

NO 1

nama : Ranti indriyani
kelas : TI22C
NIM : 20220040002

1. Jelaskan apa yang dimaksud dengan class dan object!
jawaban :

CLASS :

Class merupakan cetak biru atau kerangka dalam pembuatan program atau class adalah Class merupakan prototipe yang mendefinisikan attribute dan behavior secara umum. Saat implementasi kedalam sebuah program attribute dimodelkan sebagai variabel dan behavior dimodelkan sebagai method atau yang lebih dikenal dikalangan programmer adalah function.

OBJECT :

object adalah hasil instance atau penciptaan dari sebuah class. dan Pembuatan sebuah obyek pada umumnya dilakukan didalam method.

NO 2

2. Sebutkan dan jelaskan jenis-jenis method !
jawaban :

1. METHOD Berdasarkan sifatnya:

- a. Method Void: Method yang tidak mengembalikan nilai. Contohnya:

```
public void cetakNama() {  
    System.out.println("Nama saya Ranti");  
}
```

ini merupakan contoh Method berupa prosedur tanpa parameter

- b. Method Return: Method yang mengembalikan nilai. Contohnya:

```
public String getName() {  
    return "Ranti";  
}
```

2. METHOD Berdasarkan aksesnya:

- a. Public Method: Method yang dapat diakses dari mana saja.
- b. Private Method: Method yang hanya dapat diakses dari kelasnya sendiri.
- c. Protected Method: Method yang hanya dapat diakses dari dalam kelasnya sendiri dan kelas turunannya.

3. METHOD Berdasarkan statisnya:

- a. Static Method: Method yang dapat dipanggil tanpa perlu membuat objek terlebih dahulu. Contohnya:

```
public static void main(String[] args) {  
    System.out.println("Hello World!");  
}
```

- b. Non-Static Method: Method yang harus dipanggil dari sebuah objek. Contohnya:

b. Setter: Method yang digunakan untuk mengubah nilai dari sebuah variabel. Contohnya:

```
public void setName(String nama) {  
    this.nama = nama;  
}
```

dan ini Contoh Penerapan

Berikut adalah contoh penerapan beberapa jenis method:

```
public class Contoh {  
  
    // Public static method  
    public static void main(String[] args) {  
        System.out.println("Hello World!");  
    }  
  
    // Private method  
    private String getName() {  
        return "Bard";  
    }  
  
    // Protected method  
    protected void cetakNama() {  
        System.out.println("Nama saya " + getName());  
    }  
  
    // Public non-static method  
    public void setName(String nama) {  
        this.nama = nama;  
    }  
  
    // Overriding method  
    @Override  
    public String toString() {  
        return "Contoh{" +  
            "nama=" + nama + "}" +  
            "};  
    }  
}
```

b. Non-Static Method: Method yang harus dipanggil dari sebuah objek. Contohnya:

```
public class Contoh {  
    public void cetakNama() {  
        System.out.println("Nama saya Bard");  
    }  
}  
  
public class Main {  
    public static void main(String[] args) {  
        Contoh contoh = new Contoh();  
        contoh.cetakNama();  
    }  
}
```

4. METHOD Berdasarkan parameternya:

- a. Method tanpa parameter: Method yang tidak menerima parameter.
- b. Method dengan parameter: Method yang menerima parameter. Contohnya:

```
public void cetakNama(String nama) {  
    System.out.println("Nama saya " + nama);  
}
```

5. METHOD Berdasarkan fungsinya:

- a. Getter: Method yang digunakan untuk mendapatkan nilai dari sebuah variabel. Contohnya:

```
public String getNama() {  
    return "Ranti";  
}
```

NO 3

3. `public class Komputer {` —————→ 1 **Class** = cetak biru atau kerangka dalam pembuatan program

`String jenis_komputer;`
`private String merk;` —————→ 2 **Atribut / deklarasi atribut** = variabel yang dideklarasikan di dalam kelas. Atribut dapat digunakan untuk menyimpan data yang terkait dengan objek kelas tersebut.

`public void setDataKomputer(String jenis, String merk){`
`this.jenis_komputer = jenis;`
`this.merk = merk;`
`}` —————→ 3 **Method setter** adalah method yang digunakan untuk mengubah nilai atribut.

`public String getJenis(){`
`return jenis_komputer;`
`}` —————→ 4 **Method getter** adalah method yang digunakan untuk mendapatkan nilai atribut.

`public String getMerk() {`
`return merk;`
`}` —————→ 5 **Method getter** adalah method yang digunakan untuk mendapatkan nilai atribut.

`public static void main(String[] args){`

`Komputer mykom = new Komputer();` —————→ 6 **Instansiasi objek** adalah proses pembuatan objek baru dari sebuah kelas.

`mykom.setDataKomputer("LAPTOP", "FACEBOOK");` —————→ 7 **Pemanggilan method** adalah proses menjalankan method pada suatu objek

`System.out.println(mykom.getJenis());`
`System.out.println(mykom.getMerk());` —————→ 8 **pencetakan nilai atribut objek.**

`}`

- `System.out.println()` adalah method yang digunakan untuk mencetak teks ke konsol.
- `mykom` adalah variabel yang menampung objek.
- `getJenis()` dan `getMerk()` adalah method yang digunakan untuk mendapatkan nilai atribut jenis dan merk dari objek.