

Nama : Latifah Tasliyatunadiyah
NIM : 20220040140
Kelas : TI 22 H
Matkul : Pemrograman Berorientasi Objek
Sesi : 2

Pertanyaan 1

Jelaskan apa yang dimaksud dengan class dan object!

Jawaban 1

Class

Class adalah sebuah template atau blueprint yang digunakan untuk mendefinisikan atribut dan perilaku yang akan dimiliki oleh objek yang dibuat dari class tersebut. Class mendefinisikan struktur data dan fungsi yang akan digunakan oleh objek. Class juga dapat berisi variabel data (attributes), metode (methods), dan konstruktor. Class tidak menggunakan memori ketika didefinisikan, tetapi memori akan dialokasikan ketika sebuah objek dibuat dari class tersebut .

Object

Object adalah instance dari sebuah class. Setiap objek adalah representasi dari class dan dapat menggunakan atribut dan metode yang didefinisikan dalam class tersebut. Objek memiliki state (representasi oleh atribut), behavior (representasi oleh metode), dan identity (memberikan nama unik kepada objek). Objek dibuat menggunakan keyword new dalam bahasa pemrograman seperti Java, dan setiap objek menggunakan memori yang dialokasikan untuk menyimpan state-nya.

Pertanyaan 2

Sebutkan dan jelaskan jenis-jenis method !

Jawaban 2

Jenis-jenis Methode

1. Metode Predefined (Metode Pragma): Metode ini sudah didefinisikan dalam library kelas Java dan dapat digunakan langsung dalam program tanpa perlu deklarasinya. Contohnya termasuk metode length(), equals(), compareTo(), dan sqrt() yang dapat digunakan untuk berbagai tugas seperti operasi matematika, perbandingan string, dan lainnya. Metode ini sering disebut sebagai metode standar atau metode bawaan .
2. Metode User-defined (Metode Pengguna): Metode yang dibuat oleh programmer sendiri untuk melakukan tugas-tugas tertentu sesuai dengan kebutuhan program. Metode ini dapat dideklarasikan dengan atribut-atribut seperti jenis akses (public, private, protected, default), tipe pengembalian (return type), nama metode, dan daftar parameter. Metode ini dapat mencakup semua jenis tugas yang tidak dapat dilakukan oleh metode predefined .
3. Metode Static: Metode ini adalah bagian dari kelas, bukan dari objek kelas. Metode static dapat dipanggil tanpa perlu membuat instance dari kelas. Contohnya adalah metode main() yang digunakan sebagai titik masuk program Java. Metode static sering digunakan untuk operasi yang tidak memerlukan state objek .
4. Metode Instance: Metode ini memerlukan objek dari kelas untuk dipanggil. Metode ini memiliki akses ke variabel dan metode lain dalam kelas. Metode instance dapat dibagi

menjadi dua kategori utama: Accessor Method: Metode yang digunakan untuk mengambil nilai dari variabel instance. Biasanya, metode ini memiliki nama yang diawali dengan kata get.

5. Mutator Method: Metode yang digunakan untuk mengubah nilai dari variabel instance. Biasanya, metode ini memiliki nama yang diawali dengan kata set.
6. Metode Abstract: Metode ini dideklarasikan dalam kelas abstrak dan tidak memiliki implementasi di dalam kelas itu sendiri. Metode abstrak dapat dipanggil melalui referensi objek yang merujuk ke kelas abstrak atau kelas turunannya. Metode abstrak sering digunakan dalam konsep pewarisan dan polimorfisme dalam pemrograman berorientasi objek 4.

Pertanyaan 3

Berdasarkan gambar berikut ini, jelaskan masing-masing bagian sesuai dengan nomor yang ada!

```
public class Komputer { 1
    String jenis_komputer; 2
    private String merk;

    public void setDataKomputer(String jenis, String merk){ 3
        jenis_komputer = jenis;
        this.merk = merk;
    }

    public String getJenis(){ 4
        return jenis_komputer;
    }

    public String getMerk(){ 5
        return merk;
    }

    public static void main(String[] args){
        Komputer mykom = new Komputer(); 6
        mykom.setDataKomputer("LAPTOP", "MACBOOK"); 7
        System.out.println(mykom.getJenis());
        System.out.println(mykom.getMerk()); 8
    }
}
```

Jawaban 3

1. `public class Komputer {` (1) Pada baris satu pada kode tersebut ialah mendeklarasikan kelas Komputer sebagai public, yang berarti kelas ini dapat diakses dari kelas lain dalam aplikasi.
2. `String jenis_komputer; private String merk;` (2) Pada nomor 2 itu ada kode yang mendeklarasikan `jenis_komputer` dan `merk` sebagai variable ber tipe String, tetapi pada `merk` terdapat penggunaan modifier `private`, yang berarti variabel ini hanya dapat diakses dari dalam kelas Komputer itu sendiri.
3. `public void setDataKomputer(String jenis, String merk) { this.jenis_komputer = jenis; this.merk = merk;` (3) Pada nomor 3 Mendefinisikan metode `setDataKomputer` yang memungkinkan pengaturan nilai untuk `jenis_komputer` dan `merk`. Metode ini menerima dua parameter, `jenis` dan `merk`, yang keduanya bertipe String. Menggunakan kata kunci `this` untuk merujuk ke variabel instance `jenis_komputer` dan `merk` dari kelas Komputer, dan menetapkan nilai dari parameter `jenis` ke variabel tersebut.

4. `public String getJenis() { return jenis_komputer; }` (4) Mendefinisikan metode `getJenis` yang mengembalikan nilai dari variabel instance `jenis_komputer`. dan di bawah nya ada `return` yang berfungsi mengembalikan nilai dari variabel `jenis_komputer`.
5. `public String getMerk() { return merk; }` (5) Mendefinisikan metode `getMerk` yang mengembalikan nilai dari variabel instance `merk`. dan di bawah nya ada `return` yang berfungsi mengembalikan nilai dari variabel `merk`.
6. `Komputer mykom = new Komputer();` (6) Membuat objek `mykom` dari kelas `Komputer` menggunakan konstruktor default (yang dianggap ada oleh Java jika tidak ada konstruktor lainnya yang ditentukan).
7. `mykom.setDataKomputer("LAPTOP", "MACBOOK");` (7) Memanggil metode `setDataKomputer` pada objek `mykom` untuk mengatur `jenis_komputer` menjadi "LAPTOP" dan `merk` menjadi "MACBOOK".
8. `System.out.println(mykom.getJenis());` `System.out.println(mykom.getMerk());` (8) Mencetak jenis komputer yang telah disetel sebelumnya ke konsol. Mencetak merk komputer yang telah disetel sebelumnya ke konsol.

Pertanyaan 4

Tuliskan perbaikan pada kode program berikut ini, agar kode program tersebut dapat dicompile sehingga program akan berjalan dengan benar!

```

2  public class HandPhone {
3      String jenis_hp;
4      int tahun_pembuatan;
5
6      String setDataHP(String jenis_hp, int tahun_pembuatan){
7          jenis_hp = jenis_hp;
8          tahun_pembuatan = tahun_pembuatan;
9      }
10
11     String getJenisHP() {
12
13     }
14
15     String getTahunPembuatan() {
16
17     }
18
19     public static main void(String args[]){
20         HandPhone hp = new HandPhone();
21         hp.setDataHP(jenis_hp, tahun_pembuatan);
22         hp.getJenisHP();
23         hp.getTahunPembuatan();
24     }
25 }

```

Note : Pengumpulan tugas wajib menggunakan link dari github atau gitlab

Jawaban

Hasil perbaikan kode tersebut adalah

```

public class HandPhone {
    private String jenisHp;
    private int tahunPembuatan;
    public String setDataHP(String jenisHp, int tahunPembuatan) {
        this.jenisHp = jenisHp;
        this.tahunPembuatan = tahunPembuatan;
        return jenisHp;
    }
    public String getJenisHP() {
        return jenisHp;
    }
    public int getTahunPembuatan() {
        return tahunPembuatan;
    }
}

```

```
public static void main(String[] args) {  
    HandPhone hp = new HandPhone();  
    hp.setDataHP("Redmi Note 10S", 2021);  
    System.out.println("Jenis HP: " + hp.getJenisHP());  
    System.out.println("Tahun Pembuatan: " + hp.getTahunPembuatan());  
}  
}
```

Hasil Dari kode tersebut adalah :

