

Tugas Pendahuluan

1. Konsep OOP I: Class dan Object

a. Jelaskan perbedaan antara class dan object dalam OOP!

→ Class merupakan abstraksi dari object, dimana memiliki atribut atau variabel dan method atau subprogram. Sedangkan object didefinisikan dalam sebuah class dan memiliki semua atribut dan method pada class tersebut.

b. Berikan contoh implementasi class dan object dalam Java!

→ Implementasi class

```
class DaftarKas {  
    String nama;  
    int nominal;  
  
    DaftarKas(String nama, int nominal) {  
        this.nama = nama;  
        this.nominal = nominal;  
    }  
  
    void display(){  
        System.out.println(nama);  
        System.out.println(nominal);  
    }  
}
```

Implementasi object

```
DaftarKas daftar = new DaftarKas("raden", 23000);
```

2. Konsep OOP II: Encapsulation, polymorphism, abstraction.
 - a. Jelaskan konsep Encapsulation dalam OOP.

→ Encapsulation dapat menggabungkan data dan method ke dalam class. Encapsulation bertujuan untuk membatasi akses langsung ke data dalam sebuah objek.
 - b. Apa perbedaan antara inheritance dan polymorphism?

→ Inheritance adalah proses di mana sebuah class mewarisi atribut dan method dari class lain, sedangkan polymorphism adalah objek yang dapat mengambil berbagai bentuk berbeda tergantung konteksnya.
 - c. Apa itu abstraction dalam OOP? Jelaskan perbedaannya dengan encapsulation!

→ Abstraction bertujuan untuk menyembunyikan detail kompleks dari suatu objek dan hanya menampilkan fitur atau fungsi yang penting saja. Sedangkan tujuan encapsulation adalah untuk menyembunyikan data dengan membatasi akses ke atribut menggunakan modifier seperti private.
3. Sebutkan minimal 3 perbedaan utama antara pemrograman berorientasi objek (OOP) dan pemrograman prosedural!

→ Perbedaan:

 - OOP lebih modular, mudah diperluas dengan konsep inheritance dan polymorphism.
 - Pemrograman prosedural lebih sederhana, cocok untuk proyek kecil.
 - Keamanan data lebih baik dalam OOP karena mendukung encapsulation.

Contoh::

<p>Contoh OOP:</p> <pre>class DaftarKas { String nama; DaftarKas(String nama) { this.nama = nama; } void display(){ System.out.println(nama); } }</pre>	<p>Contoh Pemrograman prosedural:</p> <pre>public class daftarKas { public static int dataKas(int bul1, int bul2) { return bul1 + bul2; } public static void main(String[] args) { int bulan1 = 15000; int bulan2 = 15000; int jumKas = dataKas(bulan1, bulan2); } }</pre>
---	---

}	System.out.println("Jumlah kas: " + jumKas);
	}
	}

4. Jelaskan peran konstruktor dalam Java!

→ konstruktor adalah method dalam sebuah class yang digunakan untuk menginisialisasi objek saat dibuat, dan konstruktor memiliki nama yang sama dengan nama class.

<pre>class DaftarKas { String nama; DaftarKas (String nama) { nama = namaMahasiswa; } }</pre>
--

5. Jelaskan perbedaan antara static method dan non-static method dalam Java!

→ static method adalah metode yang dapat dipanggil tanpa harus membuat objek dari kelasnya. Sedangkan non-static method adalah metode yang harus dipanggil melalui objek.