Tugas Lab Modul 2

CODELAB 1

// Kelas Hewan dengan atribut nama, jenis, dan suara

class Hewan {

    String nama;

    String jenis;

    String suara;

    // Konstruktor untuk inisialisasi atribut, this=memanggil att kls

    public Hewan(String nama, String jenis, String suara) {

        this.nama = nama;

        this.jenis = jenis;

        this.suara = suara;

    }

    // Metode untuk menampilkan informasi hewan

    public void tampilkanInfo() {

        System.out.println("Nama  : " + this.nama);

        System.out.println("Jenis : " + this.jenis);

        System.out.println("Suara : " + this.suara);

    }

}

// Kelas Main dengan metode main untuk menjalankan program

public class Main {

    public static void main(String[] args) {

        // Membuat objek Hewan pertama: hewan1

        Hewan hewan1 = new Hewan("Kucing", "Mamalia", "Nyann~~");

        // Membuat objek Hewan kedua: hewan2

        Hewan hewan2 = new Hewan("Anjing", "Mamalia", "Woof-Woof!!");

        // Memanggil metode tampilkanInfo() untuk masing-masing objek

        System.out.println("Informasi Hewan 1:");

        hewan1.tampilkanInfo();

        System.out.println("\nInformasi Hewan 2:");

        hewan2.tampilkanInfo();

    }

}

CODELAB 2

// Kelas RekeningBank: Mengelola data rekening nasabah

class RekeningBank {

    // Atribut-atribut untuk menyimpan informasi rekening

    String nomorRekening;

    String namaPemilik;

    double saldo;

    // Konstruktor untuk menginisialisasi atribut

    public RekeningBank(String nomorRekening, String namaPemilik, double saldo) {

        this.nomorRekening = nomorRekening;

        this.namaPemilik = namaPemilik;

        this.saldo = saldo;

    }

    // Method untuk menampilkan informasi rekening

    public void tampilkanInfo() {

        System.out.println("Nomor Rekening : " + this.nomorRekening);

        System.out.println("Nama Pemilik   : " + this.namaPemilik);

        System.out.println("Saldo          : " + this.saldo);

    }

    // Method untuk menyetor uang ke dalam rekening

    public void setorUang(double jumlah) {

        this.saldo += jumlah;

        System.out.println("Transaksi Setor Uang");

        System.out.println("Jumlah Setoran : " + jumlah);

        System.out.println("Saldo Baru     : " + this.saldo);

    }

    // Method untuk menarik uang dari rekening jika saldo mencukupi

    public void tarikUang(double jumlah) {

        if(this.saldo >= jumlah) {

            this.saldo -= jumlah;

            System.out.println("Transaksi Tarik Uang");

            System.out.println("Jumlah Penarikan : " + jumlah);

            System.out.println("Saldo Baru       : " + this.saldo);

        } else {

            System.out.println("Saldo tidak mencukupi untuk melakukan penarikan sebesar " + jumlah);

        }

    }

}

// Kelas Main: Titik masuk eksekusi program

public class Main2 {

    public static void main(String[] args) {

        // Membuat objek rekening1 dengan data milik "kalian"

        RekeningBank rekening1 = new RekeningBank("202110370311158", "yusuf", 1000000);

        // Membuat objek rekening2 dengan data milik "teman kalian"

        RekeningBank rekening2 = new RekeningBank("202110370311437", "alif", 500000);

        // Menampilkan informasi awal kedua rekening

        System.out.println("=== Informasi Rekening 1 ===");

        rekening1.tampilkanInfo();

        System.out.println("\n=== Informasi Rekening 2 ===");

        rekening2.tampilkanInfo();

        // Contoh transaksi pada rekening1

        System.out.println("\n=== Transaksi pada Rekening 1 ===");

        rekening1.setorUang(250000);   // Menyetor uang

        rekening1.tarikUang(300000);    // Menarik uang

        // Contoh transaksi pada rekening2

        System.out.println("\n=== Transaksi pada Rekening 2 ===");

        rekening2.setorUang(150000);   // Menyetor uang

        rekening2.tarikUang(700000);    // Penarikan gagal jika saldo tidak mencukupi

    }

}