

```

public abstract class Dokter { // Mengubah kelas menjadi abstract
    private final String noLisensi; // Menambahkan keyword final
    private String nama;
    private String noTelp;

    private static final List<Dokter> dokterList = new ArrayList<>(); // Menambahkan final pada atribut
    private static Scanner input = new Scanner(System.in);

```

```

    public Dokter(String noLisensi, String nama, String noTelp) {
        this.noLisensi = noLisensi;
        this.nama = nama;
        this.noTelp = noTelp;
    }

```

```

    public String getNoLisensi() {
        return noLisensi;
    }

```

```

    public String getNama() {
        return nama;
    }

```

```

    public void setNama(String nama) {
        this.nama = nama;
    }

```

```

    public String getNoTelp() {
        return noTelp;
    }

```

```

    public final String toString() { // Menambahkan keyword final pada metode
        return "\nNo Lisensi: " + noLisensi + "\nNama: " + nama + "\nNo Telp: " + noTelp;
    }

```

```

    // Menambahkan abstract method
    public abstract String getSpesialisasi();

```

```

    public static void kelolaDokter() {
        while (true) {
            System.out.println(x: "\n=== CRUD DATA DOKTER ===");
            System.out.println(x: "1. Tambah Dokter");
            System.out.println(x: "2. Lihat Daftar Dokter");
            System.out.println(x: "3. Update Data Dokter");
            System.out.println(x: "4. Hapus Data Dokter");
            System.out.println(x: "5. Kembali ke Menu Utama");
            System.out.print(s: "Pilih opsi: ");

            int pilihan = input.nextInt();
            input.nextLine(); // Clear buffer

            switch (pilihan) {
                case 1 -> tambahDokter();
                case 2 -> lihatDokter();
                case 3 -> updateDokter();
                case 4 -> hapusDokter();
                case 5 -> {
                    System.out.println(x: "Kembali ke menu utama...");
                    return;
                }
                default -> System.out.println(x: "Pilihan tidak valid. Silakan coba lagi.");
            }
        }
    }

```

```

package model;

public class DokterSpesialis extends Dokter {
    private final String bidangSpesialisasi; // Menambahkan final pada atribut

    public DokterSpesialis(String noLisensi, String nama, String noTelp, String bidangSpesialisasi) {
        super(noLisensi, nama, noTelp);
        this.bidangSpesialisasi = bidangSpesialisasi;
    }

    public String getBidangSpesialisasi() {
        return bidangSpesialisasi;
    }

    @Override
    public String getSpesialisasi() {
        return "Spesialis " + bidangSpesialisasi;
    }
}

```

```

package model;

public final class DokterUmum extends Dokter { // Menambahkan final pada kelas

    public DokterUmum(String noLisensi, String nama, String noTelp) {
        super(noLisensi, nama, noTelp);
    }

    @Override
    public String getSpesialisasi() {
        return "Umum";
    }
}

```

```

public class Pemilik {
    private final int id; // Menambahkan final pada atribut
    private String nama;
    private String alamat;
    private String noTelp;

    private static final List<Pemilik> pemilikList = new ArrayList<>(); // Menambahkan final pada atribut
    private static final Scanner input = new Scanner(System.in); // Menambahkan final pada Scanner

    public Pemilik(int id, String nama, String alamat, String noTelp) {
        this.id = id;
        this.nama = nama;
        this.alamat = alamat;
        this.noTelp = noTelp;
    }

    public int getId() {
        return id;
    }

    public String getNama() {
        return nama;
    }
}

```

```

public final String toString() { // Menambahkan final pada metode
    return "\nID: " + id + "\nNama: " + nama + "\nAlamat: " + alamat + "\nNo Telp: " + noTelp;
}

public static void kelolaPemilik() {
    while (true) {
        System.out.println(x: "\n=== CRUD DATA PEMILIK ===");
        System.out.println(x: "1. Tambah Pemilik");
        System.out.println(x: "2. Lihat Daftar Pemilik");
        System.out.println(x: "3. Update Data Pemilik");
        System.out.println(x: "4. Hapus Data Pemilik");
        System.out.println(x: "5. Kembali ke Menu Utama");
        System.out.print(s: "Pilih opsi: ");

        int pilihan = input.nextInt();
        input.nextLine();

        switch (pilihan) {
            case 1 -> tambahPemilik();
            case 2 -> lihatPemilik();
            case 3 -> updatePemilik();
            case 4 -> hapusPemilik();
            case 5 -> {
                System.out.println(x: "Kembali ke menu utama...");
                return;
            }
            default -> System.out.println(x: "Pilihan tidak valid. Silakan coba lagi.");
        }
    }
}

public static void tambahPemilik() {

```

```

package model;

public final class PemilikVIP extends Pemilik { // Menambahkan final pada kelas
    private String membershipLevel;

    public PemilikVIP(int id, String nama, String alamat, String noTelp, String membershipLevel) {
        super(id, nama, alamat, noTelp);
        this.membershipLevel = membershipLevel;
    }

    public String getMembershipLevel() {
        return membershipLevel;
    }
}

```