



```
package model;

public class User {
    protected String email;
    protected String username;
    protected String password;

    public User(String email, String username, String password) {
        this.email = email;
        this.username = username;
        this.password = password;
    }

    public String getUsername() {
        return username;
    }
    public String getPassword() {
        return password;
    }
}
```

Override pada class Admin

```
package model;

public class Admin extends User{
    public Admin(String email, String username, String password) {
        super(email, username, password);
    }

    @Override
    public String getUsername() {
        return super.getUsername();
    }

    @Override
    public String getPassword() {
        return super.getPassword();
    }

    @Override
    public String toString() {
        return "Admin: " + username + ", email= " + email;
    }
}
```

codesnap.dev

Override pada class Pengguna

```
package model;

public class Pengguna extends User{
    public static final Pengguna DEFAULT_USER = new Pengguna("user@mail.com", "user", "user123");

    public Pengguna (String email, String username, String password){
        super(email, username, password);
    }

    @Override
    public String toString() {
        return "Pengguna: " + username + ", email: " + email;
    }
}
```

```

package model;

public class Aksesori{
    private final int id;
    public String nama;
    public String merek;
    public int harga;
    public Kategori kategori;

    public Aksesori(int id, String nama, String merek, int harga, Kategori kategori){
        this.id = id;
        this.nama = nama;
        this.merek = merek;
        this.harga = harga;
        this.kategori = kategori;
    }

    public int get_id(){
        return id;
    }
    public String get_nama(){
        return nama;
    }
    public String get_merek(){
        return merek;
    }
    public int get_harga(){
        return harga;
    }
    public Kategori get_kategori(){
        return kategori;
    }

    public void set_nama(String nama){
        this.nama = nama;
    }
    public void set_merek(String merek){
        this.merek = merek;
    }
    public void set_harga(int harga){
        this.harga = harga;
    }
    public void set_kategori(Kategori kategori){
        this.kategori = kategori;
    }

    public String dalam_string(){
        return "ID: " + id + ", Nama: " + nama + ", Merek: " + merek + ", Harga: Rp" + harga + ", Kategori: " + (kategori == null ? kategori.getNamaKategori() : "Belum ada kategori");
    }
}

```

```
package model;

public class Kategori{
    private final int idKategori;
    public String nama_kategori;

    public Kategori(int idKategori, String nama_kategori){
        this.idKategori = idKategori;
        this.nama_kategori = nama_kategori;
    }

    public int getIdKategori(){
        return idKategori;
    }

    public String getNamaKategori(){
        return nama_kategori;
    }

    public void set_nama_kategori(String nama_kategori){
        this.nama_kategori = nama_kategori;
    }

    public String dalam_string(){
        return "ID Kategori: " + idKategori + ", Nama Kategori: " + nama_kategori;
    }
}
```

```

import java.util.ArrayList;
import java.util.Scanner;
import model.Admin;
import model.Pengguna;
import service.CRUD;

public class Main {
    public static final ArrayList<Pengguna> dataPengguna = new ArrayList<>();
    public static final ArrayList<Admin> dataAdmin = new ArrayList<>();

    private static final Scanner scanner = new Scanner(System.in);

    public static void main(String[] args) {
        Admin admin = new Admin("Admin@mail.com", "admin", "admin");

        dataPengguna.add(Pengguna.DEFAULT_USER);

        while (true) {
            System.out.println("\n=== Menu Utama ===");
            System.out.println("1. Login");
            System.out.println("2. Register");
            System.out.println("3. Exit");
            System.out.print("Pilihan: ");
            int pilihan = scanner.nextInt();
            scanner.nextLine();

            switch (pilihan) {
                case 1 → login();
                case 2 → register();
                case 3 → {
                    System.out.println("Terima kasih telah menggunakan program ini.");
                    return;
                }
                default → System.out.println("Pilihan tidak valid.");
            }
        }

        private static void login() {
            System.out.println("\n=== Login ===");
            System.out.print("Username: ");
            String username = scanner.nextLine();
            System.out.print("Password: ");
            String password = scanner.nextLine();

            for (Admin admin : dataAdmin) {
                if (username.equals(admin.getUsername()) && password.equals(admin.getPassword())) {
                    System.out.println("\nLogin sebagai Admin berhasil!");
                    menuAdmin();
                    return;
                }
            }

            for (Pengguna pengguna : dataPengguna) {
                if (username.equals(pengguna.getUsername()) && password.equals(pengguna.getPassword())) {
                    System.out.println("\nLogin sebagai User berhasil!");
                    menuUser();
                    return;
                }
            }

            System.out.println("Username atau password salah.");
        }
    }
}

```

Overloading pada method register

```
private static void register() {
    System.out.println("\n=== Register ===");
    System.out.print("Email: ");
    String email = scanner.nextLine();
    System.out.print("Username: ");
    String username = scanner.nextLine();
    System.out.print("Password: ");
    String password = scanner.nextLine();
    System.out.print("Masukkan Kode Registrasi Admin (Isi 0 untuk abaikan): ");
    int kodeAdmin = scanner.nextInt();
    scanner.nextLine();

    register(email, username, password, kodeAdmin);
}

private static void register(String email, String username, String password, int kodeAdmin) {
    if (kodeAdmin == 321123) {
        Admin admin = new Admin(email, username, password);
        dataAdmin.add(admin);
        System.out.println("Registrasi Admin berhasil!");
    } else if (kodeAdmin == 0) {
        Pengguna pengguna = new Pengguna(email, username, password);
        dataPengguna.add(pengguna);
        System.out.println("Registrasi Pengguna berhasil!");
    } else {
        System.out.println("Kode rahasia tidak valid. Registrasi gagal.");
    }
}

private static void menuAdmin() {
    int pilihan;
    do {
        System.out.println("\n=== Menu Admin ===");
        System.out.println("1. Tambah Aksesori");
        System.out.println("2. Lihat Aksesori");
        System.out.println("3. Ubah Aksesori");
        System.out.println("4. Hapus Aksesori");
        System.out.println("5. Tambah Kategori");
        System.out.println("6. Lihat Kategori");
        System.out.println("7. Daftar Pengguna");
        System.out.println("8. Keluar");
        System.out.print("Pilihan: ");
        pilihan = scanner.nextInt();
        scanner.nextLine();

        switch (pilihan) {
            case 1 → CRUD.tambah_aksesori();
            case 2 → CRUD.lihat_aksesori();
            case 3 → CRUD.ubah_aksesori();
            case 4 → CRUD.hapus_aksesori();
            case 5 → CRUD.tambah_kategori();
            case 6 → CRUD.lihat_kategori();
            case 7 → {
                System.out.println("\n=== Daftar Pengguna ===");
                if (dataPengguna.isEmpty()) {
                    System.out.println("Tidak ada pengguna yang terdaftar.");
                } else {
                    for (Pengguna pengguna : dataPengguna) {
                        System.out.println(pengguna.toString());
                    }
                }
            }
            case 8 → {
                System.out.println("Keluar dari menu admin.");
                return;
            }

            default → System.out.println("Pilihan tidak valid.");
        }
    } while (pilihan < 0);
}

private static void menuUser() {
    System.out.println("\n=== Menu User ===");
    System.out.println("1. Lihat Aksesori");
    System.out.println("2. Keluar");
    System.out.print("Pilihan: ");
    int pilihan = scanner.nextInt();
    scanner.nextLine();

    if (pilihan == 1) {
        CRUD.lihat_aksesori();
    } else if (pilihan == 2) {
        System.out.println("Keluar dari menu user.");
    } else {
        System.out.println("Pilihan tidak valid.");
    }
}
}
```