

10 Soal Latihan Java Tambahan

Soal 1: Class & Object Dasar

Judul: Data Karyawan **Topik:** Class, Object, Method **Deskripsi:** Buat sebuah class **Karyawan** dengan atribut **nama** (String), **idKaryawan** (String), dan **jabatan** (String). Buat juga sebuah method bernama **perkenalan()** yang mencetak informasi lengkap karyawan ke konsol. Di **main**, buatlah dua objek **Karyawan** dengan data yang berbeda dan panggil method **perkenalan()** dari masing-masing objek.

Soal 2: Enkapsulasi dan Constructor

Judul: Stok Produk **Topik:** Enkapsulasi, Constructor, Getter/Setter **Deskripsi:** Buat class **Produk** dengan atribut **private: kodeProduk** (String), **nama** (String), dan **harga** (double). Buat sebuah *constructor* untuk menginisialisasi semua atribut saat objek dibuat. Sediakan method *getter* untuk semua atribut dan *setter* khusus untuk atribut **harga**. Di **main**, buat sebuah objek **Produk**, tampilkan datanya, lalu coba ubah harganya dan tampilkan lagi.

Soal 3: Logika Percabangan & Method

Judul: Kategori Nilai Ujian **Topik:** Method, if-else if-else **Deskripsi:** Buat sebuah program yang menentukan predikat nilai seorang siswa. Buat sebuah method **tentukanPredikat(int nilai)** yang menerima nilai integer dan mengembalikan sebuah String berisi predikatnya. - Nilai ≥ 90 : Predikat "A" - Nilai ≥ 80 : Predikat "B" - Nilai ≥ 70 : Predikat "C" - Nilai < 70 : Predikat "D" Di **main**, minta pengguna memasukkan nilai, panggil method tersebut, dan cetak hasilnya.

Soal 4: Array dan Looping

Judul: Pencatat Pengunjung Harian **Topik:** Array, Looping for **Deskripsi:** Buat program untuk mencatat jumlah pengunjung sebuah toko selama seminggu. Buat sebuah array integer berukuran 7. Minta pengguna memasukkan jumlah pengunjung untuk setiap hari (Senin-Minggu). Setelah itu, hitung dan

tampilkan total pengunjung selama seminggu dan rata-rata pengunjung per hari.

Soal 5: Inheritance Sederhana

Judul: Sistem Transportasi **Topik:** Inheritance **Deskripsi:** Buat sebuah **superclass** bernama **Kendaraan** dengan atribut **merk** dan method **klakson()**. Kemudian, buat dua **subclass**, **Mobil** dan **SepedaMotor**, yang mewarisi dari **Kendaraan**. **Mobil** memiliki atribut tambahan **jumlahPintu**, sedangkan **SepedaMotor** memiliki **jenisStang**. Di **main**, buat objek dari kedua subclass dan panggil atribut serta method-nya.

Soal 6: Polymorphism

Judul: Suara Hewan **Topik:** Polymorphism, Method Overriding **Deskripsi:** Buat **superclass** **Hewan** dengan method **bersuara()**. Buat dua **subclass**, **Kucing** dan **Anjing**, yang meng-override method **bersuara()** untuk menghasilkan suara yang sesuai (“Meow” dan “Guk Guk”). Di **main**, buat sebuah array bertipe **Hewan** yang berisi objek **Kucing** dan **Anjing**. Lakukan perulangan pada array tersebut dan panggil method **bersuara()** dari setiap objek.

Soal 7: Interface

Judul: Perangkat Digital **Topik:** Interface **Deskripsi:** Buat sebuah **interface** bernama **DapatDiisiUlang** dengan dua method: **isiDaya()** dan **cekSisaBaterai()**. Buat dua class, **Smartphone** dan **Tablet**, yang mengimplementasikan interface tersebut. Setiap class harus memiliki logika sendiri untuk method-method dari interface. Di **main**, tunjukkan bahwa objek dari kedua class dapat memanggil method-method tersebut.

Soal 8: Array of Objects

Judul: Daftar Buku Perpustakaan **Topik:** Array of Objects **Deskripsi:** Buat class Buku dengan atribut judul, penulis, dan tahunTerbit. Di main, buat sebuah array untuk menampung 3 objek Buku. Izinkan pengguna untuk memasukkan data ketiga buku tersebut. Setelah semua data dimasukkan, tampilkan kembali daftar lengkap buku yang ada di perpustakaan.

Soal 9: Kombinasi Inheritance dan Enkapsulasi

Judul: Sistem Penggajian **Topik:** Inheritance, Enkapsulasi, Polymorphism **Deskripsi:** Buat superclass Pegawai dengan atribut private nama dan id. Buat subclass PegawaiTetap (dengan atribut tambahan gajiPokok) dan PegawaiHarian (dengan atribut upahPerJam dan jumlahJamKerja). Setiap class turunan harus memiliki method hitungGaji() yang di-override untuk menghitung total gaji berdasarkan aturannya masing-masing. Terapkan enkapsulasi dengan benar.

Soal 10: Penanganan Error Sederhana

Judul: Pembagian Bilangan **Topik:** Exception Handling (try-catch) **Deskripsi:** Buat program yang meminta pengguna memasukkan dua bilangan integer. Program akan menghitung hasil pembagian bilangan pertama dengan bilangan kedua. Gunakan blok try-catch untuk menangani potensi ArithmeticException jika pengguna memasukkan angka nol sebagai pembagi. Jika terjadi error, tampilkan pesan yang informatif.