

## 1. App.java

```
import model.Hewan;
import model.Pemilik;
import model.Dokter;
import model.Kunjungan;
import model.Resep;
import model.Jadwal;
import model.Pembayaran;
import model.Obat;
import model.Stok;
import java.util.ArrayList;
import java.util.Scanner;
public class App {
    static ArrayList<Hewan> dataHewan = new ArrayList<Hewan>();
    static Scanner input = new Scanner(System.in);
    public static void main(String[] args) throws Exception {
        while (true) {
            System.out.println("\n==== MENU CRUD KLINIK HEWAN ====");
            System.out.println("1. Kelola Data Hewan");
            System.out.println("2. Kelola Data Pemilik");
            System.out.println("3. Kelola Data Dokter");
            System.out.println("4. Kelola Data Kunjungan");
            System.out.println("5. Kelola Data Resep Obat");
            System.out.println("6. Kelola Data Jadwal Konsultasi");
            System.out.println("7. Kelola Data Pembayaran");
            System.out.println("8. Kelola Data Obat");
            System.out.println("9. Kelola Data Stok Obat");
            System.out.println("10. Keluar");
            System.out.print("Pilih entitas yang ingin dikelola: ");

            int pilihan = input.nextInt();
            switch (pilihan) {
                case 1 -> Hewan.kelolaHewan();
                case 2 -> Pemilik.kelolaPemilik();
                case 3 -> Dokter.kelolaDokter();
                case 4 -> Kunjungan.kelolaKunjungan(dataHewan, new ArrayList<Dokter>());
                case 5 -> Resep.kelolaResepObat();
                case 6 -> Jadwal.kelolaJadwalKonsultasi();
                case 7 -> Pembayaran.kelolaPembayaran();
                case 8 -> Obat.kelolaObat();
                case 9 -> Stok.kelolaStokObat();
                case 10 -> {
                    System.out.println("Program Dikeluarkan");
                    input.close();
                    return;
                }
                default -> System.out.println("Pilihan tidak valid. Coba lagi!");
            }
        }
    }
}
```

## 2. Dokter.java

```
package model;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Dokter {
    private String noLisensi;
    private String nama;
    private String noTelp;

    private static List<Dokter> dokterList = new ArrayList<>();
    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 void setNoLisensi(String noLisensi) {
        this.noLisensi = noLisensi;
    }

    public String getNama() {
        return nama;
    }

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

    public String getNoTelp() {
        return noTelp;
    }

    public void setNoTelp(String noTelp) {
        this.noTelp = noTelp;
    }

    @Override
    public String toString() {
        return "\nNo Licensi: " + noLisensi + "\nNama: " + nama + "\nNo Telp: " + noTelp;
    }

    public static void kelolaDokter() {
        while (true) {
            System.out.println("\n==== CRUD DATA DOKTER ====");
            System.out.println("1. Tambah Dokter");
            System.out.println("2. Lihat Daftar Dokter");
            System.out.println("3. Update Data Dokter");
            System.out.println("4. Hapus Data Dokter");
            System.out.println("5. Kembali ke Menu Utama");
            System.out.print("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("Kembali ke menu utama...");
                    return;
                }
                default -> System.out.println("Pilihan tidak valid. Silakan coba lagi.");
            }
        }
    }
}
```

```
public static void tambahDokter() {
    System.out.println("\n> Tambah Data Dokter <<");
    System.out.print("No Lisensi: ");
    String noLisensi = input.nextLine();
    input.nextLine();
    System.out.print("Nama: ");
    String nama = input.nextLine();
    System.out.print("No Telp: ");
    String noTelp = input.nextLine();

    dokterList.add(new Dokter(noLisensi, nama, noTelp));
    System.out.println("Data dokter berhasil ditambahkan!");
}

public static void lihatDokter() {
    System.out.println("\n> Daftar Dokter <<");
    if (dokterList.isEmpty()) {
        System.out.println("Belum ada data dokter.");
        return;
    }
    for (Dokter dokter : dokterList) {
        System.out.println(dokter);
    }
}

public static void updateDokter() {
    System.out.println("\n> Update Data Dokter <<");
    System.out.print("Masukkan No Lisensi Dokter yang ingin diupdate: ");
    String noLisensi = input.nextLine();
    input.nextLine();

    for (Dokter dokter : dokterList) {
        if (dokter.getNoLisensi().equals(noLisensi)) {
            System.out.print("Nama Baru: ");
            dokter.nama = input.nextLine();
            System.out.print("No Telp Baru: ");
            dokter.noTelp = input.nextLine();

            System.out.println("Data dokter berhasil diperbarui!");
            return;
        }
    }
    System.out.println("Dokter dengan No Lisensi tersebut tidak ditemukan.");
}

public static void hapusDokter() {
    System.out.println("\n> Hapus Data Dokter <<");
    System.out.print("Masukkan No Lisensi Dokter yang ingin dihapus: ");
    String noLisensi = input.nextLine();
    input.nextLine();

    boolean removed = dokterList.removeIf(dokter ->
dokter.getNoLisensi().equals(noLisensi));
    System.out.println("Data dokter berhasil dihapus!");
} else {
    System.out.println("Dokter dengan No Lisensi tersebut tidak ditemukan.");
}
}
```

### 3. DokterSpesialis

```
package model;

public class DokterSpesialis extends Dokter {
    private String spesialisasi;

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

    public String getSpesialisasi() {
        return spesialisasi;
    }

    public void setSpesialisasi(String spesialisasi) {
        this.spesialisasi = spesialisasi;
    }

    @Override
    public String toString() {
        return "[Dokter Spesialis]\n" + super.toString() + "\nSpesialisasi: " + spesialisasi;
    }
}
```

### 4. DokterUmum

```
package model;

public class DokterUmum extends Dokter {

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

    @Override
    public String toString() {
        return "[Dokter Umum]\n" + super.toString();
    }
}
```

## 5. Hewan

```
package model;

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

public class Hewan {
    private int id;
    private String nama;
    private String jenis;
    private String ras;
    private String jenisKelamin;
    private int usia;
    private boolean statusVaksinasi;

    public Hewan(int id, String nama, String jenis, String ras, String jenisKelamin, int usia, boolean statusVaksinasi) {
        this.id = id;
        this.nama = nama;
        this.jenis = jenis;
        this.ras = ras;
        this.jenisKelamin = jenisKelamin;
        this.usia = usia;
        this.statusVaksinasi = statusVaksinasi;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getNama() {
        return nama;
    }

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

    public String getJenis() {
        return jenis;
    }

    public void setJenis(String jenis) {
        this.jenis = jenis;
    }

    public String getRas() {
        return ras;
    }

    public void setRas(String ras) {
        this.ras = ras;
    }

    public String getJenisKelamin() {
        return jenisKelamin;
    }

    public void setJenisKelamin(String jenisKelamin) {
        this.jenisKelamin = jenisKelamin;
    }

    public int getUsia() {
        return usia;
    }

    public void setUsia(int usia) {
        this.usia = usia;
    }

    public boolean isStatusVaksinasi() {
        return statusVaksinasi;
    }

    public void setStatusVaksinasi(boolean statusVaksinasi) {
        this.statusVaksinasi = statusVaksinasi;
    }

    @Override
    public String toString() {
        return "\nID: " + id + "\nNama: " + nama + "\nJenis: " + jenis + "\nRas: " + ras +
            "\nJenis Kelamin: " + jenisKelamin + "\nUsia: " + usia +
            "\nStatus Vaksinasi: " + (statusVaksinasi ? "Sudah" : "Belum");
    }
}

static ArrayList<Hewan> daftarHewan = new ArrayList<>();
static Scanner input = new Scanner(System.in);

public static void kelolaHewan() {
    while (true) {
        System.out.println("\n==== CRUD DATA HEWAN ====");
        System.out.println("1. Tambah Hewan");
        System.out.println("2. Lihat Daftar Hewan");
        System.out.println("3. Update Data Hewan");
        System.out.println("4. Hapus Data Hewan");
        System.out.println("5. Kembali ke Menu Utama");
        System.out.print("Pilih opsi: ");
    }
}
```

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

switch (pilihan) {
    case 1 -> tambahHewan();
    case 2 -> lihatHewan();
    case 3 -> updateHewan();
    case 4 -> hapusHewan();
    case 5 -> {
        System.out.println("Kembali ke menu utama...");
        return;
    }
    default -> System.out.println("Pilihan tidak valid. Silakan coba lagi.");
}

public static void tambahHewan() {
    System.out.println("\n>> Tambah Data Hewan <<");
    System.out.print("ID Hewan: ");
    int id = input.nextInt();
    input.nextLine();
    System.out.print("Nama Hewan: ");
    String nama = input.nextLine();
    System.out.print("Jenis: ");
    String jenis = input.nextLine();
    System.out.print("Ras: ");
    String ras = input.nextLine();
    System.out.print("Jenis Kelamin: ");
    String jenisKelamin = input.nextLine();
    System.out.print("Usia: ");
    int usia = input.nextInt();
    input.nextLine();
    System.out.print("Status Vaksinasi (true/false): ");
    boolean vaksinasi = input.nextBoolean();

    daftarHewan.add(new Hewan(id, nama, jenis, ras, jenisKelamin, usia, vaksinasi));
    System.out.println("Data hewan berhasil ditambahkan!");
}

public static void lihatHewan() {
    System.out.println("\n>> Daftar Hewan <<");
    if (daftarHewan.isEmpty()) {
        System.out.println("Belum ada data hewan.");
        return;
    }
    for (Hewan hewan : daftarHewan) {
        System.out.println(hewan);
    }
}

public static void updateHewan() {
    System.out.println("\n>> Update Data Hewan <<");
    System.out.print("Masukkan ID Hewan yang ingin diupdate: ");
    int id = input.nextInt();
    input.nextLine();

    for (Hewan hewan : daftarHewan) {
        if (hewan.getId() == id) {
            System.out.print("Nama Baru: ");
            hewan.setNama(input.nextLine());
            System.out.print("Jenis Baru: ");
            hewan.setJenis(input.nextLine());
            System.out.print("Ras Baru: ");
            hewan.setRas(input.nextLine());
            System.out.print("Jenis Kelamin Baru: ");
            hewan.setJenisKelamin(input.nextLine());
            System.out.print("Usia Baru: ");
            hewan.setUsia(input.nextInt());
            input.nextLine();
            System.out.print("Status Vaksinasi Baru (true/false): ");
            hewan.setStatusVaksinasi(input.nextBoolean());

            System.out.println("Data hewan berhasil diperbarui!");
            return;
        }
    }
    System.out.println("Hewan dengan ID tersebut tidak ditemukan.");
}

public static void hapusHewan() {
    System.out.println("\n>> Hapus Data Hewan <<");
    System.out.print("Masukkan ID Hewan yang ingin dihapus: ");
    int id = input.nextInt();
    input.nextLine();

    boolean removed = daftarHewan.removeIf(hewan -> hewan.getId() == id);
    if (removed) {
        System.out.println("Data hewan berhasil dihapus!");
    } else {
        System.out.println("Hewan dengan ID tersebut tidak ditemukan.");
    }
}
```

## 6. Jadwal



```
package model;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Jadwal {
    private int id_jadwal;
    private int id_hewan;
    private String no_lisensi;
    private String tanggal_waktu;

    private static List<Jadwal> jadwalList = new ArrayList<>();
    private static Scanner input = new Scanner(System.in);

    public Jadwal(int id_jadwal, int id_hewan, String no_lisensi, String tanggal_waktu)
    {
        this.id_jadwal = id_jadwal;
        this.id_hewan = id_hewan;
        this.no_lisensi = no_lisensi;
        this.tanggal_waktu = tanggal_waktu;
    }

    public int getIdJadwal() {
        return id_jadwal;
    }

    public void setIdJadwal(int id_jadwal) {
        this.id_jadwal = id_jadwal;
    }

    public int getIdHewan() {
        return id_hewan;
    }

    public void setIdHewan(int id_hewan) {
        this.id_hewan = id_hewan;
    }

    public String getNoLisensi() {
        return no_lisensi;
    }

    public void setNoLisensi(String no_lisensi) {
        this.no_lisensi = no_lisensi;
    }

    public String getTanggalWaktu() {
        return tanggal_waktu;
    }

    public void setTanggalWaktu(String tanggal_waktu) {
        this.tanggal_waktu = tanggal_waktu;
    }

    @Override
    public String toString() {
        return "\nJadwal{" +
            "id_jadwal=" + id_jadwal +
            ", id_hewan=" + id_hewan +
            ", no_lisensi='" + no_lisensi + '\'' +
            ", tanggal_waktu='" + tanggal_waktu + '\'' +
            '}';
    }
}

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

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

        switch (pilihan) {
            case 1 -> tambahJadwal();
            case 2 -> lihatJadwal();
            case 3 -> updateJadwal();
            case 4 -> hapusJadwal();
            case 5 -> {
                System.out.println("Kembali ke menu utama...");
                return;
            }
            default -> System.out.println("Pilihan tidak valid. Silakan coba lagi.");
        }
    }
}
```

```
public static void tambahJadwal() {
    System.out.println("\n>> Tambah Data Jadwal <<");
    System.out.print("ID Jadwal: ");
    int id_jadwal = input.nextInt();
    input.nextLine();
    System.out.print("ID Hewan: ");
    int id_hewan = input.nextInt();
    input.nextLine();
    System.out.print("No Lisensi: ");
    String no_lisensi = input.nextLine();
    System.out.print("Tanggal dan Waktu: ");
    String tanggal_waktu = input.nextLine();

    jadwallist.add(new Jadwal(id_jadwal, id_hewan, no_lisensi, tanggal_waktu));
    System.out.println("Data jadwal berhasil ditambahkan!");
}

public static void lihatJadwal() {
    System.out.println("\n>> Daftar Jadwal <<");
    if (jadwallist.isEmpty()) {
        System.out.println("Belum ada data jadwal.");
        return;
    }
    for (Jadwal jadwal : jadwallist) {
        System.out.println(jadwal);
    }
}

public static void updateJadwal() {
    System.out.println("\n>> Update Data Jadwal <<");
    System.out.print("Masukkan ID Jadwal yang ingin diupdate: ");
    int id_jadwal = input.nextInt();
    input.nextLine();

    for (Jadwal jadwal : jadwallist) {
        if (jadwal.getIdJadwal() == id_jadwal) {
            System.out.print("ID Hewan Baru: ");
            jadwal.setIdHewan(input.nextInt());
            input.nextLine();
            System.out.print("No Lisensi Baru: ");
            jadwal.setNoLisensi(input.nextLine());
            System.out.print("Tanggal dan Waktu Baru: ");
            jadwal.setTanggalWaktu(input.nextLine());

            System.out.println("Data jadwal berhasil diperbarui!");
            return;
        }
    }
    System.out.println("Jadwal dengan ID tersebut tidak ditemukan.");
}

public static void hapusJadwal() {
    System.out.println("\n>> Hapus Data Jadwal <<");
    System.out.print("Masukkan ID Jadwal yang ingin dihapus: ");
    int id_jadwal = input.nextInt();
    input.nextLine();

    boolean removed = jadwallist.removeIf(jadwal -> jadwal.getIdJadwal() == id_jadwal);
    if (removed) {
        System.out.println("Data jadwal berhasil dihapus!");
    } else {
        System.out.println("Jadwal dengan ID tersebut tidak ditemukan.");
    }
}
```

## 7. Kunjungan

```
package model;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Kunjungan {
    private int id_kunjungan;
    private int id_hewan;
    private String no_lisensi;
    private String tanggal_kunjungan;
    private String keluhan;
    private String diagnosa;
    private String tindakan;

    private static List<Kunjungan> kunjunganList = new ArrayList<>();
    private static Scanner input = new Scanner(System.in);

    public Kunjungan(int id_kunjungan, int id_hewan, String no_lisensi, String tanggal_kunjungan,
String keluhan, String diagnosa, String tindakan) {
        this.id_kunjungan = id_kunjungan;
        this.id_hewan = id_hewan;
        this.no_lisensi = no_lisensi;
        this.tanggal_kunjungan = tanggal_kunjungan;
        this.keluhan = keluhan;
        this.diagnosa = diagnosa;
        this.tindakan = tindakan;
    }

    public int getIdKunjungan() {
        return id_kunjungan;
    }

    public void setIdKunjungan(int id_kunjungan) {
        this.id_kunjungan = id_kunjungan;
    }

    public int getIdHewan() {
        return id_hewan;
    }

    public void setIdHewan(int id_hewan) {
        this.id_hewan = id_hewan;
    }

    public String getNoLisensi() {
        return no_lisensi;
    }

    public void setNoLisensi(String no_lisensi) {
        this.no_lisensi = no_lisensi;
    }

    public String getTanggalKunjungan() {
        return tanggal_kunjungan;
    }

    public void setTanggalKunjungan(String tanggal_kunjungan) {
        this.tanggal_kunjungan = tanggal_kunjungan;
    }

    public String getKeluhan() {
        return keluhan;
    }

    public void setKeluhan(String keluhan) {
        this.keluhan = keluhan;
    }

    public String getDiagnosa() {
        return diagnosa;
    }

    public void setDiagnosa(String diagnosa) {
        this.diagnosa = diagnosa;
    }

    public String getTindakan() {
        return tindakan;
    }

    public void setTindakan(String tindakan) {
        this.tindakan = tindakan;
    }

    @Override
    public String toString() {
        return "\nKunjungan" +
            "id_kunjungan=" + id_kunjungan +
            ", id_hewan=" + id_hewan +
            ", no_lisensi=" + no_lisensi + '\'' +
            ", tanggal_kunjungan=" + tanggal_kunjungan + '\'' +
            ", keluhan=" + keluhan + '\'' +
            ", diagnosa=" + diagnosa + '\'' +
            ", tindakan=" + tindakan + '\'' +
            '}';
    }
}
```

```
  public static void kelolaKunjungan(List<Hewan> hewanList, List<Dokter> dokterList) {
    while (true) {
        System.out.println("\n==== CRUD DATA KUNJUNGAN ===");
        System.out.println("1. Tambah Kunjungan");
        System.out.println("2. Lihat Daftar Kunjungan");
        System.out.println("3. Update Data Kunjungan");
        System.out.println("4. Hapus Data Kunjungan");
        System.out.println("5. Kembali ke Menu Utama");
        System.out.print("Pilih opsi: ");

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

        switch (pilihan) {
            case 1 -> tambahKunjungan(hewanList, dokterList, kunjunganList, input);
            case 2 -> lihatKunjungan();
            case 3 -> updateKunjungan();
            case 4 -> hapusKunjungan();
            case 5 -> {
                System.out.println("Kembali ke menu utama...");
                return;
            }
            default -> System.out.println("Pilihan tidak valid. Silakan coba lagi.");
        }
    }
}

public static void tambahKunjungan(List<Hewan> hewanList, List<Dokter> dokterList, List<Kunjungan> kunjunganList, Scanner input) {
    System.out.println("\n>>> Tambah Data Kunjungan <<<");

    System.out.print("ID Kunjungan: ");
    int id_kunjungan = input.nextInt();
    input.nextLine();
    System.out.print("ID Hewan: ");
    int id_hewan = input.nextInt();
    input.nextLine();
    System.out.print("No Licensi: ");
    String no_lisensi = input.nextLine();
    System.out.print("Tanggal Kunjungan: ");
    String tanggal_kunjungan = input.nextLine();
    System.out.print("Keluhan: ");
    String keluhan = input.nextLine();
    System.out.print("Diagnosa: ");
    String diagnosa = input.nextLine();
    System.out.print("Tindakan: ");
    String tindakan = input.nextLine();

    kunjunganList.add(new Kunjungan(id_kunjungan, id_hewan, no_lisensi, tanggal_kunjungan, keluhan, diagnosa, tindakan));
    System.out.println("✓ Data kunjungan berhasil ditambahkan!");
}

public static void lihatKunjungan() {
    System.out.println("\n>>> Daftar Kunjungan <<<");
    if (kunjunganList.isEmpty()) {
        System.out.println("Belum ada data kunjungan.");
        return;
    }
    for (Kunjungan kunjungan : kunjunganList) {
        System.out.println(kunjungan);
    }
}

public static void updateKunjungan() {
    System.out.println("\n>>> Update Data Kunjungan <<<");
    System.out.print("Masukkan ID Kunjungan yang ingin diupdate: ");
    int id_kunjungan = input.nextInt();
    input.nextLine();

    for (Kunjungan kunjungan : kunjunganList) {
        if (kunjungan.getIdKunjungan() == id_kunjungan) {
            System.out.print("ID Hewan Baru: ");
            kunjungan.setIdHewan(input.nextInt());
            input.nextLine();
            System.out.print("No Licensi Baru: ");
            kunjungan.setNoLicensi(input.nextLine());
            System.out.print("Tanggal Kunjungan Baru: ");
            kunjungan.setTanggalKunjungan(input.nextLine());
            System.out.print("Keluhan Baru: ");
            kunjungan.setKeluhan(input.nextLine());
            System.out.print("Diagnosa Baru: ");
            kunjungan.setDiagnosa(input.nextLine());
            System.out.print("Tindakan Baru: ");
            kunjungan.setTindakan(input.nextLine());

            System.out.println("Data kunjungan berhasil diperbarui!");
            return;
        }
    }
    System.out.println("Kunjungan dengan ID tersebut tidak ditemukan.");
}

public static void hapusKunjungan() {
    System.out.println("\n>>> Hapus Data Kunjungan <<<");
    System.out.print("Masukkan ID Kunjungan yang ingin dihapus: ");
    int id_kunjungan = input.nextInt();
    input.nextLine();

    boolean removed = kunjunganList.removeIf(kunjungan -> kunjungan.getIdKunjungan() ==
id_kunjungan);
    if (removed) {
        System.out.println("Data kunjungan berhasil dihapus!");
    } else {
        System.out.println("Kunjungan dengan ID tersebut tidak ditemukan.");
    }
}
}
```

## 8. Obat



```
package model;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Obat {
    private int id_obat;
    private String nama_obat;
    private String kategori;
    private String komposisi;
    private String indikasi;
    private String dosis;
    private String efek_samping;

    private static List<Obat> obatList = new ArrayList<>();
    private static Scanner input = new Scanner(System.in);

    public Obat(int id_obat, String nama_obat, String kategori, String komposisi, String indikasi,
    String dosis, String efek_samping) {
        this.id_obat = id_obat;
        this.nama_obat = nama_obat;
        this.kategori = kategori;
        this.komposisi = komposisi;
        this.indikasi = indikasi;
        this.dosis = dosis;
        this.efek_samping = efek_samping;
    }

    public int getIdObat() {
        return id_obat;
    }

    public void setIdObat(int id_obat) {
        this.id_obat = id_obat;
    }

    public String getNamaObat() {
        return nama_obat;
    }

    public void setNamaObat(String nama_obat) {
        this.nama_obat = nama_obat;
    }

    public String getKategori() {
        return kategori;
    }

    public void setKategori(String kategori) {
        this.kategori = kategori;
    }

    public String getKomposisi() {
        return komposisi;
    }

    public void setKomposisi(String komposisi) {
        this.komposisi = komposisi;
    }

    public String getIndikasi() {
        return indikasi;
    }

    public void setIndikasi(String indikasi) {
        this.indikasi = indikasi;
    }

    public String getDosis() {
        return dosis;
    }

    public void setDosis(String dosis) {
        this.dosis = dosis;
    }

    public String getEfekSamping() {
        return efek_samping;
    }

    public void setEfekSamping(String efek_samping) {
        this.efek_samping = efek_samping;
    }

    @Override
    public String toString() {
        return "Id Obat: " + id_obat + "\nNama Obat: " + nama_obat + "\nKategori: " + kategori +
        "\nKomposisi: " + komposisi + "\nIndikasi: " + indikasi + "\nDosis: " + dosis +
        "\nEfek Samping: " + efek_samping;
    }
}

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

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

        switch (pilihan) {
            case 1 -> tambahObat();
            case 2 -> lihatObat();
            case 3 -> updateObat();
            case 4 -> hapusObat();
            case 5 -> {
                System.out.println("Kembali ke menu utama...");
                return;
            }
            default -> System.out.println("Pilihan tidak valid. Silakan coba lagi.");
        }
    }
}
```

```
public static void tambahObat() {
    System.out.println("\n> Tambah Data Obat <<");
    System.out.print("ID Obat: ");
    int id_obat = input.nextInt();
    input.nextLine();
    System.out.print("Nama Obat: ");
    String nama_obat = input.nextLine();
    System.out.print("Kategori: ");
    String kategori = input.nextLine();
    System.out.print("Komposisi: ");
    String komposisi = input.nextLine();
    System.out.print("Indikasi: ");
    String indikasi = input.nextLine();
    System.out.print("Dosis: ");
    String dosis = input.nextLine();
    System.out.print("Efek Samping: ");
    String efek_samping = input.nextLine();

    obatList.add(new Obat(id_obat, nama_obat, kategori, komposisi, indikasi, dosis, efek_samping));
    System.out.println("Data obat berhasil ditambahkan!");
}

public static void lihatObat() {
    System.out.println("\n> Daftar Obat <<");
    if (obatList.isEmpty()) {
        System.out.println("Belum ada data obat.");
        return;
    }
    for (Obat obat : obatList) {
        System.out.println(obat);
    }
}

public static void updateObat() {
    System.out.println("\n> Update Data Obat <<");
    System.out.print("Masukkan ID Obat yang ingin diupdate: ");
    int id_obat = input.nextInt();
    input.nextLine();

    for (Obat obat : obatList) {
        if (obat.getIdObat() == id_obat) {
            System.out.print("Nama Obat Baru: ");
            obat.setNamaObat(input.nextLine());
            System.out.print("Kategori Baru: ");
            obat.setKategori(input.nextLine());
            System.out.print("Komposisi Baru: ");
            obat.setKomposisi(input.nextLine());
            System.out.print("Indikasi Baru: ");
            obat.setIndikasi(input.nextLine());
            System.out.print("Dosis Baru: ");
            obat.setDosis(input.nextLine());
            System.out.print("Efek Samping Baru: ");
            obat.setEfekSamping(input.nextLine());

            System.out.println("Data obat berhasil diperbarui!");
            return;
        }
    }
    System.out.println("Obat dengan ID tersebut tidak ditemukan.");
}

public static void hapusObat() {
    System.out.println("\n> Hapus Data Obat <<");
    System.out.print("Masukkan ID Obat yang ingin dihapus: ");
    int id_obat = input.nextInt();
    input.nextLine();

    boolean removed = obatList.removeIf(obat -> obat.getIdObat() == id_obat);
    if (removed) {
        System.out.println("Data obat berhasil dihapus!");
    } else {
        System.out.println("Obat dengan ID tersebut tidak ditemukan.");
    }
}
```

## 9. Pembayaran

```
package model;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Pembayaran {
    private int id_pembayaran;
    private int id_kunjungan;
    private double total_biaya;
    private String metode_pembayaran;

    private static List<Pembayaran> pembayaranList = new ArrayList<>();
    private static Scanner input = new Scanner(System.in);

    public Pembayaran(int id_pembayaran, int id_kunjungan, double total_biaya, String
metode_pembayaran) {
        this.id_pembayaran = id_pembayaran;
        this.id_kunjungan = id_kunjungan;
        this.total_biaya = total_biaya;
        this.metode_pembayaran = metode_pembayaran;
    }

    public int getIdPembayaran() {
        return id_pembayaran;
    }

    public void setIdPembayaran(int id_pembayaran) {
        this.id_pembayaran = id_pembayaran;
    }

    public int getIdKunjungan() {
        return id_kunjungan;
    }

    public void setIdKunjungan(int id_kunjungan) {
        this.id_kunjungan = id_kunjungan;
    }

    public double getTotalBiaya() {
        return total_biaya;
    }

    public void setTotalBiaya(double total_biaya) {
        this.total_biaya = total_biaya;
    }

    public String getMetodePembayaran() {
        return metode_pembayaran;
    }

    public void setMetodePembayaran(String metode_pembayaran) {
        this.metode_pembayaran = metode_pembayaran;
    }

    @Override
    public String toString() {
        return "\nPembayaran{" +
            "id_pembayaran=" + id_pembayaran +
            ", id_kunjungan=" + id_kunjungan +
            ", total_biaya=" + total_biaya +
            ", metode_pembayaran='" + metode_pembayaran + '\'' +
            '}';
    }
}

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

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

        switch (pilihan) {
            case 1 -> tambahPembayaran();
            case 2 -> lihatPembayaran();
            case 3 -> updatePembayaran();
            case 4 -> hapusPembayaran();
            case 5 -> {
                System.out.println("Kembali ke menu utama...");
                return;
            }
            default -> System.out.println("Pilihan tidak valid. Silakan coba lagi.");
        }
    }
}
```

```
● ● ●

public static void tambahPembayaran() {
    System.out.println("\n>> Tambah Data Pembayaran <<");
    System.out.print("ID Pembayaran: ");
    int id_pembayaran = input.nextInt();
    input.nextLine();
    System.out.print("ID Kunjungan: ");
    int id_kunjungan = input.nextInt();
    input.nextLine();
    System.out.print("Total Biaya: ");
    double total_biaya = input.nextDouble();
    input.nextLine();
    System.out.print("Metode Pembayaran: ");
    String metode_pembayaran = input.nextLine();

    pembayaranList.add(new Pembayaran(id_pembayaran, id_kunjungan, total_biaya,
    metode_pembayaran));
    System.out.println("Data pembayaran berhasil ditambahkan!");
}

public static void lihatPembayaran() {
    System.out.println("\n>> Daftar Pembayaran <<");
    if (pembayaranList.isEmpty()) {
        System.out.println("Belum ada data pembayaran.");
        return;
    }
    for (Pembayaran pembayaran : pembayaranList) {
        System.out.println(pembayaran);
    }
}

public static void updatePembayaran() {
    System.out.println("\n>> Update Data Pembayaran <<");
    System.out.print("Masukkan ID Pembayaran yang ingin diupdate: ");
    int id_pembayaran = input.nextInt();
    input.nextLine();

    for (Pembayaran pembayaran : pembayaranList) {
        if (pembayaran.getIdPembayaran() == id_pembayaran) {
            System.out.print("ID Kunjungan Baru: ");
            pembayaran.setIdKunjungan(input.nextInt());
            input.nextLine();
            System.out.print("Total Biaya Baru: ");
            pembayaran.setTotalBiaya(input.nextDouble());
            input.nextLine();
            System.out.print("Metode Pembayaran Baru: ");
            pembayaran.setMetodePembayaran(input.nextLine());

            System.out.println("Data pembayaran berhasil diperbarui!");
            return;
        }
    }
    System.out.println("Pembayaran dengan ID tersebut tidak ditemukan.");
}

public static void hapusPembayaran() {
    System.out.println("\n>> Hapus Data Pembayaran <<");
    System.out.print("Masukkan ID Pembayaran yang ingin dihapus: ");
    int id_pembayaran = input.nextInt();
    input.nextLine();

    boolean removed = pembayaranList.removeIf(pembayaran -> pembayaran.getIdPembayaran() ==
    id_pembayaran);
    if (removed) {
        System.out.println("Data pembayaran berhasil dihapus!");
    } else {
        System.out.println("Pembayaran dengan ID tersebut tidak ditemukan.");
    }
}
```

## 10. Pemilik



```
package model;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Pemilik {
    private int id;
    private String nama;
    private String alamat;
    private String noTelp;
    private String email;

    private static List<Pemilik> pemilikList = new ArrayList<>();
    private static Scanner input = new Scanner(System.in);

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

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getNama() {
        return nama;
    }

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

    public String getAlamat() {
        return alamat;
    }

    public void setAlamat(String alamat) {
        this.alamat = alamat;
    }

    public String getNoTelp() {
        return noTelp;
    }

    public void setNoTelp(String noTelp) {
        this.noTelp = noTelp;
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }

    @Override
    public String toString() {
        return "\nID: " + id + "\nNama: " + nama + "\nAlamat: " + alamat +
               "\nNo Telp: " + noTelp + "\nEmail: " + email;
    }
}

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

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

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

```
● ● ●

public static void tambahPemilik() {
    System.out.println("\n>> Tambah Data Pemilik <<");
    System.out.print("ID Pemilik: ");
    int id = input.nextInt();
    input.nextLine();
    System.out.print("Nama Pemilik: ");
    String nama = input.nextLine();
    System.out.print("Alamat: ");
    String alamat = input.nextLine();
    System.out.print("No Telp: ");
    String noTelp = input.nextLine();
    System.out.print("Email: ");
    String email = input.nextLine();

    if (email.trim().isEmpty()) {
        tambahPemilik(id, nama, alamat, noTelp);
    } else {
        tambahPemilik(id, nama, alamat, noTelp, email);
    }
}

public static void tambahPemilik(int id, String nama, String alamat, String noTelp) {
    pemilikList.add(new Pemilik(id, nama, alamat, noTelp, "N/A"));
    System.out.println("Data pemilik berhasil ditambahkan dengan email default!");
}

public static void tambahPemilik(int id, String nama, String alamat, String noTelp, String email)
{
    pemilikList.add(new Pemilik(id, nama, alamat, noTelp, email));
    System.out.println("Data pemilik berhasil ditambahkan!");
}

public static void lihatPemilik() {
    System.out.println("\n>> Daftar Pemilik <<");
    if (pemilikList.isEmpty()) {
        System.out.println("Belum ada data pemilik.");
        return;
    }
    for (Pemilik pemilik : pemilikList) {
        System.out.println(pemilik);
    }
}

public static void updatePemilik() {
    System.out.println("\n>> Update Data Pemilik <<");
    System.out.print("Masukkan ID Pemilik yang ingin diupdate: ");
    int id = input.nextInt();
    input.nextLine();

    for (Pemilik pemilik : pemilikList) {
        if (pemilik.getId() == id) {
            System.out.print("Nama Baru: ");
            pemilik.setNama(input.nextLine());
            System.out.print("Alamat Baru: ");
            pemilik.setAlamat(input.nextLine());
            System.out.print("No Telp Baru: ");
            pemilik.setNoTelp(input.nextLine());
            System.out.print("Email Baru: ");
            pemilik.setEmail(input.nextLine());

            System.out.println("Data pemilik berhasil diperbarui!");
            return;
        }
    }
    System.out.println("Pemilik dengan ID tersebut tidak ditemukan.");
}

public static void hapusPemilik() {
    System.out.println("\n>> Hapus Data Pemilik <<");
    System.out.print("Masukkan ID Pemilik yang ingin dihapus: ");
    int id = input.nextInt();
    input.nextLine();

    boolean removed = pemilikList.removeIf(pemilik -> pemilik.getId() == id);
    if (removed) {
        System.out.println("Data pemilik berhasil dihapus!");
    } else {
        System.out.println("Pemilik dengan ID tersebut tidak ditemukan.");
    }
}
```

## 11. PemilikVIP

```
package model;

public class PemilikVIP extends Pemilik {
    private String membershipLevel;

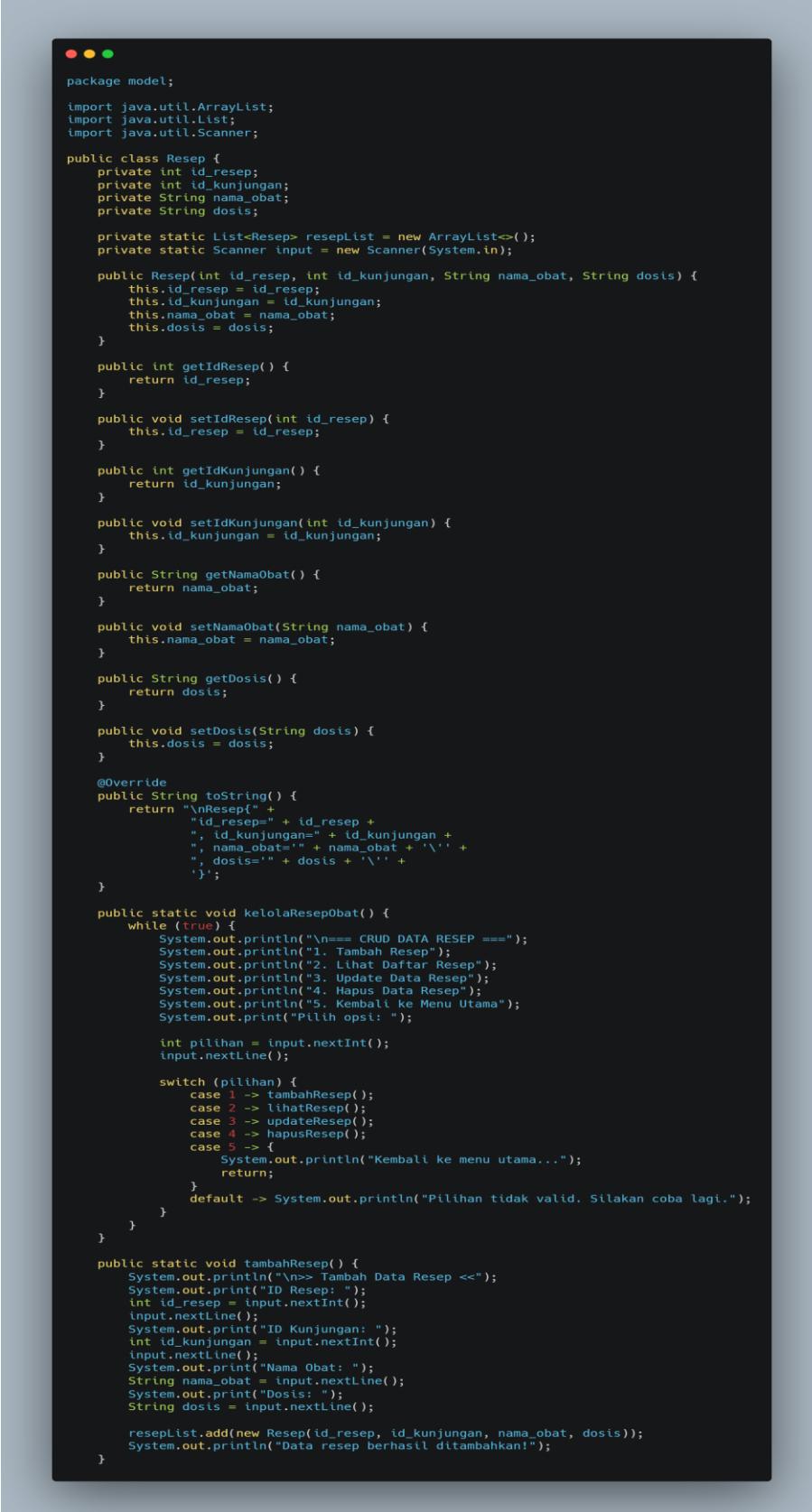
    public PemilikVIP(int id, String nama, String alamat, String noTelp, String email, String membershipLevel) {
        super(id, nama, alamat, noTelp, email);
        this.membershipLevel = membershipLevel;
    }

    public String getMembershipLevel() {
        return membershipLevel;
    }

    public void setMembershipLevel(String membershipLevel) {
        this.membershipLevel = membershipLevel;
    }

    @Override
    public String toString() {
        return super.toString() + "\nMembership Level: " + membershipLevel;
    }
}
```

## 12. Resep



```
package model;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Resep {
    private int id_resep;
    private int id_kunjungan;
    private String nama_obat;
    private String dosis;

    private static List<Resep> resepList = new ArrayList<>();
    private static Scanner input = new Scanner(System.in);

    public Resep(int id_resep, int id_kunjungan, String nama_obat, String dosis) {
        this.id_resep = id_resep;
        this.id_kunjungan = id_kunjungan;
        this.nama_obat = nama_obat;
        this.dosis = dosis;
    }

    public int getIdResep() {
        return id_resep;
    }

    public void setIdResep(int id_resep) {
        this.id_resep = id_resep;
    }

    public int getIdKunjungan() {
        return id_kunjungan;
    }

    public void setIdKunjungan(int id_kunjungan) {
        this.id_kunjungan = id_kunjungan;
    }

    public String getNamaObat() {
        return nama_obat;
    }

    public void setNamaObat(String nama_obat) {
        this.nama_obat = nama_obat;
    }

    public String getDosis() {
        return dosis;
    }

    public void setDosis(String dosis) {
        this.dosis = dosis;
    }

    @Override
    public String toString() {
        return "\nResep{" +
            "id_resep=" + id_resep +
            ", id_kunjungan=" + id_kunjungan +
            ", nama_obat='"+ nama_obat + '\'' +
            ", dosis='"+ dosis + '\'' +
            '}';
    }
}

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

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

        switch (pilihan) {
            case 1 -> tambahResep();
            case 2 -> lihatResep();
            case 3 -> updateResep();
            case 4 -> hapusResep();
            case 5 -> {
                System.out.println("Kembali ke menu utama...");
                return;
            }
            default -> System.out.println("Pilihan tidak valid. Silakan coba lagi.");
        }
    }
}

public static void tambahResep() {
    System.out.println("\n>> Tambah Data Resep <<");
    System.out.print("ID Resep: ");
    int id_resep = input.nextInt();
    input.nextLine();
    System.out.print("ID Kunjungan: ");
    int id_kunjungan = input.nextInt();
    input.nextLine();
    System.out.print("Nama Obat: ");
    String nama_obat = input.nextLine();
    System.out.print("Dosis: ");
    String dosis = input.nextLine();

    resepList.add(new Resep(id_resep, id_kunjungan, nama_obat, dosis));
    System.out.println("Data resep berhasil ditambahkan!");
}
```

```
public static void lihatResep() {
    System.out.println("\n>> Daftar Resep <<");
    if (resepList.isEmpty()) {
        System.out.println("Belum ada data resep.");
        return;
    }
    for (Resep resep : resepList) {
        System.out.println(resep);
    }
}

public static void updateResep() {
    System.out.println("\n>> Update Data Resep <<");
    System.out.print("Masukkan ID Resep yang ingin diupdate: ");
    int id_resep = input.nextInt();
    input.nextLine();

    for (Resep resep : resepList) {
        if (resep.getIdResep() == id_resep) {
            System.out.print("ID Kunjungan Baru: ");
            resep.setIdKunjungan(input.nextInt());
            input.nextLine();
            System.out.print("Nama Obat Baru: ");
            resep.setNamaObat(input.nextLine());
            System.out.print("Dosis Baru: ");
            resep.setDosis(input.nextLine());

            System.out.println("Data resep berhasil diperbarui!");
            return;
        }
    }
    System.out.println("Resep dengan ID tersebut tidak ditemukan.");
}

public static void hapusResep() {
    System.out.println("\n>> Hapus Data Resep <<");
    System.out.print("Masukkan ID Resep yang ingin dihapus: ");
    int id_resep = input.nextInt();
    input.nextLine();

    boolean removed = resepList.removeIf(resep -> resep.getIdResep() ==
id_resep);
    if (removed) {
        System.out.println("Data resep berhasil dihapus!");
    } else {
        System.out.println("Resep dengan ID tersebut tidak ditemukan.");
    }
}
```

## 13. Stok



```
package model;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Stok {
    private int id_stok;
    private int id_obat;
    private int jumlah_stok;
    private String kadaluarsa;

    private static List<Stok> stokList = new ArrayList<>();
    private static Scanner input = new Scanner(System.in);

    public Stok(int id_stok, int id_obat, int jumlah_stok, String kadaluarsa) {
        this.id_stok = id_stok;
        this.id_obat = id_obat;
        this.jumlah_stok = jumlah_stok;
        this.kadaluarsa = kadaluarsa;
    }

    public int getIdStok() {
        return id_stok;
    }

    public void setIdStok(int id_stok) {
        this.id_stok = id_stok;
    }

    public int getIdObat() {
        return id_obat;
    }

    public void setIdObat(int id_obat) {
        this.id_obat = id_obat;
    }

    public int getJumlahStok() {
        return jumlah_stok;
    }

    public void setJumlahStok(int jumlah_stok) {
        this.jumlah_stok = jumlah_stok;
    }

    public String getKadaluarsa() {
        return kadaluarsa;
    }

    public void setKadaluarsa(String kadaluarsa) {
        this.kadaluarsa = kadaluarsa;
    }

    @Override
    public String toString() {
        return "\nID Stok: " + id_stok + " ID Obat: " + id_obat + "\nJumlah Stok: " + jumlah_stok +
               "\nKadaluarsa: " + kadaluarsa;
    }
}

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

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

        switch (pilihan) {
            case 1 -> tambahStok();
            case 2 -> lihatStok();
            case 3 -> updateStok();
            case 4 -> hapusStok();
            case 5 -> {
                System.out.println("Kembali ke menu utama...");
                return;
            }
            default -> System.out.println("Pilihan tidak valid. Silakan coba lagi.");
        }
    }
}

public static void tambahStok() {
    System.out.println("\n>>> Tambah Data Stok <<<");
    System.out.print("ID Stok: ");
    int id_stok = input.nextInt();
    input.nextLine();
    System.out.print("ID Obat: ");
    int id_obat = input.nextInt();
    input.nextLine();
    System.out.print("Jumlah Stok: ");
    int jumlah_stok = input.nextInt();
    input.nextLine();
    System.out.print("Kadaluarsa: ");
    String kadaluarsa = input.nextLine();

    stokList.add(new Stok(id_stok, id_obat, jumlah_stok, kadaluarsa));
    System.out.println("Data stok berhasil ditambahkan!");
}
```

```
public static void lihatStok() {
    System.out.println("\n>> Daftar Stok <<");
    if (stokList.isEmpty()) {
        System.out.println("Belum ada data stok.");
        return;
    }
    for (Stok stok : stokList) {
        System.out.println(stok);
    }
}

public static void updateStok() {
    System.out.println("\n>> Update Data Stok <<");
    System.out.print("Masukkan ID Stok yang ingin diupdate: ");
    int id_stok = input.nextInt();
    input.nextLine();

    for (Stok stok : stokList) {
        if (stok.getIdStok() == id_stok) {
            System.out.print("ID Obat Baru: ");
            stok.setIdObat(input.nextInt());
            input.nextLine();
            System.out.print("Jumlah Stok Baru: ");
            stok.setJumlahStok(input.nextInt());
            input.nextLine();
            System.out.print("Kadaluarsa Baru: ");
            stok.setKadaluarsa(input.nextLine());

            System.out.println("Data stok berhasil diperbarui!");
            return;
        }
    }
    System.out.println("Stok dengan ID tersebut tidak ditemukan.");
}

public static void hapusStok() {
    System.out.println("\n>> Hapus Data Stok <<");
    System.out.print("Masukkan ID Stok yang ingin dihapus: ");
    int id_stok = input.nextInt();
    input.nextLine();

    boolean removed = stokList.removeIf(stok -> stok.getIdStok() == id_stok);
    if (removed) {
        System.out.println("Data stok berhasil dihapus!");
    } else {
        System.out.println("Stok dengan ID tersebut tidak ditemukan.");
    }
}
```