

NAMA : MUHAMMAD CAHYA RIFQI

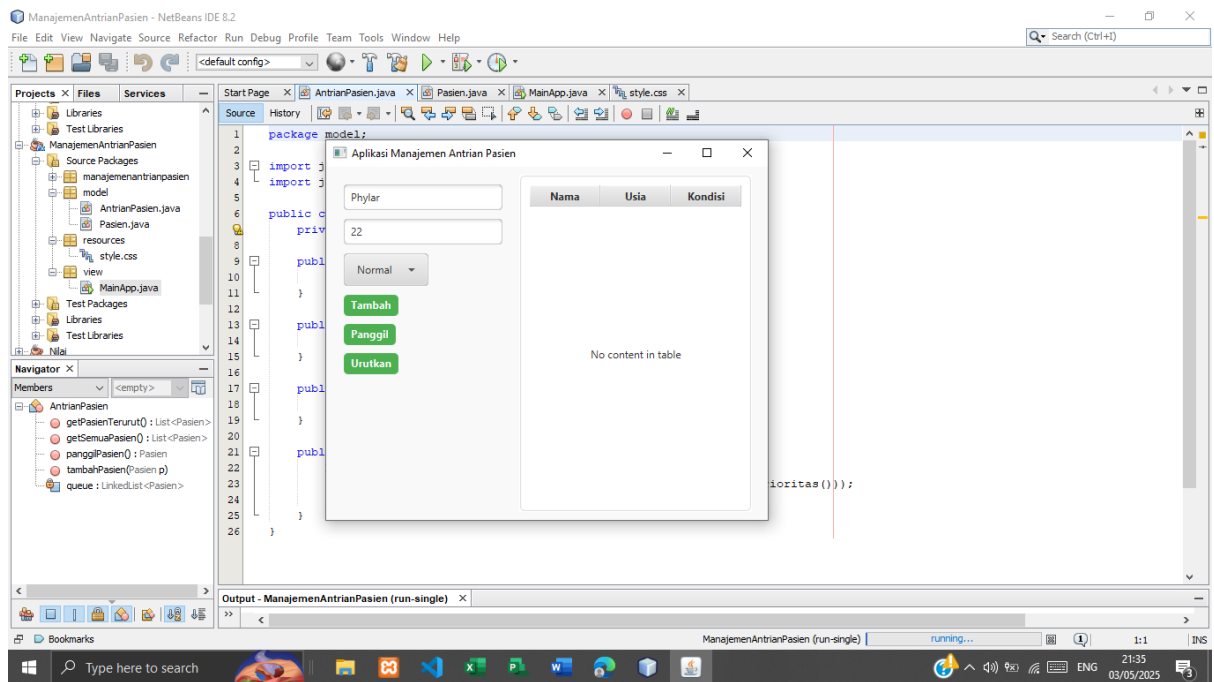
TUGAS

NIM : 211011401317

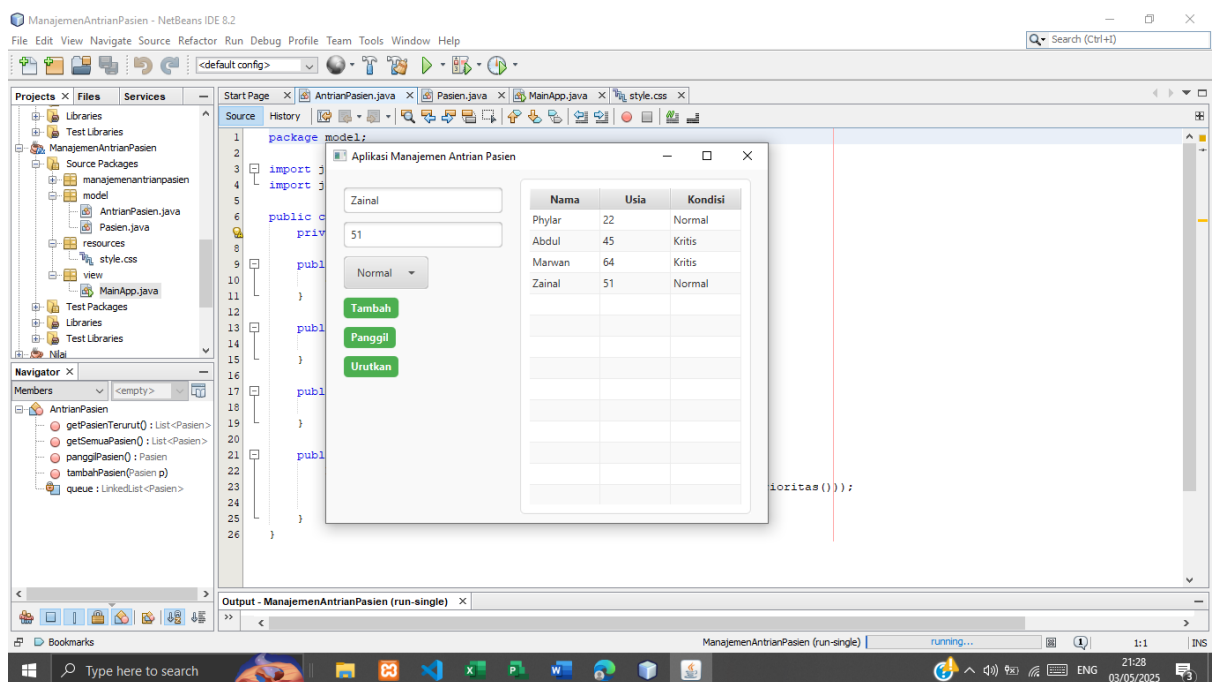
MATKUL : PEMOGRAMAN 2

A. QUEUE

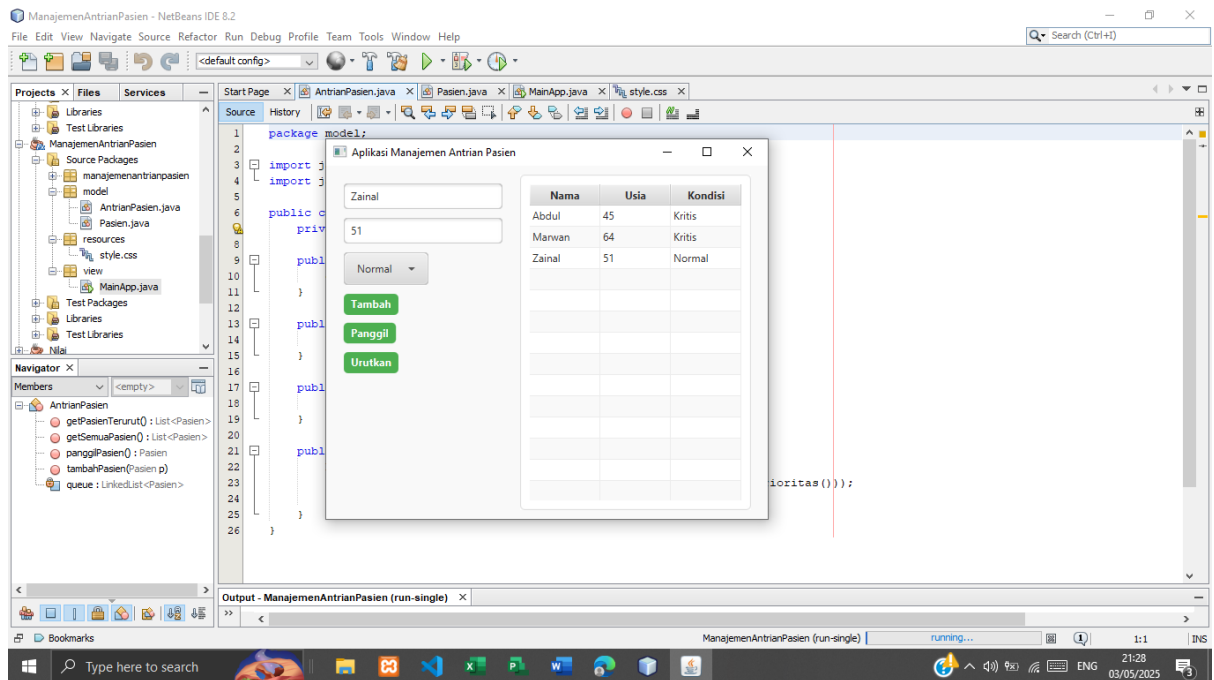
Menambahkan Pasien setelah diisi seperti contoh di bawah ini:



Ini adalah tampilan ketika semua Nama Pasien telah dimasukkan:

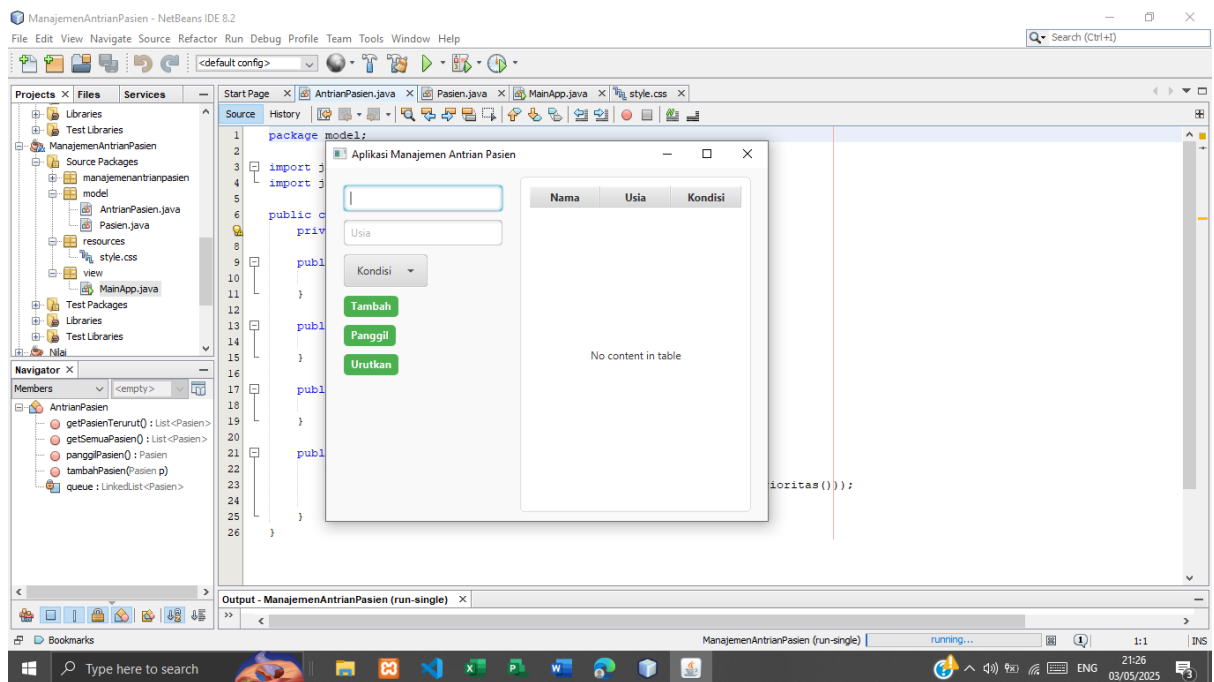


Ini adalah tampilan ketika Nama Pasien yang sudah dipanggil akan otomatis hilang, dimulai dari nama pasien yang dimasukkan terlebih dahulu:



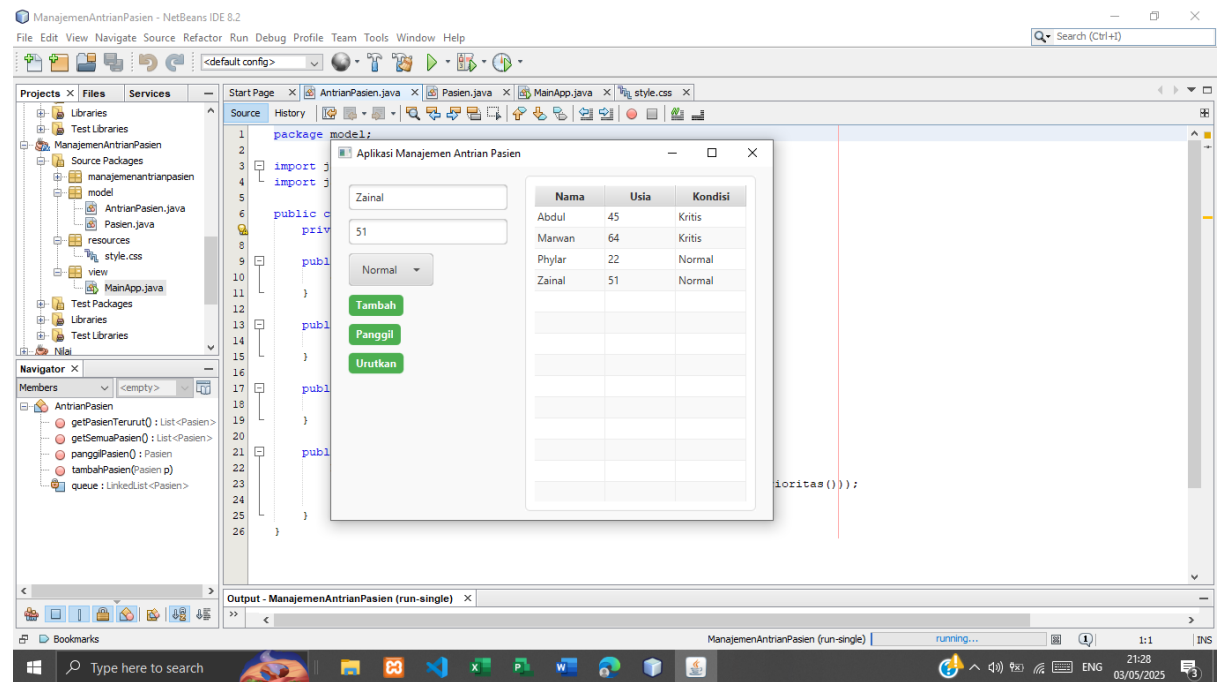
B. GUI JAVAFX

Ini adalah tampilan Form dan Tabel Kosong:



C. SORTING

Ini adalah Tabel Pasien yang sudah diurutkan sesuai abjad:

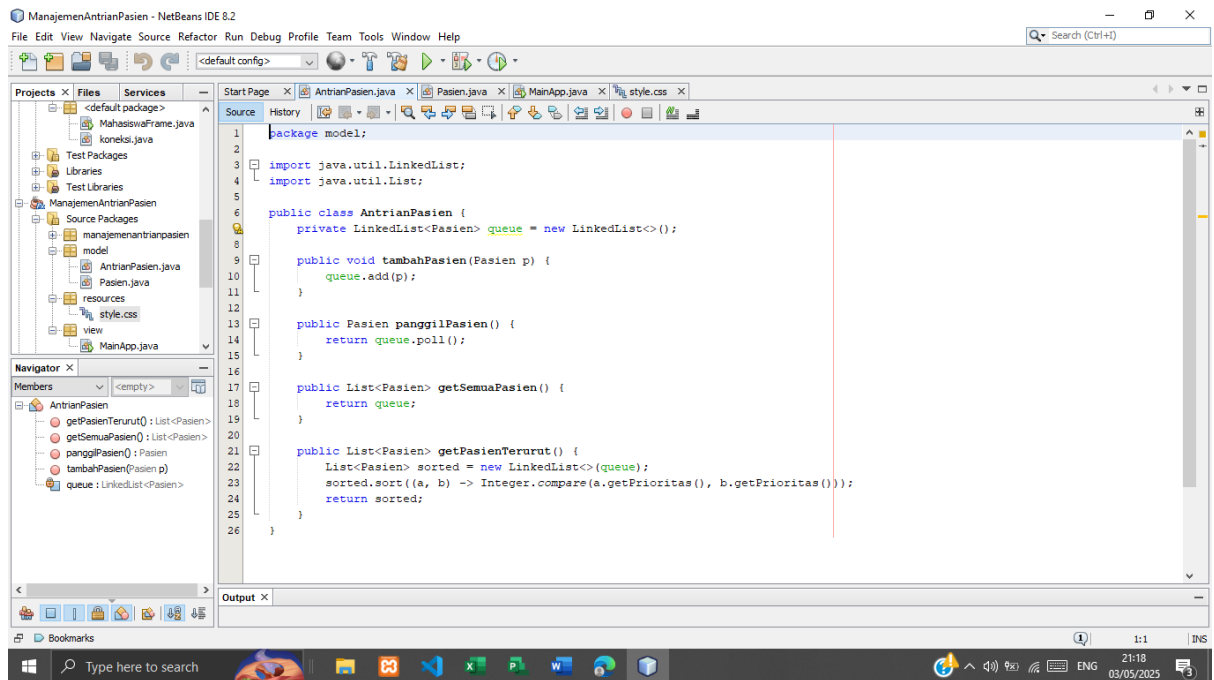


The screenshot shows the NetBeans IDE with a Java project named "ManajemenAntrianPasien". The main application window, titled "Aplikasi Manajemen Antrian Pasien", displays a table of patients sorted by name. The table has three columns: "Nama", "Usia", and "Kondisi". The data rows are: Abdul (45, Kritis), Marwan (64, Kritis), Phylar (22, Normal), and Zainal (51, Normal). The application also includes input fields for name and age, a dropdown menu for condition (currently set to "Normal"), and buttons for "Tambah", "Panggil", and "Urutkan". The background shows the Java source code for the application, including the package declaration and imports.

Nama	Usia	Kondisi
Abdul	45	Kritis
Marwan	64	Kritis
Phylar	22	Normal
Zainal	51	Normal

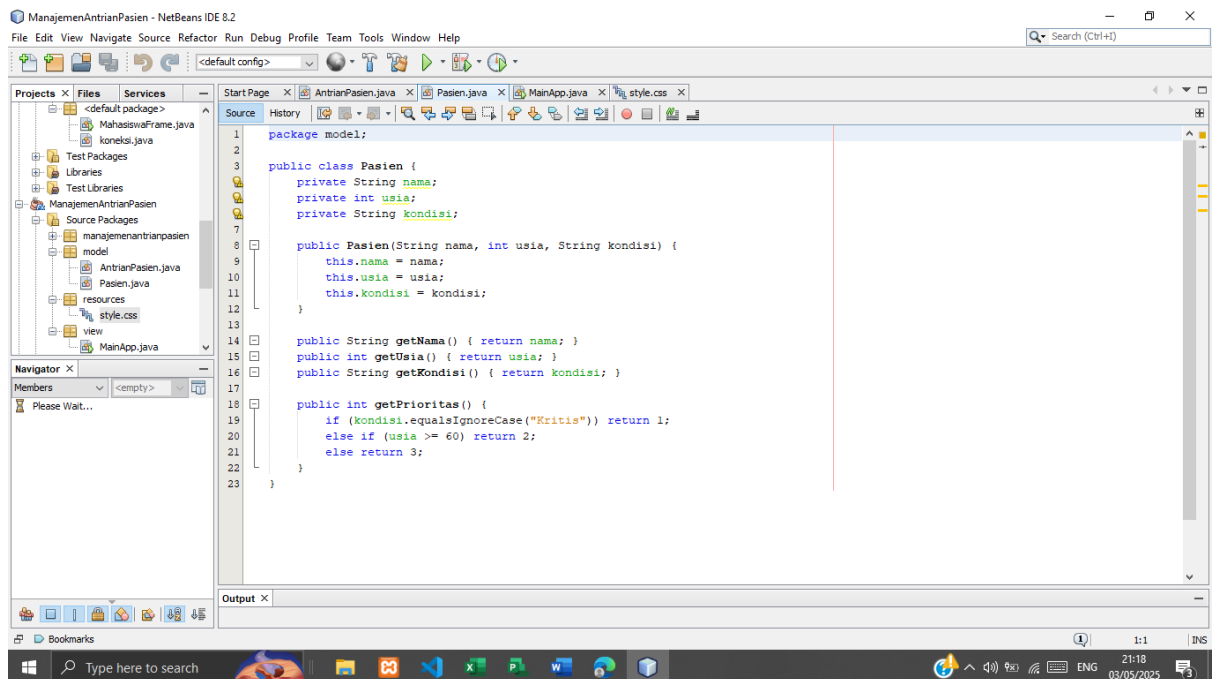
❖ SOURCE CODE:

A. AntrianPasien



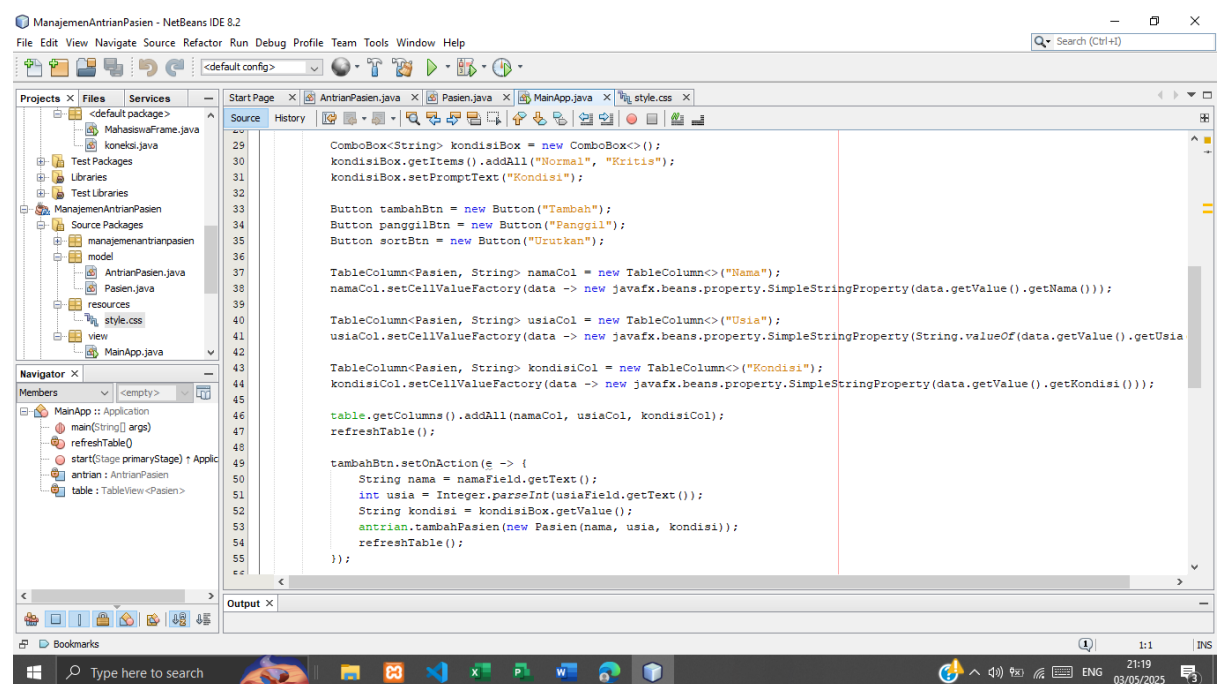
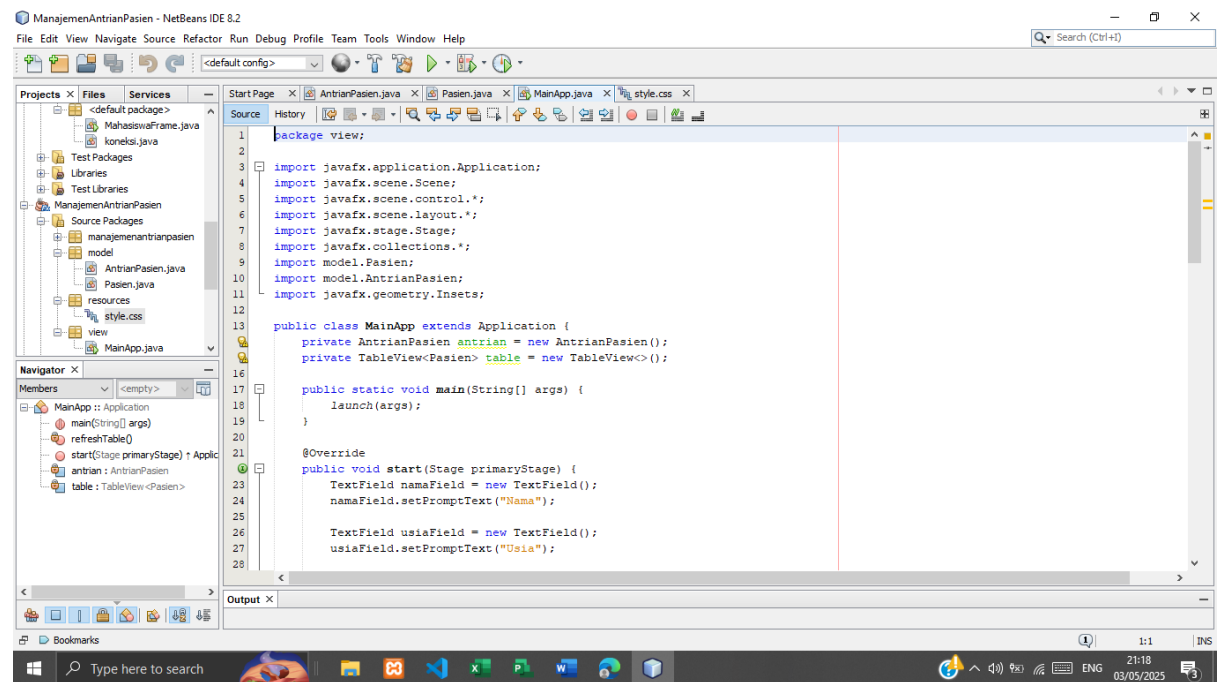
```
1 package model;
2
3 import java.util.LinkedList;
4 import java.util.List;
5
6 public class AntrianPasien {
7     private LinkedList<Pasien> queue = new LinkedList<>();
8
9     public void tambahPasien(Pasien p) {
10         queue.add(p);
11     }
12
13     public Pasien panggilPasien() {
14         return queue.poll();
15     }
16
17     public List<Pasien> getSemuaPasien() {
18         return queue;
19     }
20
21     public List<Pasien> getPasienTerurut() {
22         List<Pasien> sorted = new LinkedList<>(queue);
23         sorted.sort((a, b) -> Integer.compare(a.getPrioritas(), b.getPrioritas()));
24         return sorted;
25     }
26 }
```

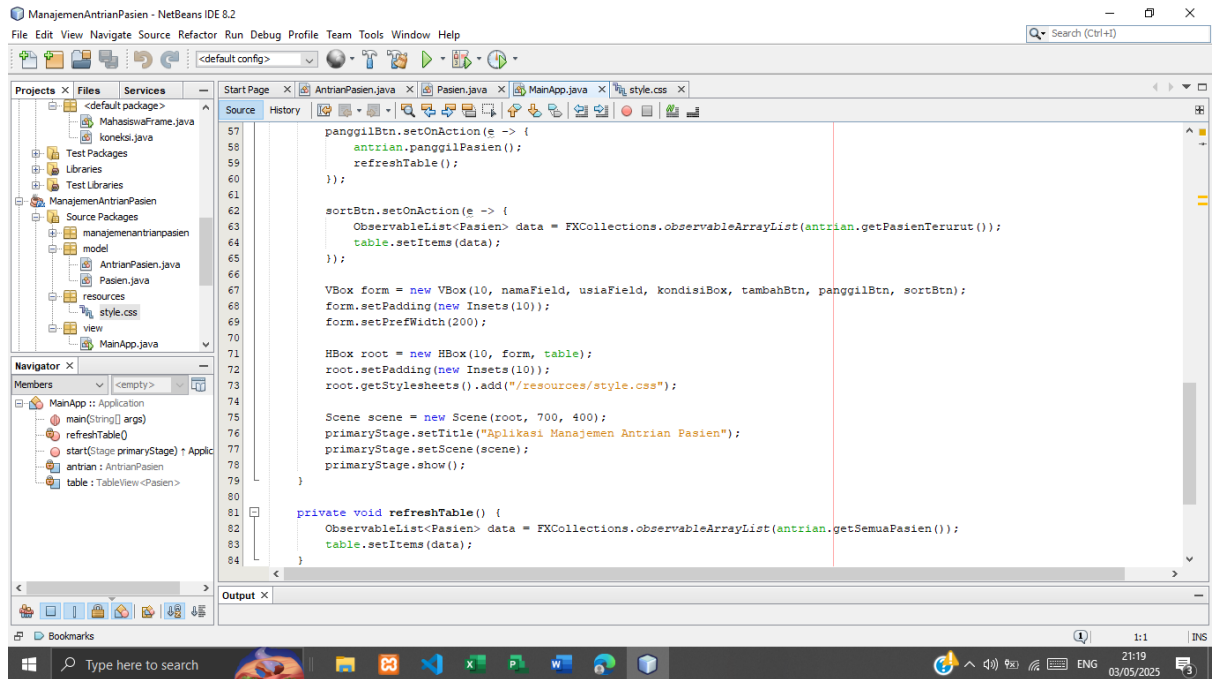
B. Pasien



```
1 package model;
2
3 public class Pasien {
4     private String nama;
5     private int usia;
6     private String kondisi;
7
8     public Pasien(String nama, int usia, String kondisi) {
9         this.nama = nama;
10        this.usia = usia;
11        this.kondisi = kondisi;
12    }
13
14    public String getNama() { return nama; }
15    public int getUsia() { return usia; }
16    public String getKondisi() { return kondisi; }
17
18    public int getPrioritas() {
19        if (kondisi.equalsIgnoreCase("Kritis")) return 1;
20        else if (usia >= 60) return 2;
21        else return 3;
22    }
23 }
```


C. MainApp





D. style

