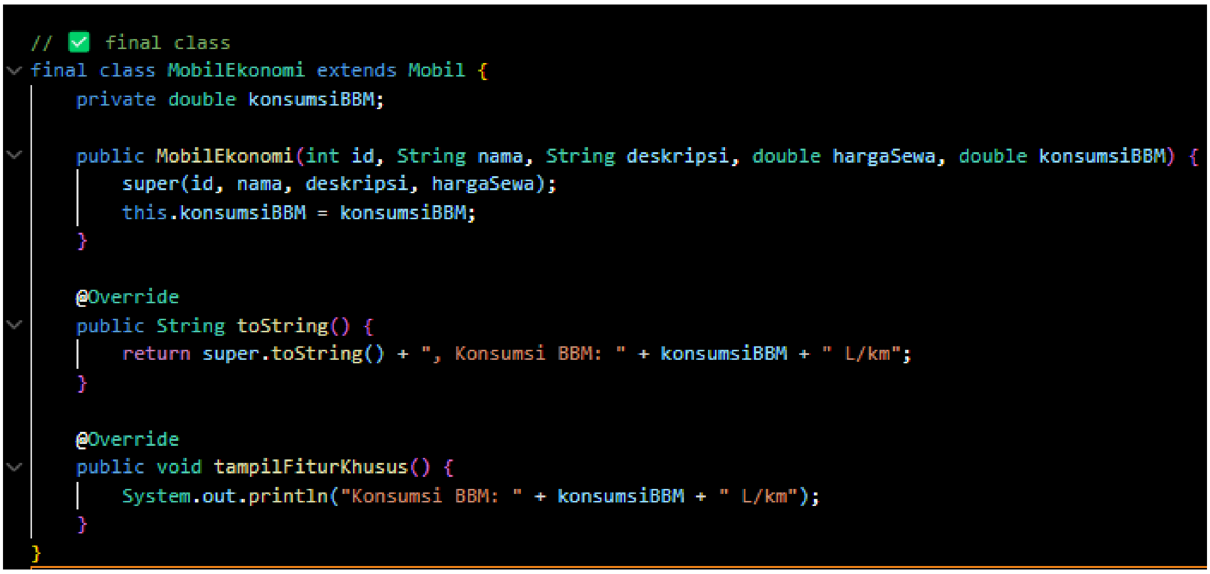
```
    public static void hapusMobil() {
        System.out.print(s:"Masukkan ID mobil yang ingin dihapus: ");
        int id = scanner.nextInt();
        scanner.nextLine();

        Mobil mobilDihapus = null;
        for (Mobil mobil : daftarMobil) {
            if (mobil.getId() == id) {
                mobilDihapus = mobil;
                break;
            }
        }

        if (mobilDihapus != null) {
            daftarMobil.remove(mobilDihapus);
            System.out.println(x:"Mobil berhasil dihapus.");
        } else {
            System.out.println(x:"Mobil tidak ditemukan.");
        }
    }
}
```



```java
// ✅ Abstract Class
abstract class Mobil {
    private int id;
    protected final String nama; // ✅ final variable
    private String deskripsi;
    private double hargaSewa;

    public Mobil(int id, String nama, String deskripsi, double hargaSewa) {
        this.id = id;
        this.nama = nama;
        this.deskripsi = deskripsi;
        this.hargaSewa = hargaSewa;
    }

    public final int getId() { // ✅ final method
        return id;
    }

    public String getNama() {
        return nama;
    }

    public String getDeskripsi() {
        return deskripsi;
    }

    public double getHargaSewa() {
        return hargaSewa;
    }

    public void setHargaSewa(double harga) {
        this.hargaSewa = harga;
    }

    public void setHargaSewa(String harga) {
        this.hargaSewa = Double.parseDouble(harga);
    }

    public abstract void tampilFiturKhusus(); // ✅ abstract method

    @Override
    public String toString() {
        return "ID: " + id + ", Nama: " + nama + ", Deskripsi: " + deskripsi + ", Harga Sewa: " + hargaSewa;
    }
}
```



```java
// ✅ Inheriting abstract class
class MobilMewah extends Mobil {
    private boolean adaWiFi;
    private boolean adaTV;

    public MobilMewah(int id, String nama, String deskripsi, double hargaSewa, boolean adaWiFi, boolean adaTV) {
        super(id, nama, deskripsi, hargaSewa);
        this.adaWiFi = adaWiFi;
        this.adaTV = adaTV;
    }

    @Override
    public String toString() {
        return super.toString() + ", WiFi: " + (adaWiFi ? "Ya" : "Tidak") + ", TV: " + (adaTV ? "Ya" : "Tidak");
    }

    @Override
    public void tampilFiturKhusus() {
        System.out.println("Fitur: WiFi = " + (adaWiFi ? "Ya" : "Tidak") + ", TV = " + (adaTV ? "Ya" : "Tidak"));
    }
}
```



```
// ✅ final class
final class MobilEkonomi extends Mobil {
    private double konsumsiBBM;

    public MobilEkonomi(int id, String nama, String deskripsi, double hargaSewa, double konsumsiBBM) {
        super(id, nama, deskripsi, hargaSewa);
        this.konsumsiBBM = konsumsiBBM;
    }

    @Override
    public String toString() {
        return super.toString() + ", Konsumsi BBM: " + konsumsiBBM + " L/km";
    }

    @Override
    public void tampilFiturKhusus() {
        System.out.println("Konsumsi BBM: " + konsumsiBBM + " L/km");
    }
}
```