

```

1 import java.util.ArrayList;
2 import java.util.Scanner;
3
4 // FINAL CLASS
5 final class Lapangan {
6     private int id;
7     private String nama;
8     private double harga;
9
10    public Lapangan(int id, String nama, double harga) {
11        this.id = id;
12        this.nama = nama;
13        this.harga = harga;
14    }
15
16    // Getter dan Setter
17    public int getId() { return id; }
18    public String getNama() { return nama; }
19    public double getHarga() { return harga; }
20
21    public void setNama(String nama) { this.nama = nama; }
22    public void setHarga(double harga) { this.harga = harga; }
23
24    // FINAL METHOD
25    public final void tampilkanInfo() {
26        System.out.println("ID : " + id + ", Nama : " + nama + ", Harga: Rp " + harga + " /jam");
27    }
28 }
29
30 // ABSTRACT CLASS
31 abstract class Reservasi {
32     private int id;
33     private final String namaLapangan; // FINAL ATTRIBUTE
34     private String tanggal;
35     private int durasi;
36     private Lapangan lapangan;
37
38     public Reservasi(int id, String namaLapangan, String tanggal, int durasi, Lapangan lapangan) {
39         this.id = id;
40         this.namaLapangan = namaLapangan;
41         this.tanggal = tanggal;
42         this.durasi = durasi;
43         this.lapangan = lapangan;
44     }
45
46     // Getter dan Setter
47     public int getId() { return id; }
48     public String getNamaLapangan() { return namaLapangan; }
49     public String getTanggal() { return tanggal; }
50     public int getDurasi() { return durasi; }
51     public Lapangan getLapangan() { return lapangan; }
52
53     public void setTanggal(String tanggal) { this.tanggal = tanggal; }
54     public void setDurasi(int durasi) { this.durasi = durasi; }
55
56     public void tampilkanInfo() {
57         System.out.println("Reservasi : " + id + ", Lapangan : " + namaLapangan +
58             ", Tanggal : " + tanggal + ", Durasi : " + durasi + " jam, Lapangan : " + lapangan.getNama());
59     }
60
61     // Abstract method
62     public abstract void cetakDetail();
63 }
64
65 // Anak class untuk override abstract method
66 class ReservasiLapangan extends Reservasi {
67     public ReservasiLapangan(int id, String namaLapangan, String tanggal, int durasi, Lapangan lapangan) {
68         super(id, namaLapangan, tanggal, durasi, lapangan);
69     }
70
71     @Override
72     public void cetakDetail() {
73         tampilkanInfo();
74         System.out.println("Total Biaya: Rp " + getDurasi() * getLapangan().getHarga());
75     }
76 }
77
78 public class Main {
79     public static void main(String[] args) {
80         ArrayList<Lapangan> daftarLapangan = new ArrayList<>();
81         ArrayList<Reservasi> daftarReservasi = new ArrayList<>();
82         int idReservasi = 1;
83
84         // Inisialisasi Lapangan
85         daftarLapangan.add(new Lapangan(1, "Lapangan A", 5000));
86         daftarLapangan.add(new Lapangan(2, "Lapangan B", 8000));
87
88         while (true) {
89             System.out.println("Menu Sistem Reservasi Lapangan Futsal:");
90             System.out.println("1. Daftar Lapangan");
91             System.out.println("2. Lihat Daftar Lapangan");
92             System.out.println("3. Lihat Reservasi");
93             System.out.println("4. Lihat Reservasi");
94             System.out.println("5. Buat Reservasi");
95             System.out.println("6. Hapus Reservasi");
96             System.out.println("7. Lihat Menu");
97
98             String pilihan = input.nextLine();
99             if (pilihan.matches("\\d+")) {
100                 System.out.println("Input harus valid: harus angka 1-7");
101                 continue;
102             }
103
104             int pilihan = Integer.parseInt(pilihan);
105
106             switch (pilihan) {
107                 case 1 -> lihatLapangan();
108                 case 2 -> tambahReservasi();
109                 case 3 -> lihatReservasi();
110                 case 4 -> editReservasi();
111                 case 5 -> buatReservasi();
112                 case 6 -> {
113                     System.out.println("Input harus valid");
114                     return;
115                 }
116                 default -> System.out.println("Pilihan tidak valid");
117             }
118         }
119
120         static void lihatLapangan() {
121             System.out.println("Daftar Lapangan:");
122             for (Lapangan l : daftarLapangan) {
123                 l.tampilkanInfo();
124             }
125         }
126
127         static void tambahReservasi() {
128             System.out.println("Nama Lapangan: ");
129             String nama = input.nextLine();
130             System.out.println("Tanggal (YYYY-MM-DD): ");
131             String tanggal = input.nextLine();
132             System.out.println("Durasi (jam): ");
133             int durasi = input.nextInt();
134             input.nextLine();
135             Lapangan lapangan;
136             System.out.println("Pilih ID Lapangan: ");
137             int idLap = input.nextInt();
138             input.nextLine();
139             lapangan = daftarLapangan[idLap];
140
141             if (lapangan != null) {
142                 Reservasi reservasi = new ReservasiLapangan(idReservasi, nama, tanggal, durasi, lapangan);
143                 daftarReservasi.add(reservasi);
144                 System.out.println("Reservasi berhasil ditambahkan");
145             } else {
146                 System.out.println("Lapangan tidak ditemukan");
147             }
148             idReservasi++;
149         }
150
151         static void lihatReservasi() {
152             if (daftarReservasi.isEmpty()) {
153                 System.out.println("Tidak ada reservasi");
154             } else {
155                 System.out.println("Daftar Reservasi:");
156                 for (Reservasi r : daftarReservasi) {
157                     r.cetakDetail();
158                 }
159                 System.out.println("-----");
160             }
161         }
162
163         static void editReservasi() {
164             lihatReservasi();
165             if (daftarReservasi.isEmpty()) return;
166
167             System.out.println("Masukkan ID reservasi yang ingin diedit: ");
168             int id = input.nextInt();
169             input.nextLine();
170
171             Reservasi reservasi = cariReservasi(id);
172             if (reservasi != null) {
173                 System.out.println("Tanggal Baru (YYYY-MM-DD): ");
174                 reservasi.setTanggal(input.nextLine());
175                 System.out.println("Durasi Baru (jam): ");
176                 reservasi.setDurasi(input.nextInt());
177                 input.nextLine();
178                 System.out.println("Reservasi berhasil diperbarui");
179             } else {
180                 System.out.println("Reservasi tidak ditemukan");
181             }
182         }
183
184         static void hapusReservasi() {
185             lihatReservasi();
186             if (daftarReservasi.isEmpty()) return;
187
188             System.out.println("Masukkan ID reservasi yang ingin dihapus: ");
189             int id = input.nextInt();
190             input.nextLine();
191
192             Reservasi reservasi = cariReservasi(id);
193             if (reservasi != null) {
194                 daftarReservasi.remove(reservasi);
195                 System.out.println("Reservasi berhasil dihapus");
196             } else {
197                 System.out.println("Reservasi tidak ditemukan");
198             }
199         }
200
201         static Lapangan cariLapangan(int id) {
202             for (Lapangan l : daftarLapangan) {
203                 if (l.getId() == id) return l;
204             }
205             return null;
206         }
207
208         static Reservasi cariReservasi(int id) {
209             for (Reservasi r : daftarReservasi) {
210                 if (r.getId() == id) return r;
211             }
212             return null;
213         }
214     }
215 }

```