**Implementasi Class dan Object**

Nama : Muhammad Naufal Nurmaryadi

Kelas : 2B – D3 Teknik Informatika

NIM : 241511053

**Deskripsi Singkat**

Aplikasi kali ini adalah tentang Rumah Sakit versi sederhana nya, menggunakan konsep OOP dalam Bahasa Java. Fitur yang tersedia dalam aplikasi ini pun cukup simple, hanya ada 4 entitas yaitu Pasien, Dokter, Obat, dan Rumah Sakit. Dalam fitur yang ada pun pengguna bisa menambah beberapa data seperti berikut ini:

* Nama dan alamat rumah sakit.
* IdPasien, nama, umur, penyakit.
* IdDokter, nama, spesialis.
* IdObat, nama, dosis.

1. Class Pasien
   1. class Pasien {
   2. private String idPasien;
   3. private String nama;
   4. private int umur;
   5. private String penyakit;
   6. public Pasien(String idPasien, String nama, int umur, String penyakit) {
   7. this.idPasien = idPasien;
   8. this.nama = nama;
   9. this.umur = umur;
   10. this.penyakit = penyakit;
   11. }
   12. public String getInfo() {
   13. return nama + " (ID: " + idPasien + ", Usia: " + umur + " tahun, Penyakit: " + penyakit + ")";
   14. }
   15. }

* Fungsi : Untuk data pasien
* Atribut : IdPasien, nama, umur, penyakit
* Getter : Mengambil nilai getInfo
* Metode : getInfo untuk menerima input dari user yang berisi data passien

1. Class Dokter
   1. class Dokter {
   2. private String idDokter;
   3. private String nama;
   4. private String spesialis;
   5. public Dokter(String idDokter, String nama, String spesialis) {
   6. this.idDokter = idDokter;
   7. this.nama = nama;
   8. this.spesialis = spesialis;
   9. }
   10. public void melayaniPasien(Pasien p) {
   11. System.out.println("Dokter " + nama +
   12. " (ID: " + idDokter + ", Spesialis: " + spesialis +
   13. ") sedang melayani pasien " + p.getInfo());
   14. }
   15. @Override
   16. public String toString() {
   17. return nama + " (ID: " + idDokter + ", Spesialis: " + spesialis + ")";
   18. }
   19. }

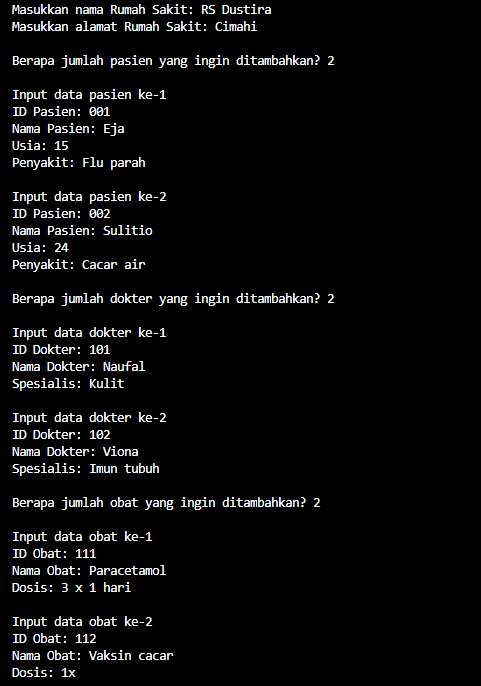
* **Fungsi**: Untuk mnyimpan data dokter dan menampilkan info dokter serta melayani pasien.
* **Atribut**: IdDokter, nama, specialis
* **Metode**: Dokter untuk membuat objek dokter baru, melayaniPasien untuk menampilkan pesan dokter melayani pasien, dan toString untuk enampilkan info dokter dalam bentuk string.

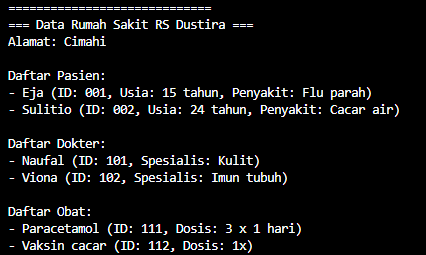
1. Class Obat
   1. class Obat {
   2. private String idObat;
   3. private String namaObat;
   4. private String dosis;
   5. public Obat(String idObat, String namaObat, String dosis) {
   6. this.idObat = idObat;
   7. this.namaObat = namaObat;
   8. this.dosis = dosis;
   9. }
   10. public String getInfoObat() {
   11. return namaObat + " (ID: " + idObat + ", Dosis: " + dosis + ")";
   12. }
   13. }
       * Fungsi : Menyimpan dan menampilkan data obat.
       * Atribut : namaPeminjam, barang, dan jumlahPeminjaman
       * Metode : Obat untuk konstruktor untuk inisialisasi objek obat dan getInfoObat untuk mengembalikan informasi detail obat dalam bentuk string..
2. Class RumahSakit
   1. class RumahSakit {
   2. private String namaRS;
   3. private String alamat;
   4. private ArrayList<Pasien> daftarPasien = new ArrayList<>();
   5. private ArrayList<Dokter> daftarDokter = new ArrayList<>();
   6. private ArrayList<Obat> daftarObat = new ArrayList<>();
   7. public RumahSakit(String namaRS, String alamat) {
   8. this.namaRS = namaRS;
   9. this.alamat = alamat;
   10. }
   11. public void tambahPasien(Pasien p) {
   12. daftarPasien.add(p);
   13. }
   14. public void tambahDokter(Dokter d) {
   15. daftarDokter.add(d);
   16. }
   17. public void tambahObat(Obat o) {
   18. daftarObat.add(o);
   19. }
   20. public void tampilkanData() {
   21. System.out.println("=== Data Rumah Sakit " + namaRS + " ===");
   22. System.out.println("Alamat: " + alamat);
   23. System.out.println("\nDaftar Pasien:");
   24. for (Pasien p : daftarPasien) {
   25. System.out.println("- " + p.getInfo());
   26. }
   27. System.out.println("\nDaftar Dokter:");
   28. for (Dokter d : daftarDokter) {
   29. System.out.println("- " + d);
   30. }
   31. System.out.println("\nDaftar Obat:");
   32. for (Obat o : daftarObat) {
   33. System.out.println("- " + o.getInfoObat());
   34. }
   35. }
   36. }

* Fungsi : Menyimpan mengelola data rumah sakit, pasien, dokter, dan obat.
* Atribut : namaRS, alamat, daftarPasien, daftarDokter, dan daftarObat
* Metode : rumatSakit untuk inisialisasi objek rummah sakit, tambahPasien untuk menambah data pasien ke daftar, begitu juga tambahDokter dan tambahObat, juga tampilkanData untuk menampilkan semua daftar data.

1. Class Main
2. public class Main {
3. public static void main(String[] args) {
4. Scanner input = new Scanner(System.in);
5. // Buat Rumah Sakit
6. System.out.print("Masukkan nama Rumah Sakit: ");
7. String namaRS = input.nextLine();
8. System.out.print("Masukkan alamat Rumah Sakit: ");
9. String alamatRS = input.nextLine();
10. RumahSakit rs = new RumahSakit(namaRS, alamatRS);
11. // Input jumlah pasien
12. System.out.print("\nBerapa jumlah pasien yang ingin ditambahkan? ");
13. int jumlahPasien = input.nextInt();
14. input.nextLine();
15. for (int i = 0; i < jumlahPasien; i++) {
16. System.out.println("\nInput data pasien ke-" + (i+1));
17. System.out.print("ID Pasien: ");
18. String idPasien = input.nextLine();
19. System.out.print("Nama Pasien: ");
20. String nama = input.nextLine();
21. System.out.print("Usia: ");
22. int usia = input.nextInt();
23. input.nextLine();
24. System.out.print("Penyakit: ");
25. String penyakit = input.nextLine();
26. Pasien p = new Pasien(idPasien, nama, usia, penyakit);
27. rs.tambahPasien(p);
28. }
29. // Input jumlah dokter
30. System.out.print("\nBerapa jumlah dokter yang ingin ditambahkan? ");
31. int jumlahDokter = input.nextInt();
32. input.nextLine();
33. for (int i = 0; i < jumlahDokter; i++) {
34. System.out.println("\nInput data dokter ke-" + (i+1));
35. System.out.print("ID Dokter: ");
36. String idDokter = input.nextLine();
37. System.out.print("Nama Dokter: ");
38. String nama = input.nextLine();
39. System.out.print("Spesialis: ");
40. String spesialis = input.nextLine();
41. Dokter d = new Dokter(idDokter, nama, spesialis);
42. rs.tambahDokter(d);
43. }
44. // Input jumlah obat
45. System.out.print("\nBerapa jumlah obat yang ingin ditambahkan? ");
46. int jumlahObat = input.nextInt();
47. input.nextLine();
48. for (int i = 0; i < jumlahObat; i++) {
49. System.out.println("\nInput data obat ke-" + (i+1));
50. System.out.print("ID Obat: ");
51. String idObat = input.nextLine();
52. System.out.print("Nama Obat: ");
53. String namaObat = input.nextLine();
54. System.out.print("Dosis: ");
55. String dosis = input.nextLine();
56. Obat o = new Obat(idObat, namaObat, dosis);
57. rs.tambahObat(o);
58. }
59. // Tampilkan semua data
60. System.out.println("\n=============================");
61. rs.tampilkanData();
62. }
63. }
    * + Fungsi : Menjalankan program, menerima input dari user, membuat objek, dan menampilkan data.

Hasil program:





LINK GITHUB:

<https://github.com/itsspall/Sistem-Rumah-Sakit-Sederhana>