

**Nama: Raihan Rahmadini Putri**

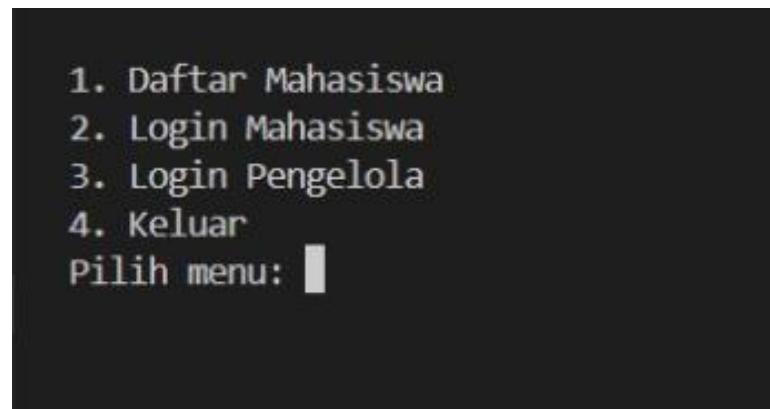
**Nim : 2309106008**

## **Posttest 5 PBO**

---

### **Screenshot Program**

- Menu awal



- Registras Mahasiswa



- Login Mahasiswa

```
1. Daftar Mahasiswa
2. Login Mahasiswa
3. Login Pengelola
4. Keluar
Pilih menu: 2
NIM: 2309106008
Password: 1234567890
```

- Tampilan Menu Mahasiswa

- Pilihan ke 1 Menu Mahasiswa

```
1. Buat Pesanan
2. Lihat Pesanan Saya
3. Logout
Pilih menu: 1
ID Pesanan: PSN-0bfb7de1
Pilih layanan:
1. Cuci Kering - Rp 10.000 - 2 Jam
2. Cuci Setrika - Rp 15.000 - 3 Jam
3. Cuci Kilat - Rp 20.000 - 1 Jam
Pilihan: 1
Pilih jenis barang:
1. Pakaian
2. Sprei
Pilihan: 1
? Pesanan berhasil dibuat!
? ID: PSN-0bfb7de1
? Layanan: Cuci Kering
? Estimasi Harga: Rp 10000
? Perkiraan Waktu: 2 Jam
```

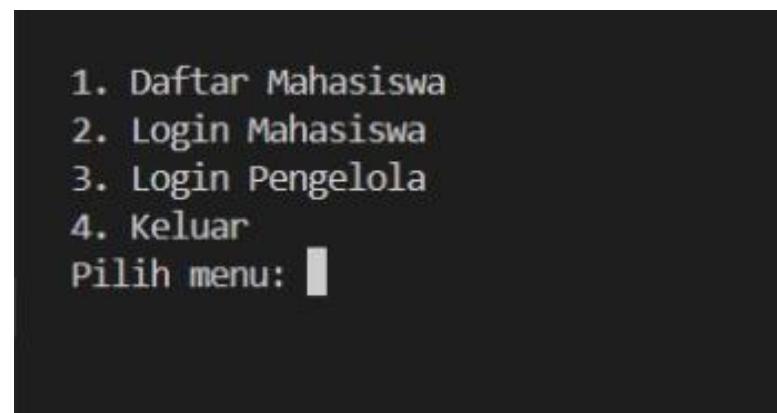
- Pilihan ke 2 Menu Mahasiswa

```
1. Buat Pesanan
2. Lihat Pesanan Saya
3. Logout
Pilih menu: 2
ID: PSN-0bfb7de1 | Layanan: Cuci Kering | Harga: Rp 15000 | Waktu: 2 Jam | Status: Menunggu
```

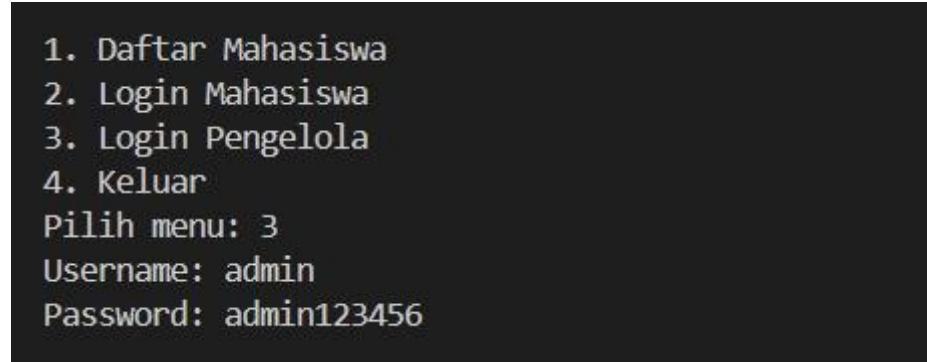
- Pilihan ke 2 jika pengelola menghapus pesanan Mahasiswa

```
1. Buat Pesanan
2. Lihat Pesanan Saya
3. Logout
Pilih menu: 2
?? Anda belum memiliki pesanan.
```

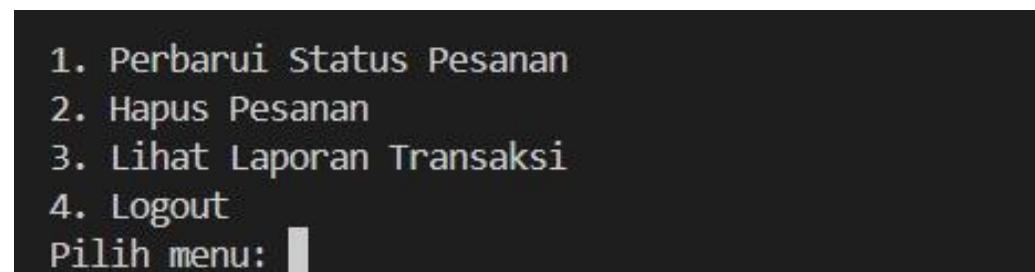
- Pilihan 3 jika logout akan Kembali ke menu awal



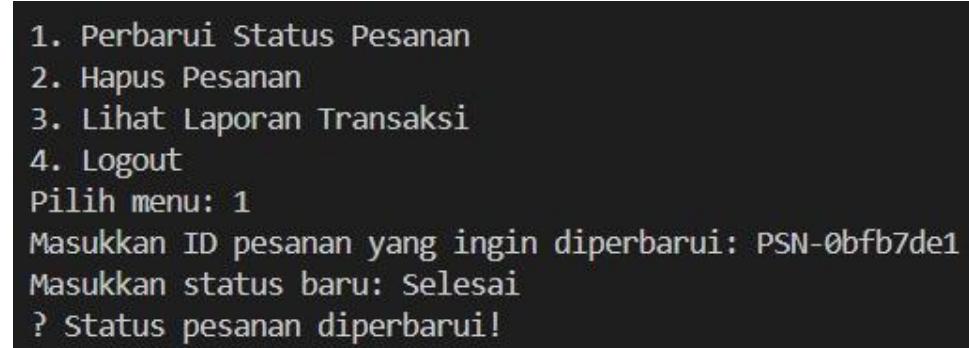
- Login Pengelola



- Tampilkan Menu Pengelola



- Pilihan ke 1 Menu Pengelola



- Pilihan ke 3 Menu Pengelola

```
1. Perbarui Status Pesanan
2. Hapus Pesanan
3. Lihat Laporan Transaksi
4. Logout
Pilih menu: 3
Total pesanan: 1
```

- Pilihan ke 2 Menu Pengelola

```
1. Perbarui Status Pesanan
2. Hapus Pesanan
3. Lihat Laporan Transaksi
4. Logout
Pilih menu: 2
Masukkan ID pesanan yang ingin dihapus: PSN-0fbfb7de1
Pesanan dihapus!
```

- Pilihan ke 4 kembali ke menu awal

```
1. Daftar Mahasiswa
2. Login Mahasiswa
3. Login Pengelola
4. Keluar
Pilih menu: ■
```

- Pilihan ke 4 keluar

```
1. Daftar Mahasiswa
2. Login Mahasiswa
3. Login Pengelola
4. Keluar
Pilih menu: 4
PS C:\Users\raiha\Downloads\Posttest3_PBO\src> ■
```

## Code Program

- **Package Model**

- **Class Layanan (induk)**

```
src > model > J Layanan.java > Java > Layanan > getNamaLayanan()
1 package model;
2
3 public abstract class Layanan {
4
5     public final String namalayanan;
6     public final int estimasiHarga;
7     public final String perkiraanWaktu;
8
9     public Layanan(String namalayanan, int estimasiHarga, String perkiraanWaktu) {
10        this.namalayanan = namalayanan;
11        this.estimasiHarga = estimasiHarga;
12        this.perkiraanWaktu = perkiraanWaktu;
13    }
14
15    public abstract int hitungHarga(String jenisBarang);
16
17    public final String getNamaLayanan() {
18        return namalayanan;
19    }
20
21    public String getPerkiraanWaktu() {
22        return perkiraanWaktu;
23    }
24
25    public int getEstimasiHarga() {
26        return estimasiHarga;
27    }
28}
```

- **Subclass LayananCuciKering**

```
src > model > J LayananCuciKering.java > Java > LayananCuciKering
1 package model;
2
3 public class LayananCuciKering extends Layanan {
4     public LayananCuciKering() {
5         super(namalayanan:"Cuci Kering", estimasiHarga:10000, perkiraanWaktu:"2 Jam");
6     }
7
8     @Override
9     public int hitungHarga(String jenisBarang) {
10        if ("pakaian".equals(jenisBarang)) {
11            return estimasiHarga;
12        } else if ("sprei".equals(jenisBarang)) {
13            return estimasiHarga + 5000; // Harga lebih mahal untuk sprei
14        }
15        return 0;
16    }
17}
```

- **Subclass LayananCuciKilat**

```
src > model > J LayananCuciKilat.java > Language Support for Java(TM) by Red Hat > LayananCuciKilat
1 package model;
2
3 public class LayananCuciKilat extends Layanan {
4     public LayananCuciKilat() {
5         super(namalayanan:"Cuci Kilat", estimasiHarga:20000, perkiraanWaktu:"1 Jam");
6     }
7
8     @Override
9     public int hitungHarga(String jenisBarang) {
10        if ("pakaian".equals(jenisBarang)) {
11            return estimasiHarga;
12        } else if ("sprei".equals(jenisBarang)) {
13            return estimasiHarga + 10000; // Harga lebih mahal untuk sprei
14        }
15        return 0;
16    }
17}
```

- **Subclass LayananCuciSetrika**

```
src > model > LayananCuciSetrika.java > Language Support for Java(TM) by Red Hat > LayananCuciSetrika
1 package model;
2
3 public final class LayananCuciSetrika extends Layanan {
4     public LayananCuciSetrika() {
5         super("NamaLayanan":"Cuci Setrika", estimasiHarga:15000, perkiraanWaktu:"3 Jam");
6     }
7
8     @Override
9     public int hitungHarga(String jenisBarang) {
10        if ("pakalan".equals(jenisBarang)) {
11            return estimasiHarga;
12        } else if ("sprei".equals(jenisBarang)) {
13            return estimasiHarga + 7000; // Harga lebih mahal untuk sprei
14        }
15        return 0;
16    }
17 }
```

## ■ Class Mahasiswa

```
src > model > Mahasiswa.java > Language Support for Java(TM) by Red Hat > Mahasiswa > Mahasiswa(String, String, String, String, String)
1 package model;
2
3 public class Mahasiswa {
4     private final String nim;
5     private String nama;
6     private String email;
7     private String noTelepon;
8     private String password;
9
10    // Constructor
11    public Mahasiswa(String nim, String nama, String email, String noTelepon, String password) {
12        this.nim = nim;
13        this.nama = nama;
14        this.email = email;
15        this.noTelepon = noTelepon;
16        this.password = password;
17    }
18
19    // Getter
20    public String getNim() {
21        return nim;
22    }
23
24    public String getNama() {
25        return nama;
26    }
27
28    public String getEmail() {
29        return email;
30    }
31
32    public String getNoTelepon() {
33        return noTelepon;
34    }
35 }
```

```
36    public String getPassword() {
37        return password;
38    }
39
40    public void setNama(String nama) {
41        if (!nama.trim().isEmpty()) {
42            this.nama = nama;
43        } else {
44            System.out.println("▲ Nama tidak boleh kosong!");
45        }
46    }
47
48    public void setEmail(String email) {
49        if (!email.contains("@")) {
50            this.email = email;
51        } else {
52            System.out.println("▲ Email tidak valid!");
53        }
54    }
55
56    public void setNoTelepon(String noTelepon) {
57        if (!noTelepon.matches("\\d{10,13}")) {
58            this.noTelepon = noTelepon;
59        } else {
60            System.out.println("▲ No telepon tidak valid!");
61        }
62    }
63 }
```

```

54     public void setPassword(String password) {
55         if (password.length() >= 6) {
56             this.password = password;
57         } else {
58             System.out.println("⚠️ Password harus minimal 6 karakter!");
59         }
60     }
61
62     public void tampilkan() {
63         System.out.println("NIM: " + nim + ", Nama: " + nama + ", Email: " + email + ", No Telepon: " + noTelepon);
64     }
65 }
66

```

## ▪ Class Pengelola

```

src > model > Pengelolajava > Language Support for Java(TM) by Red Hat > Pengelola > getPassword0
1 package model;
2
3 public class Pengelola {
4     private final String username;
5     private String password;
6
7     public Pengelola(String username, String password) {
8         this.username = "admin";
9         this.password = "admin123456";
10    }
11
12    public String getUsername() {
13        return username;
14    }
15
16    public String getPassword() {
17        return password;
18    }
19
20    public void setPassword(String password) {
21        if (password.length() >= 6) {
22            this.password = password;
23        } else {
24            System.out.println("⚠️ Password harus minimal 6 karakter!");
25        }
26    }
27
28    protected void resetPassword() {
29        this.password = "default123";
30    }
31 }
32

```

## ▪ Class Pesanan

```

src > model > Pesanan > Language Support for Java(TM) by Red Hat > Pesanan > getLayanan0
1 package model;
2
3 public class Pesanan {
4     private final String idPesanan;
5     private final String nimMahasiswa;
6     private final Layanan layanan; // Tipe Layanan bisa berubah sesuai jenis layanan yang dipilih
7     private String status;
8     private final String jenisBarang;
9
10    public Pesanan(String idPesanan, String nimMahasiswa, Layanan layanan, String status, String jenisBarang) {
11        this.idPesanan = idPesanan;
12        this.nimMahasiswa = nimMahasiswa;
13        this.layanan = layanan;
14        this.status = status;
15        this.jenisBarang = jenisBarang;
16    }
17
18    public String getIdPesanan() {
19        return idPesanan;
20    }
21
22    public String getNimMahasiswa() {
23        return nimMahasiswa;
24    }
25
26    public Layanan getLayanan() {
27        return layanan;
28    }
29
30    public String getStatus() {
31        return status;
32    }
33
34    public void setStatus(String status) {
35        this.status = status;
36    }
37

```

```
37
38     public String getJenisBarang() {
39         return jenisBarang;
40     }
41
42     public int getEstimasiHarga() {
43         return layanan.hitungHarga(jenisBarang);
44     }
45
46     public String getPerkiraanWaktu() {
47         return layanan.getPerkiraanWaktu();
48     }
49 }
50 }
```

- **Package Service**

- **Class LaundryService**

```
1 package service;
2
3 import java.util.ArrayList;
4 import java.util.List;
5 import java.util.Scanner;
6 import java.util.UUID;
7 import model.Layanan;
8 import model.LayananCuciKering;
9 import model.LayananCuciKilat;
10 import model.LayananCuciSetrika;
11 import model.Mahasiswa;
12 import model.Pengelola;
13 import model.Pesanan;
14
15
16 public class LaundryService {
17     private final List<Mahasiswa> mahasiswaList = new ArrayList<>();
18     private final List<Pengelola> pengelolaList = new ArrayList<>();
19     private final List<Pesanan> pesananList = new ArrayList<>();
20     private final Scanner scanner = new Scanner(System.in);
21     private Mahasiswa loggedInMahasiswa = null;
22     private Pengelola loggedInPengelola = null;
23
24     public void mulai() {
25         pengelolaList.add(new Pengelola("admin", "admin123"));
26
27         while (true) {
28             System.out.println("\n1. Daftar Mahasiswa\n2. Login
29 Mahasiswa\n3. Login Pengelola\n4. Keluar");
29 Mahasiswa\n3. Login Pengelola\n4. Keluar");
30             System.out.print("Pilih menu: ");
31             int pilihan = scanner.nextInt();
32             scanner.nextLine();
33
34             switch (pilihan) {
35                 case 1 -> daftarMahasiswa();
36                 case 2 -> loginMahasiswa();
37                 case 3 -> loginPengelola();
38                 case 4 -> { return; }
39                 default -> System.out.println("Pilihan tidak valid!");
40             }
41         }
42     }
}
```

```
43
44 private void daftarMahasiswa() {
45     System.out.print("NIM: ");
46     String nim = scanner.nextLine();
47     System.out.print("Nama: ");
48     String nama = scanner.nextLine();
49     System.out.print("Email: ");
50     String email = scanner.nextLine();
51     System.out.print("No Telepon: ");
52     String noTelepon = scanner.nextLine();
53     System.out.print("Password: ");
54     String password = scanner.nextLine();
55
56     mahasiswaList.add(new Mahasiswa(nim, nama, email, noTelepon,
57 password));
57
58     System.out.println("Mahasiswa berhasil terdaftar!");
59 }
60
61 private void loginMahasiswa() {
62     System.out.print("NIM: ");
63     String nim = scanner.nextLine();
64     System.out.print("Password: ");
65     String password = scanner.nextLine();
66
67     for (Mahasiswa m : mahasiswaList) {
68         if (m.getNim().equals(nim) &&
69 m.getPassword().equals(password)) {
70             loggedInMahasiswa = m;
71             menuMahasiswa();
72             return;
73         }
74     }
75     System.out.println("Login gagal! Periksa NIM dan password.");
76 }
77
78 private void loginPengelola() {
79     System.out.print("Username: ");
80     String username = scanner.nextLine();
81     System.out.print("Password: ");
82     String password = scanner.nextLine();
83
84     for (Pengelola p : pengelolaList) {
85         if (p.getUsername().equals(username) &&
86 p.getPassword().equals(password)) {
87             loggedInPengelola = p;
88             menuPengelola();
89             return;
90         }
91     }
92     System.out.println("Login gagal! Periksa username dan
93 password.");
93
94 }
95
96 private void menuMahasiswa() {
97     while (true) {
```

```

99             System.out.println("\n1. Buat Pesanan\n2. Lihat Pesanan
100 Saya\n3. Logout");
101         System.out.print("Pilih menu: ");
102
103     int pilihan;
104     try {
105         pilihan = Integer.parseInt(scanner.nextLine());
106     } catch (NumberFormatException e) {
107         System.out.println("⚠ Input harus berupa angka!
108 Silakan coba lagi.");
109         continue;
110     }
111
112     switch (pilihan) {
113         case 1 -> buatPesanan();
114         case 2 -> lihatPesananMahasiswa();
115         case 3 -> {
116             loggedInMahasiswa = null;
117             return;
118         }
119         default -> System.out.println("⚠ Pilihan tidak valid!
120 Silakan pilih 1-3.");
121     }
122 }
123 }
124
125
126 private void menuPengelola() {
127     if (loggedInPengelola == null) {
128         System.out.println("Anda belum login sebagai pengelola!");
129         return;
130     }
131
132     while (true) {
133         System.out.println("\n1. Perbarui Status Pesanan\n2. Hapus
134 Pesanan\n3. Lihat Laporan Transaksi\n4. Logout");
135         System.out.print("Pilih menu: ");
136
137         int pilihan;
138         try {
139             pilihan = Integer.parseInt(scanner.nextLine());
140         } catch (NumberFormatException e) {
141             System.out.println("⚠ Input harus berupa angka!
142 Silakan coba lagi.");
143             continue;
144         }
145
146         switch (pilihan) {
147             case 1 -> perbaruiStatusPesanan();
148             case 2 -> hapusPesanan();
149             case 3 -> lihatLaporanTransaksi();
150             case 4 -> {
151                 System.out.println("Logout berhasil, " +
152 loggedInPengelola.getUsername());
153                 loggedInPengelola = null;
154             return;

```

```

155         }
156         default -> System.out.println("⚠️ Pilihan tidak valid!
157 Silakan pilih 1-4.");
158     }
159 }
160 }
161
162 private void buatPesanan() {
163     String idPesanan = "PSN-" +
164 UUID.randomUUID().toString().substring(0, 8);
165     System.out.println("ID Pesanan: " + idPesanan);
166
167     int pilihan;
168     while (true) {
169         System.out.println("Pilih layanan:");
170         System.out.println("1. Cuci Kering - Rp 10.000 - 2 Jam");
171         System.out.println("2. Cuci Setrika - Rp 15.000 - 3 Jam");
172         System.out.println("3. Cuci Kilat - Rp 20.000 - 1 Jam");
173         System.out.print("Pilihan: ");
174
175         try {
176             pilihan = Integer.parseInt(scanner.nextLine());
177             if (pilihan >= 1 && pilihan <= 3) break;
178             System.out.println("⚠️ Pilihan tidak valid! Masukkan
179 angka 1-3.");
180         } catch (NumberFormatException e) {
181             System.out.println("⚠️ Input harus berupa angka!
182 Silakan coba lagi.");
183         }
184     }
185
186     // Deklarasikan layanan
187     Layanan layanan;
188
189     // Inisialisasi sesuai pilihan
190     switch (pilihan) {
191         case 1 -> layanan = new LayananCuciKering();
192         case 2 -> layanan = new LayananCuciSetrika();
193         case 3 -> layanan = new LayananCuciKilat();
194         default -> {
195             System.out.println("✖️ Pilihan layanan tidak valid.");
196             return;
197         }
198     }
199
200     // Pilih jenis barang (pakaian atau sprei)
201     String jenisBarang;
202     while (true) {
203         System.out.println("Pilih jenis barang:");
204         System.out.println("1. Pakaian");
205         System.out.println("2. Sprei");
206         System.out.print("Pilihan: ");
207
208         try {
209             int jenisPilihan =
210             Integer.parseInt(scanner.nextLine());

```

```

211
212         switch (jenisPilihan) {
213             case 1:
214                 jenisBarang = "pakaian";
215                 break;
216             case 2:
217                 jenisBarang = "sprei";
218                 break;
219             default:
220                 System.out.println("⚠️ Pilihan tidak valid!");
221             Masukkan angka 1-2.");
222             continue;
223         }
224         break;
225     } catch (NumberFormatException e) {
226         System.out.println("⚠️ Input harus berupa angka!");
227     Silakan coba lagi.);
228     }
229 }
230
231     // Buat dan simpan pesanan
232     Pesanan pesanan = new Pesanan(idPesanan,
233 loggedInMahasiswa.getNim(), layanan, "Menunggu", jenisBarang);
234     pesananList.add(pesanan);
235
236     // Tampilkan detail pesanan
237     System.out.println("☑️ Pesanan berhasil dibuat!");
238     System.out.println("◊ ID: " + pesanan.getIdPesanan());
239     System.out.println("◊ Layanan: " + layanan.getNamaLayanan());
240     System.out.println("⌚ Estimasi Harga: Rp " +
241 pesanan.getEstimasiHarga());
242     System.out.println("⌚ Perkiraan Waktu: " +
243 layanan.getPerkiraanWaktu());
244 }
245
246
247
248     private void lihatPesananMahasiswa() {
249         if (loggedInMahasiswa == null) {
250             System.out.println("⚠️ Anda belum login.");
251             return;
252         }
253
254         boolean adaPesanan = false;
255         for (Pesanan p : pesananList) {
256             if
257 (p.getNimMahasiswa().equals(loggedInMahasiswa.getNim())) {
258                 Layanan layanan = p.getLayanan();
259                 System.out.println("ID: " + p.getIdPesanan()
260                     + " | Layanan: " + layanan.getNamaLayanan()
261                     + " | Harga: Rp " + layanan.getEstimasiHarga()
262                     + " | Waktu: " + layanan.getPerkiraanWaktu()
263                     + " | Status: " + p.getStatus());
264                 adaPesanan = true;
265             }
266         }

```

```

267
268     if (!adaPesanan) {
269         System.out.println("⚠️ Anda belum memiliki pesanan.");
270     }
271 }
272
273
274
275 private void perbaruiStatusPesanan() {
276     while (true) {
277         System.out.print("Masukkan ID pesanan yang ingin
278 diperbarui: ");
279         String id = scanner.nextLine();
280
281         for (Pesanan p : pesananList) {
282             if (p.getIdPesanan().equals(id)) {
283                 System.out.print("Masukkan status baru: ");
284                 p.setStatus(scanner.nextLine());
285                 System.out.println("☑ Status pesanan
diperbarui!");
286             }
287         }
288
289         System.out.println("⚠️ ID tidak ditemukan! Coba lagi.");
290     }
291 }
292
293
294
295 private void hapusPesanan() {
296     System.out.print("Masukkan ID pesanan yang ingin dihapus: ");
297     String id = scanner.nextLine();
298     pesananList.removeIf(p -> p.getIdPesanan().equals(id));
299     System.out.println("Pesanan dihapus!");
300 }
301
302
303 private void lihatLaporanTransaksi() {
304     System.out.println("Total pesanan: " + pesananList.size());
305 }
306 }
```

- Main Class

```

src > J App.java > Language Support for Java(TM) by Red Hat > ⚙️ App
1 import service.LaundryService;
2
3 public class App {
4     Run | Debug | Run main | Debug main
5     public static void main(String[] args) throws Exception {
6         LaundryService app = new LaundryService();
7         app.mulai();
8     }
9 }
```