

Nama : Tengku Nabila Zaskiah

NIM : 2309106104

Kelas : Informatika C'23

Posttest 4 Pemrograman Berorientasi Objek

SS Program

1. Menampilkan menu masuk sebagai Admin/Penumpang

```
Masukkan peran Anda (Admin/Penumpang):
```

2. Menampilkan menu admin

```
Masukkan peran Anda (Admin/Penumpang):  
Admin
```

- ```
1. Tambah Penerbangan
2. Hapus Penerbangan
3. Lihat Penerbangan
4. Keluar
```

```
Pilih opsi:
```

3. Menampilkan menu tambah penerbangan pada admin

```
1. Tambah Penerbangan
2. Hapus Penerbangan
3. Lihat Penerbangan
4. Keluar
Pilih opsi: 1
Flight ID: 01
Asal: Samarinda
Tujuan: Bojong Gede
Tanggal: 1 Januari 2025
Maskapai: Garuda Indonesia
Harga: 2.000.000
Penerbangan berhasil ditambahkan!
```

4. Menampilkan menu hapus penerbangan pada admin

```
1. Tambah Penerbangan
2. Hapus Penerbangan
3. Lihat Penerbangan
4. Keluar
Pilih opsi: 2

+-----+-----+-----+-----+-----+-----+
| Flight ID | Asal | Tujuan | Tanggal | Maskapai | Harga |
+-----+-----+-----+-----+-----+-----+
| 01 | Samarinda | Bojong Gede | 1 Januari 2025 | Garuda Indonesia | Rp2000000,00 |
+-----+-----+-----+-----+-----+-----+
Masukkan Flight ID yang ingin dihapus: 01

Penerbangan 01 berhasil dihapus.
```

5. Menampilkan menu lihat penerbangan pada admin

```

1. Tambah Penerbangan
2. Hapus Penerbangan
3. Lihat Penerbangan
4. Keluar
Pilih opsi: 3

+-----+-----+-----+-----+-----+-----+
| Flight ID | Asal | Tujuan | Tanggal | Maskapai | Harga |
+-----+-----+-----+-----+-----+-----+
| 01 | Samarinda | Bohong Gede | 1 Januari 2025 | Garuda Indonesia | Rp2000000,00 |
+-----+-----+-----+-----+-----+-----+

```

6. Keluar pada menu admin akan masuk ke menu utama

```

1. Tambah Penerbangan
2. Hapus Penerbangan
3. Lihat Penerbangan
4. Keluar
Pilih opsi: 4
Kembali ke menu utama...

=====
Airfly Ticket Booking System
=====
1. Tambah Penerbangan (Admin)
2. Lihat Penerbangan
3. Buat Reservasi
4. Lihat Reservasi
5. Verifikasi Pembayaran
6. Keluar
Masukkan pilihan: █

```

7. Langsung ke menu buat reservasi karena menu 1&2 sama seperti di menu admin sebelumnya

```

=====
Airfly Ticket Booking System
=====
1. Tambah Penerbangan (Admin)
2. Lihat Penerbangan
3. Buat Reservasi
4. Lihat Reservasi
5. Verifikasi Pembayaran
6. Keluar
Masukkan pilihan: 3

+-----+-----+-----+-----+-----+-----+
| Flight ID | Asal | Tujuan | Tanggal | Maskapai | Harga |
+-----+-----+-----+-----+-----+-----+
| 01 | Samarinda | Bojong Gede | 1 Januari 2025 | Garuda Indonesia | Rp2000000,00 |
+-----+-----+-----+-----+-----+-----+
Masukkan Reservasi ID: 122
Masukkan Flight ID: 01
Masukkan nama penumpang: Nabila
Masukkan Harga: 2000000
Reservasi berhasil dibuat dengan harga Rp 2000000.0

```

8. Menampilkan menu lihat rservasi

```

=====
Airfly Ticket Booking System
=====
1. Tambah Penerbangan (Admin)
2. Lihat Penerbangan
3. Buat Reservasi
4. Lihat Reservasi
5. Verifikasi Pembayaran
6. Keluar
Masukkan pilihan: 4

+-----+-----+-----+-----+
| Reservasi ID | Flight ID | Nama Penumpang | Status |
+-----+-----+-----+-----+
| 122 | 01 | Nabila | Pending |
+-----+-----+-----+-----+

```

9. Menampilkan menu verifikasi pembayaran, lalu mengecek Kembali status pembayarannya apakah sudah berubah atau belum

```

=====
Airfly Ticket Booking System
=====
1. Tambah Penerbangan (Admin)
2. Lihat Penerbangan
3. Buat Reservasi
4. Lihat Reservasi
5. Verifikasi Pembayaran
6. Keluar
Masukkan pilihan: 5
Masukkan Reservasi ID untuk verifikasi: 122

Pembayaran berhasil diverifikasi!

=====
Airfly Ticket Booking System
=====
1. Tambah Penerbangan (Admin)
2. Lihat Penerbangan
3. Buat Reservasi
4. Lihat Reservasi
5. Verifikasi Pembayaran
6. Keluar
Masukkan pilihan: 4

+-----+-----+-----+-----+
| Reservasi ID | Flight ID | Nama Penumpang | Status |
+-----+-----+-----+-----+
| 122 | 01 | Nabila | Lunas |
+-----+-----+-----+-----+

```

10. Menampilkan menu keluar, jika memilih “y” maka akan keluar habis dari program, tetapi jika memilih “n” maka tidak jadi keluar dari program

```

=====
Airfly Ticket Booking System
=====
1. Tambah Penerbangan (Admin)
2. Lihat Penerbangan
3. Buat Reservasi
4. Lihat Reservasi
5. Verifikasi Pembayaran
6. Keluar
Masukkan pilihan: 6
Apakah Anda yakin ingin keluar? (y/n):

```

## Source Code :

```
1 import java.util.ArrayList;
2 import java.util.List;
3 import java.util.ListIterator;
4 import java.util.Scanner;
5
6 // Class Pengguna dengan Encapsulation
7 abstract class Pengguna {
8 protected String userID;
9 protected String nama;
10
11 public Pengguna(String userID, String nama) {
12 this.userID = userID;
13 this.nama = nama;
14 }
15
16 public String getUserID() {
17 return userID;
18 }
19
20 public String getNama() {
21 return nama;
22 }
23
24 public abstract void menu(SistemPemesanan sistem, Scanner scanner);
25 }
26
27 // Subclass Admin
28 class Admin extends Pengguna {
29 public Admin(String userID, String nama) {
30 super(userID, nama);
31 }
32
33 @Override
34 public void menu(SistemPemesanan sistem, Scanner scanner) {
35 int choice;
36 do {
37 System.out.println("\n1. Tambah Penerbangan");
38 System.out.println("2. Hapus Penerbangan");
39 System.out.println("3. Lihat Penerbangan");
40 System.out.println("4. Keluar");
41 System.out.print("Pilih opsi: ");
42
43 try {
44 choice = scanner.nextInt();
45 scanner.nextLine();
46
47 switch (choice) {
48 case 1:
49 System.out.print("Flight ID: ");
50 String fid = scanner.nextLine();
51 System.out.print("Asal: ");
52 String asal = scanner.nextLine();
53 System.out.print("Tujuan: ");
54 String tujuan = scanner.nextLine();
55 System.out.print("Tanggal: ");
56 String tanggal = scanner.nextLine();
57 System.out.print("Maskapai: ");
58 String maskapai = scanner.nextLine();
59 System.out.print("Harga: ");
60 double harga = scanner.nextDouble();
61 scanner.nextLine();
62 sistem.tambahPenerbangan(fid, asal, tujuan, tanggal, maskapai, harga);
63 break;
64 case 2:
65 sistem.lihatPenerbangan();
66 System.out.print("Masukkan Flight ID yang ingin dihapus: ");
67 String hapusID = scanner.nextLine();
68 sistem.hapusPenerbangan(hapusID);
69 break;
70 case 3:
71 sistem.lihatPenerbangan();
72 break;
73 case 4:
74 System.out.println("Kembali ke menu utama...");
75 return; // Kembali ke menu utama
76 default:
77 System.out.println("Pilihan tidak valid.");
78 }
79 } catch (Exception e) {
80 System.out.println("Input tidak valid! Masukkan angka.");
81 scanner.nextLine();
82 }
83 } while (true);
84 }
85 }
86
87
```

```

86
87
88 // Subclass Penumpang
89 class Penumpang extends Pengguna {
90 public Penumpang(String userID, String nama) {
91 super(userID, nama);
92 }
93
94 @Override
95 public void menu(SistemPemesanan sistem, Scanner scanner) {
96 int choice;
97 do {
98 System.out.println("\n1. Lihat Penerbangan");
99 System.out.println("2. Buat Reservasi");
100 System.out.println("3. Lihat Reservasi");
101 System.out.println("4. Bayar Reservasi");
102 System.out.println("5. Keluar");
103 System.out.print("Pilih opsi: ");
104
105 try {
106 choice = scanner.nextInt();
107 scanner.nextLine();
108
109 switch (choice) {
110 case 1:
111 sistem.lihatPenerbangan();
112 break;
113 case 2:
114 sistem.lihatPenerbangan();
115 System.out.print("Masukkan Flight ID yang ingin dipesan: ");
116 String flightID = scanner.nextLine();
117 System.out.print("Masukkan Reservasi ID: ");
118 String reservasiID = scanner.nextLine();
119 sistem.buatReservasi(reservasiID, flightID, nama, 0.0);
120 break;
121 case 3:
122 sistem.lihatReservasi();
123 break;
124 case 4:
125 sistem.lihatReservasi();
126 System.out.print("Masukkan Reservasi ID yang ingin dibayar: ");
127 String bayarID = scanner.nextLine();
128 sistem.verifikasiPembayaran(bayarID);
129 break;
130 case 5:
131 System.out.println("Kembali ke menu utama...");
132 return; // Keluar dari menu
133 default:
134 System.out.println("Pilihan tidak valid.");
135 }
136 } catch (Exception e) {
137 System.out.println("Input tidak valid! Masukkan angka.");
138 scanner.nextLine();
139 }
140 } while (true);
141 }
142 }
143
144 class Flight {
145 String id;
146 Flight(String id) {
147 this.id = id;
148 }
149 }
150
151 // Class Pembayaran dengan Encapsulation
152 class Pembayaran {
153 private String reservasiID;
154 private double jumlah;
155 private boolean status;
156
157 public Pembayaran(String reservasiID, double jumlah) {
158 this.reservasiID = reservasiID;
159 this.jumlah = jumlah;
160 this.status = false;
161 }
162
163 // Getter untuk jumlah
164 public double getJumlah() {
165 return jumlah;
166 }
167
168 // Metode untuk menampilkan jumlah pembayaran
169 public void tampilkanJumlah() {
170 System.out.println("Jumlah pembayaran: Rp " + jumlah);
171 }
172
173 // Getter untuk status pembayaran
174 public boolean isLunas() {
175 return status;
176 }
177
178 // Metode untuk konfirmasi pembayaran
179 public void konfirmasiPembayaran() {
180 this.status = true;
181 }
182
183 public String getReservasiID() {
184 return reservasiID;
185 }
186 }
187

```

```

187
188 // Class Penerbangan dengan Encapsulation
189 class Penerbangan {
190 private String flightID, asal, tujuan, tanggal, maskapai;
191 private double harga;
192
193 public Penerbangan(String flightID, String asal, String tujuan, String tanggal, String maskapai, double harga) {
194 this.flightID = flightID;
195 this.asal = asal;
196 this.tujuan = tujuan;
197 this.tanggal = tanggal;
198 this.maskapai = maskapai;
199 this.harga = harga;
200 }
201
202 // Getter untuk Flight ID
203 public String getFlightID() {
204 return flightID;
205 }
206
207 // Getter untuk Harga
208 public double getHarga() {
209 return harga;
210 }
211
212 @Override
213 public String toString() {
214 return String.format("| %-10s | %-10s | %-10s | %-12s | %-10s | Rp%-10.2f |", flightID, asal, tujuan, tanggal, maskapai, harga);
215 }
216 }
217
218 // Class Reservasi dengan Encapsulation
219 class Reservasi {
220 private String reservasiID, flightID, namaPenumpang;
221 private boolean statusPembayaran;
222
223 public Reservasi(String reservasiID, String flightID, String namaPenumpang) {
224 this.reservasiID = reservasiID;
225 this.flightID = flightID;
226 this.namaPenumpang = namaPenumpang;
227 this.statusPembayaran = false;
228 }
229
230 // Getter untuk reservasiID
231 public String getReservasiID() {
232 return reservasiID;
233 }
234
235 // Getter untuk status pembayaran
236 public boolean isLunas() {
237 return statusPembayaran;
238 }
239
240 // Setter untuk mengubah status pembayaran
241 public void setStatusPembayaran(boolean status) {
242 this.statusPembayaran = status;
243 }
244
245 @Override
246 public String toString() {
247 return String.format("| %-12s | %-10s | %-15s | %-10s |", reservasiID, flightID, namaPenumpang, (statusPembayaran ? "Lunas" : "Pending"));
248 }
249 }
250
251 // Sistem Pemesanan dengan perubahan Encapsulation
252 class SistemPemesanan {
253 private ArrayList<Penerbangan> penerbanganList = new ArrayList<>();
254 private ArrayList<Reservasi> reservasiList = new ArrayList<>();
255 private ArrayList<Pembayaran> pembayaranList = new ArrayList<>();
256
257 public void tambahPenerbangan(String flightID, String asal, String tujuan, String tanggal, String maskapai, double harga) {
258 for (Penerbangan p : penerbanganList) {
259 if (p.getFlightID().equals(flightID)) {
260 System.out.println("Flight ID sudah ada!");
261 return;
262 }
263 }
264 penerbanganList.add(new Penerbangan(flightID, asal, tujuan, tanggal, maskapai, harga));
265 System.out.println("Penerbangan berhasil ditambahkan!");
266 }
267
268 // Getter untuk penerbanganList

```

```

267
268 // Menghapus penerbangan
269 public void hapusPenerbangan(String flightID) {
270 if (penerbanganList.isEmpty()) {
271 System.out.println("\nTidak ada penerbangan yang tersedia.");
272 return;
273 }
274
275 ListIterator<Penerbangan> iterator = penerbanganList.listIterator();
276 while (iterator.hasNext()) {
277 Penerbangan p = iterator.next();
278 if (p.getFlightID().equals(flightID)) {
279 iterator.remove();
280 System.out.println("\nPenerbangan " + flightID + " berhasil dihapus.");
281 return;
282 }
283 }
284 System.out.println("\nPenerbangan tidak ditemukan!");
285 }
286
287 public void lihatPenerbangan() {
288 if (penerbanganList.isEmpty()) {
289 System.out.println("\nTidak ada penerbangan yang tersedia.");
290 } else {
291 System.out.println("\n+-----+-----+-----+-----+-----+");
292 System.out.println("| Flight ID | Asal | Tujuan | Tanggal | Maskapai | Harga |");
293 System.out.println("+-----+-----+-----+-----+-----+");
294 for (Penerbangan p : penerbanganList) {
295 System.out.println(p);
296 }
297 System.out.println("+-----+-----+-----+-----+-----+");
298 }
299 }
300
301 // Versi utama: dengan harga eksplisit (walau tidak digunakan karena pakai harga dari penerbangan)
302 public void buatReservasi(String reservasiID, String flightID, String namaPenumpang, double hargaReservasi) {
303 if (isReservasiIDExist(reservasiID)) {
304 System.out.println("Reservasi ID sudah digunakan!");
305 return;
306 }
307
308 Penerbangan flight = cariPenerbangan(flightID);
309 if (flight != null) {
310 reservasiList.add(new Reservasi(reservasiID, flightID, namaPenumpang));
311 pembayaranList.add(new Pembayaran(reservasiID, flight.getHarga()));
312 System.out.println("Reservasi berhasil dibuat dengan harga Rp " + flight.getHarga());
313 } else {
314 System.out.println("Penerbangan tidak ditemukan!");
315 }
316 }
317
318 // Overloading 1: tanpa harga, akan otomatis ambil dari harga penerbangan
319 public void buatReservasi(String reservasiID, String flightID, String namaPenumpang) {
320 buatReservasi(reservasiID, flightID, namaPenumpang, 0);
321 }
322
323 // Overloading 2: tanpa flightID (misal untuk pre-booking atau draft reservasi)
324 public void buatReservasi(String reservasiID, String namaPenumpang) {
325 if (isReservasiIDExist(reservasiID)) {
326 System.out.println("Reservasi ID sudah digunakan!");
327 return;
328 }
329 reservasiList.add(new Reservasi(reservasiID, "Belum-Ditentukan", namaPenumpang));
330 pembayaranList.add(new Pembayaran(reservasiID, 0));
331 System.out.println("Reservasi sementara berhasil dibuat untuk " + namaPenumpang);
332 }
333
334 // Fungsi bantu
335 private boolean isReservasiIDExist(String reservasiID) {
336 for (Reservasi r : reservasiList) {
337 if (r.getReservasiID().equals(reservasiID)) {
338 return true;
339 }
340 }
341 return false;
342 }
343
344 private Penerbangan cariPenerbangan(String flightID) {
345 for (Penerbangan p : penerbanganList) {
346 if (p.getFlightID().equals(flightID)) {
347 return p;
348 }
349 }
350 return null;
351 }
352

```

```

352
353 public void verifikasiPembayaran(String reservasiID) {
354 for (Pembayaran p : pembayaranList) {
355 if (p.getReservasiID().equals(reservasiID)) {
356 p.konfirmasiPembayaran();
357 for (Reservasi r : reservasiList) {
358 if (r.getReservasiID().equals(reservasiID)) {
359 r.setStatusPembayaran(true);
360 System.out.println("\nPembayaran berhasil diverifikasi!");
361 return;
362 }
363 }
364 }
365 }
366 System.out.println("\nReservasi tidak ditemukan atau pembayaran belum dilakukan.");
367 }
368
369 public void lihatReservasi() {
370 if (reservasiList.isEmpty()) {
371 System.out.println("\nTidak ada reservasi.");
372 } else {
373 System.out.println("\n+-----+-----+-----+-----+");
374 System.out.println("| Reservasi ID | Flight ID | Nama Penumpang | Status |");
375 System.out.println("+-----+-----+-----+-----+");
376 for (Reservasi r : reservasiList) {
377 System.out.println(r);
378 }
379 System.out.println("+-----+-----+-----+-----+");
380 }
381 }
382 }
383
384 // Main Program
385 public class AirflyTicketPSystem {
386 public static void main(String[] args) {
387 Scanner scanner = new Scanner(System.in);
388 SistemPemesanan sistem = new SistemPemesanan();
389 boolean running = true; // Tambahkan variabel running agar tidak error
390
391 do {
392 System.out.println("Masukkan peran Anda (Admin/Penumpang): ");
393 String role = scanner.nextLine();
394
395 if (role.equalsIgnoreCase("Admin")) {
396 Admin admin = new Admin("A001", "Admin1");
397 admin.menu(sistem, scanner);
398 } else {
399 Penumpang penumpang = new Penumpang("P001", "Nabila");
400 penumpang.menu(sistem, scanner);
401 }
402 }
403
404 while (running) {
405 System.out.println("\n=====");
406 System.out.println(" Airfly Ticket Booking System ");
407 System.out.println("=====");
408 System.out.println("1. Tambah Penerbangan (Admin)");
409 System.out.println("2. Lihat Penerbangan");
410 System.out.println("3. Buat Reservasi");
411 System.out.println("4. Lihat Reservasi");
412 System.out.println("5. Verifikasi Pembayaran");
413 System.out.println("6. Keluar");
414 System.out.print("Masukkan pilihan: ");
415
416 int choice = scanner.nextInt();
417 scanner.nextLine();
418
419 switch (choice) {
420 case 1:
421 System.out.print("Masukkan Flight ID: ");
422 String fid = scanner.nextLine();
423 System.out.print("Masukkan Asal: ");
424 String asal = scanner.nextLine();
425 System.out.print("Masukkan Tujuan: ");
426 String tujuan = scanner.nextLine();
427 System.out.print("Masukkan Tanggal: ");
428 String tanggal = scanner.nextLine();
429 System.out.print("Masukkan Maskapai: ");
430 String maskapai = scanner.nextLine();
431 System.out.print("Masukkan Harga: ");
432 double harga = scanner.nextDouble();
433 scanner.nextLine();
434 sistem.tambahPenerbangan(fid, asal, tujuan, tanggal, maskapai, harga);
435 break;

```



```

435 break;
436 case 2:
437 sistem.lihatPenerbangan();
438 break;
439 case 3:
440 sistem.lihatPenerbangan();
441 System.out.print("Masukkan Reservasi ID: ");
442 String reservasiID = scanner.nextLine();
443 System.out.print("Masukkan Flight ID: ");
444 String flightID = scanner.nextLine();
445 System.out.print("Masukkan nama penumpang: ");
446 String namaPenumpang = scanner.nextLine();
447 System.out.print("Masukkan Harga: ");
448 double hargaReservasi = scanner.nextDouble();
449 scanner.nextLine();
450 sistem.buatReservasi(reservasiID, flightID, namaPenumpang, hargaReservasi);
451 break;
452 case 4:
453 sistem.lihatReservasi();
454 break;
455 case 5:
456 System.out.print("Masukkan Reservasi ID untuk verifikasi: ");
457 String verifikasiID = scanner.nextLine();
458 sistem.verifikasiPembayaran(verifikasiID);
459 break;
460 case 6:
461 System.out.print("Apakah Anda yakin ingin keluar? (y/n): ");
462 String konfirmasi = scanner.nextLine();
463 if (konfirmasi.equalsIgnoreCase("y")) {
464 System.out.println("Terimakasih Telah Menggunakan Program Ini....");
465 running = false;
466 }
467 break;
468 default:
469 System.out.println("Pilihan tidak valid, coba lagi.");
470 }
471 }
472 } while (running);
473
474 scanner.close();
475 }
476 }
477
478

```