

PEMROGRAMAN BERORIENTASI OBJEK SESI_2

Nama : Andrean Maulana
NIM : 20220040300
Prodi / Kelas : Teknik Informatika / TI22C

Soal:

1. Jelaskan apa yang dimaksud dengan class dan object !
2. Sebutkan dan jelaskan jenis-jenis method
3. Berdasarkan gambar berikut ini, jelaskan masing-masing bagian sesuai dengan nomor yang ada !

Jawaban:

1. Jelaskan apa yang dimaksud dengan class dan object

Dalam pemrograman berorientasi objek (OOP), konsep class dan object adalah dua pilar utama yang membentuk dasar dari cara pemrograman dilakukan. Berikut ini penjelasan masing-masing:

- **Class**

Class adalah cetakan atau blueprint untuk membuat sesuatu yang disebut object. Dalam class, didefinisikan karakteristik dari objek yang akan dibuat, termasuk data yang bisa disimpan (biasanya disebut sebagai fields atau properties) dan operasi atau fungsi yang bisa dilakukan (biasanya disebut sebagai methods). Class memberikan struktur yang diperlukan untuk menggabungkan data dan fungsi-fungsi yang beroperasi pada data tersebut dalam satu unit yang rapi.

Class sendiri tidak mengonsumsi sumber daya apa pun saat program dijalankan; ia hanya mendefinisikan struktur dan perilaku yang akan dimiliki oleh objek-objek yang diciptakan darinya.

- **Object**

Object adalah instansi atau realisasi dari sebuah class. Saat sebuah program dijalankan, object dibuat menggunakan cetakan yang disediakan oleh class dan memori dialokasikan untuk object tersebut. Setiap object dapat memiliki data yang berbeda yang ditentukan oleh fields pada classnya, namun akan berbagi metode yang sama. Dengan kata lain, jika Anda memiliki class "Mobil" yang memiliki properties seperti "warna" dan "merk" serta metode seperti "maju" dan "berhenti", maka setiap object yang diciptakan dari class "Mobil" akan memiliki properties dan metode tersebut. Namun, nilai dari properties (misalnya, warna atau merk) bisa berbeda antara satu object dengan object lainnya.

Sebagai analogi, Anda bisa memikirkan class sebagai desain arsitektur sebuah rumah yang menentukan dimana ruangan, pintu, dan jendela berada (tetapi tanpa rumah fisik yang

sebenarnya dibangun), sementara object adalah rumah fisik yang dibangun menggunakan desain tersebut, dengan warna cat tertentu, jenis jendela tertentu, dll.

Dalam pemrograman berorientasi objek, class dan object memungkinkan programmer untuk membuat program yang lebih modular, mudah untuk di-maintain, dan reusable, dengan mengelompokkan data dan operasi yang terkait menjadi satu unit.

2. sebutkan dan jelaskan jenis-jenis method!

Dalam pemrograman berorientasi objek (OOP), metode (methods) adalah fungsi-fungsi yang terasosiasi dengan class dan digunakan untuk mendefinisikan perilaku dari objek. Ada beberapa jenis metode berdasarkan cara kerja dan tujuan penggunaannya. Berikut adalah beberapa jenis metode yang umum digunakan:

- **Constructor Method**

Constructor adalah metode khusus yang otomatis dipanggil pada saat pembuatan (instansiasi) objek. Tujuan utama dari constructor adalah untuk inisialisasi objek, seperti mengatur nilai awal untuk properti objek. Dalam banyak bahasa pemrograman, constructor memiliki nama khusus, seperti `_init_` di Python atau nama yang sama dengan nama class di Java dan C++.

- **Destructor Method**

Destructor adalah metode yang dipanggil secara otomatis ketika objek tidak lagi diperlukan atau ketika objek akan dihancurkan. Destructor digunakan untuk melakukan pembersihan sumber daya, seperti menutup file atau koneksi database. Seperti constructor, destructor juga memiliki nama khusus tergantung pada bahasa pemrograman, seperti `_del_` di Python.

- **Accessor Methods**

Accessor methods, juga dikenal sebagai getter, digunakan untuk membaca nilai dari properti privat objek. Metode ini memberikan cara yang terkontrol untuk mengakses data dari objek tanpa memberikan akses langsung ke properti tersebut. Ini membantu dalam menjaga integritas data.

- **Mutator Methods**

Mutator methods, juga dikenal sebagai setter, digunakan untuk mengatur atau memodifikasi nilai dari properti privat objek. Seperti accessor, mutator memberikan cara yang terkontrol untuk mengubah data dari objek, memungkinkan validasi atau pemrosesan data sebelum diubah.

- **Instance Methods**

Instance methods adalah metode yang memerlukan instansi dari class untuk dipanggil dan dapat mengakses data dan properti yang spesifik untuk instansi tersebut. Dalam definisi metode, instance methods selalu mengambil minimal satu parameter pertama yang merujuk ke objek itu sendiri, seperti `self` di Python.

- **Static Methods**

Static methods adalah metode yang, meskipun terasosiasi dengan class, tidak memerlukan instansi objek untuk dipanggil. Static methods tidak dapat mengakses atau memodifikasi

keadaan objek dan umumnya digunakan untuk operasi yang tidak bergantung pada data objek. Dalam banyak bahasa pemrograman, static methods dideklarasikan dengan kata kunci static.

- **Class Methods**

Class methods mirip dengan static methods karena mereka juga dipanggil tanpa instansi objek. Namun, class methods mengambil referensi ke class sebagai parameter pertama mereka (biasanya dinamai `cls` di Python) dan dapat mengakses variabel class atau metode lain yang terkait dengan class tersebut.

Setiap jenis metode ini memiliki peran dan penggunaan spesifik dalam desain dan implementasi class dalam pemrograman berorientasi objek, memungkinkan enkapsulasi, pewarisan, dan polimorfisme.

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

```
public class Komputer {  
    String jenis_komputer;  
    private String merk;  
  
    public void setDataKomputer(String jenis, String merk){  
        jenis_komputer = jenis;  
        this.merk = merk;  
    }  
  
    public String getJenis(){  
        return jenis_komputer;  
    }  
  
    public String getMerk(){  
        return merk;  
    }  
  
    public static void main(String[] args){  
        Komputer mykom = new Komputer();  
        mykom.setDataKomputer("LAPTOP", "MALIBOOK");  
        System.out.println(mykom.getJenis());  
        System.out.println(mykom.getMerk());  
    }  
}
```

The image shows a Java code snippet for a class named `Komputer`. The code is annotated with red boxes and numbers 1 through 8, indicating specific parts of the code:

- 1. `public class Komputer {`
- 2. `String jenis_komputer;`
- 3. `private String merk;`
- 4. `public void setDataKomputer(String jenis, String merk){`
- 5. `jenis_komputer = jenis;`
- 6. `this.merk = merk;`
- 7. `public String getJenis(){`
- 8. `return jenis_komputer;`

1. Nama Class
2. Attribute

- 3. Method Declaration (Setter)
 - 4. Method Declaration (Getter)
 - 5. Method Declaration (Getter)
 - 6. Instantiante
 - 7. Method setter (prosedur)
 - 8. Method getter (fungsi)
-
- Method
- Object