PEMPROGRAMAN BERBASIS OBJEK

POSTTEST 4

Informatika A 1'23
Davina Putri Ananta
2309106002

PROGRAM STUDI INFORMATIKA

UNIVERSITAS MULAWARMAN

SAMARINDA

2025

SS PROGRAM (hanya ss program yang diganti atau ditambahkan aja sesuai dengan aturan penerapan yang diperintahkan pada posttest 4):

Admin.java :

```java
package kelas;

public class Admin extends Pelanggan {
    public Admin(String idAdmin, String nama, String email, String password) {
        super(idAdmin, nama, email, password, alamat:"-", noTelp:"-");
    }

    public void kelolaSistem() {
        System.out.println("Admin " + getNama() + " sedang mengelola sistem...");
    }
    @Override
    public void displayInfo() {
        System.out.println("Admin ID: " + getIdPelanggan() +
                           ", Nama: " + getNama() + " (Role: Administrator)");
    }
}
```

CustomParfum.java:

```java
package kelas;

public class CustomParfum extends Pelanggan {
    public String idCustom;
    public String kelasParfum;
    public String notesParfum;
    public double hargaCustom;

    public CustomParfum(String idPelanggan, String idCustom, String kelasParfum,
                        String notesParfum, double hargaCustom) {
        super(idPelanggan, nama:"Pelanggan Custom", email:"-", password:"-", alamat:"-", noTelp:"-");
        this.idCustom = idCustom;
        this.kelasParfum = kelasParfum;
        this.notesParfum = notesParfum;
        this.hargaCustom = hargaCustom;
    }
    @Override
    public void displayInfo() {
        System.out.println("Pelanggan Custom - ID: " + getIdPelanggan() +
                           ", Custom ID: " + idCustom + ", Kelas: " + kelasParfum);
    }
    public String getkelasparfum() {

        throw new UnsupportedOperationException(message:"Unimplemented method 'getkelasparfum'");
    }
}
```

Pelanggan.java:

```java
package kelas;

public class Pelanggan {
    private String idPelanggan;
    private String nama;
    private String email;
    private String password;
    private String alamat;
    private String noTelp;

    public Pelanggan(String idPelanggan, String nama, String email, String password, String alamat, String noTelp) {
        this.idPelanggan = idPelanggan;
        this.nama = nama;
        this.email = email;
        this.password = password;
        this.alamat = alamat;
        this.noTelp = noTelp;
    }

    public String getIdPelanggan() {
        return idPelanggan;
    }

    public void setIdPelanggan(String idPelanggan) {
        this.idPelanggan = idPelanggan;
    }

    public String getNama() {
        return nama;
    }

    public void setNama(String nama) {
        this.nama = nama;
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }

    public String getAlamat() {
        return alamat;
    }

    public void setAlamat(String alamat) {
        this.alamat = alamat;
    }

    public String getNoTelp() {
        return noTelp;
    }

    public void setNoTelp(String noTelp) {
        this.noTelp = noTelp;
    }
        public void displayInfo() {
    System.out.println("ID: " + idPelanggan + ", Nama: " + nama + ", Email: " + email);
    }
}
```

ScentiqueSystem.java:

```java
import java.util.*;
import kelas.*;

public class ScentiqueSystem {
    static Scanner scanner = new Scanner(System.in);
    static ArrayList<Parfum> daftarParfum = new ArrayList<>();
    static ArrayList<Pelanggan> daftarPelanggan = new ArrayList<>();

    public static void tampilkanInfoPelanggan(Pelanggan pelanggan) {
        pelanggan.displayInfo();
    }

    public static void tampilkanInfoPelanggan(Pelanggan pelanggan, boolean showDetail) {
        if (showDetail) {
            System.out.println(x:"\n=== DETAIL PELANGGAN ===");
            pelanggan.displayInfo();
            if (pelanggan instanceof Admin) {       instanceof pattern can be used here
                ((Admin)pelanggan).kelolaSistem();
            } else if (pelanggan instanceof CustomParfum) {       instanceof pattern can be used here
                System.out.println("Kelas Parfum: " + ((CustomParfum)pelanggan).getkelasparfum()); // Method khusus CustomParfum
            }
        } else {
            pelanggan.displayInfo();
        }
    }

    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        daftarPelanggan.add(new Admin(idAdmin:"ADM001", nama:"Admin Scentique", email:"admin@scentique.com", password:"admin123"));
        daftarPelanggan.add(new CustomParfum(idPelanggan:"CUST001", idCustom:"ORDER123", kelasParfum:"Premium", notesParfu…"Aroma Kayu", 500000));

        int pilihan;
        do {
            System.out.println(x:"\nSistem Pemesanan dan Custom Parfum di Scentique");
            System.out.println(x:"1. Tambah Parfum");
            System.out.println(x:"2. Lihat Parfum");
            System.out.println(x:"3. Hapus Parfum");
            System.out.println(x:"4. Lihat Contoh Polymorphism"); // Menu baru
            System.out.println(x:"5. Keluar");
            System.out.print(s:"Pilih menu: ");

            while (!scanner.hasNextInt()) {
                System.out.println(x:"Input harus berupa angka!");
                scanner.next();
            }
```

```java
            }
            pilihan = scanner.nextInt();
            scanner.nextLine();

            switch (pilihan) {       Convert switch to rule switch
                case 1:
                    tambahParfum();
                    break;
                case 2:
                    lihatParfum();
                    break;
                case 3:
                    hapusParfum();
                    break;
                case 4:
                    System.out.println(x:"\nDemo Polymorphism:");
                    for (Pelanggan p : daftarPelanggan) {
                        tampilkanInfoPelanggan(p, showDetail:true);
                    }
                    break;
                case 5:
                    System.out.println(x:"Terima kasih telah menggunakan sistem Scentique!");
                    break;
                default:
                    System.out.println(x:"Pilihan tidak valid.");
            }
        } while (pilihan != 5);
    }

    static void tambahParfum() {
        System.out.print(s:"Masukkan ID Parfum: ");
        String idParfum = scanner.nextLine().trim();
        if (idParfum.isEmpty()) {
            System.out.println(x:"ID Parfum tidak boleh kosong!");
            return;
        }

        System.out.print(s:"Masukkan Nama Parfum: ");
        String namaParfum = scanner.nextLine().trim();
        if (namaParfum.isEmpty()) {
            System.out.println(x:"Nama Parfum tidak boleh kosong!");
            return;
```

```java
 85                 return;
 86             }
 87
 88         System.out.print(s:"Masukkan Harga Parfum: ");
 89         while (!scanner.hasNextDouble()) {
 90             System.out.println(x:"Input harus berupa angka!");
 91             scanner.next();
 92         }
 93         double hargaParfum = scanner.nextDouble();
 94
 95         System.out.print(s:"Masukkan Stok Parfum: ");
 96         while (!scanner.hasNextInt()) {
 97             System.out.println(x:"Input harus berupa angka!");
 98             scanner.next();
 99         }
100         int stok = scanner.nextInt();
101         scanner.nextLine();
102
103         daftarParfum.add(new Parfum(idParfum, namaParfum, hargaParfum, stok));
104         System.out.println(x:"Parfum berhasil ditambahkan!");
105     }
106
107     static void lihatParfum() {
108         if (daftarParfum.isEmpty()) {
109             System.out.println(x:"Tidak ada parfum dalam daftar.");
110         } else {
111             System.out.println(x:"Daftar Parfum:");
112             for (Parfum p : daftarParfum) {
113                 System.out.println("ID: " + p.getIdParfum() + ", Nama: " + p.getNamaParfum() +
114                                     ", Harga: " + p.getHargaParfum() + ", Stok: " + p.getStok());
115             }
116         }
117     }
118
119     static void hapusParfum() {
120         if (daftarParfum.isEmpty()) {
121             System.out.println(x:"Tidak ada parfum dalam daftar.");
122             return;
123         }
124
125         System.out.print(s:"Masukkan ID Parfum yang ingin dihapus: ");
126         String idParfum = scanner.nextLine().trim();
127         boolean ditemukan = false;
128
129         for (int i = 0; i < daftarParfum.size(); i++) {
130             if (daftarParfum.get(i).getIdParfum().equalsIgnoreCase(idParfum)) {
131                 daftarParfum.remove(i);
132                 System.out.println(x:"Parfum berhasil dihapus.");
133                 ditemukan = true;
134                 break;
135             }
136         }
137
138         if (!ditemukan) {
139             System.out.println(x:"Parfum tidak ditemukan.");
140         }
141     }
142 }
```