**PEMROGRAMAN BERORIENTASI OBJEK SESI\_2**

Nama : Juniar Akhsan

Kelas :TI22C

NIM : 20220040085

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: Dalam pemrograman berorientasi objek (OOP), sebuah "class" adalah suatu blueprint atau cetak biru untuk menciptakan objek. Class menyediakan struktur dasar atau kerangka kerja yang mendefinisikan atribut (data) dan metode (fungsi) yang akan dimiliki oleh objek yang dibuat dari class tersebut. Class dapat dianggap sebagai jenis atau tipe data baru yang dapat kita definisikan sendiri.

* **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.

* Contoh pembuatan objek dari class Kendaraan:

Dalam contoh ini, mobil\_saya dan motor\_saya adalah dua objek yang dibuat dari class Kendaraan. Setiap objek memiliki nilai sendiri untuk atribut jenis dan warna, tetapi struktur dan metode dari class tetap sama.

Dengan menggunakan konsep class dan objek, pemrogram dapat mengorganisir dan mengelola kode secara lebih terstruktur dan dapat digunakan kembali. OOP memungkinkan untuk menggabungkan data dan perilaku dalam satu entitas yang dapat diperluas dan diubah sesuai kebutuhan.

1. **sebutkan dan jelaskan jenis-jenis method!**

Dalam pemrograman berorientasi objek, method adalah fungsi atau perilaku yang terkait dengan suatu objek atau kelas. Berikut adalah beberapa jenis method yang umum digunakan:

1. **Method Instansiasi (Constructor):**

* Deskripsi: Method khusus yang dipanggil saat objek dari suatu kelas dibuat.
* Contoh: \_\_init\_\_(self, parameter)

1. **Method Setter dan Getter:**

**Setter:**

* Deskripsi: Mengatur nilai suatu atribut objek.
* Contoh: set\_nama(self, nama)

**Getter:**

* Deskripsi: Mengembalikan nilai suatu atribut objek.
* Contoh: get\_nama(self)

1. **Method Instansiasi Lainnya:**

* Deskripsi: Method lain yang berkaitan dengan pembuatan dan pengelolaan objek, di luar constructor.
* Contoh: create\_instance(self, parameter)

**Method Kelas (Class Method):**

* Deskripsi: Method yang berkaitan dengan kelas secara keseluruhan, bukan objek individu. Dapat diakses menggunakan kelas itu sendiri.
* Contoh: @classmethod, class\_method(cls, parameter)

1. **Method Statik (Static Method):**

* Deskripsi: Method yang tidak berkaitan dengan objek atau kelas. Tidak memiliki parameter self atau cls.
* Contoh: @staticmethod, static\_method(parameter)

1. **Method Penghancur (Destructor):**

* Deskripsi: Method khusus yang dipanggil saat objek dihancurkan atau keluar dari lingkup.
* Contoh: \_\_del\_\_(self)

1. **Method String Representasi (String Representation Method):**

* Deskripsi: Mengembalikan representasi string dari objek, biasanya digunakan untuk keperluan debugging atau untuk memberikan representasi yang lebih informatif.
* Contoh: \_\_str\_\_(self)

1. **Method Operator Overloading:**

* Deskripsi: Memungkinkan kelas untuk mendefinisikan perilaku operasi operator tertentu.
* Contoh: \_\_add\_\_(self, other), \_\_eq\_\_(self, other)

1. **Method Iterasi:**

* Deskripsi: Memungkinkan objek untuk diiterasi, biasanya menggunakan loop for.
* Contoh: \_\_iter\_\_(self), \_\_next\_\_(self)

1. **Method Callback:**

* Deskripsi: Memungkinkan objek untuk memanggil fungsi atau method lainnya sebagai respons terhadap suatu peristiwa atau aksi.
* Contoh: register\_callback(self, callback\_function)

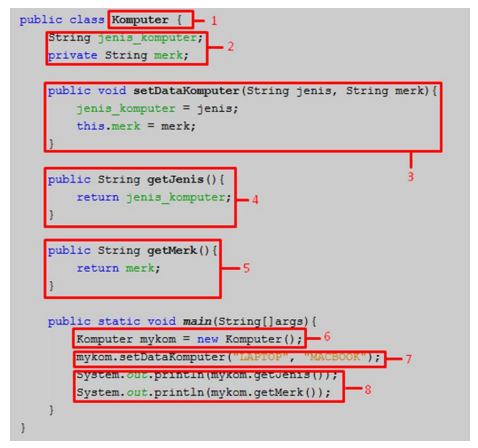
1. **Method Abstract:**

* Deskripsi: Method tanpa implementasi dalam kelas induk, harus diimplementasikan oleh kelas turunan.
* Contoh: @abstractmethod, def abstract\_method(self)

Penting untuk diingat bahwa tidak semua jenis method akan ditemukan dalam setiap bahasa pemrograman, dan beberapa bahasa mungkin memiliki pendekatan yang berbeda terkait dengan konsep-konsep ini. Contoh-contoh di atas lebih umum digunakan dalam konteks bahasa pemrograman seperti Python atau Java yang mendukung pemrograman berorientasi objek.

Top of Form

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

****

1. Nama Class
2. Attribute
3. Method Declaration (Setter)
4. Method Declaration (Getter) Method
5. Method Declaration (Getter)
6. Instantiante
7. Method setter (prosedur) Object
8. Method getter (fungsi)