

<b>Nama &amp; NPM</b>	<b>Topik:</b>	<b>Tanggal:</b>
<b>PESINOVITASARI G1F024008</b>	<b>Kelas, objek, method</b>	<b>20 September 2024</b>

## LATIHAN 1

### [No. 1] Identifikasi Masalah:

- 1) Uraikan permasalahan dan variabel

#### Contoh 1:

```
public class Manusia { // deklarasi kelas
    // deklarasi variabel
    String nama;
    String rambut;

    // deklarasi constructor tanpa parameter
    public Manusia() {
        System.out.println("Kelas Manusia tanpa nama");
    }
}
```

#### Latihan 1:

- 1.1. Analisa ciri-ciri umum Kelas Manusia yang dapat menjadi
  - a. atribut variabel, dan
  - b. perilaku/ behavior untuk method!

### [No.1] Analisis dan Argumentasi

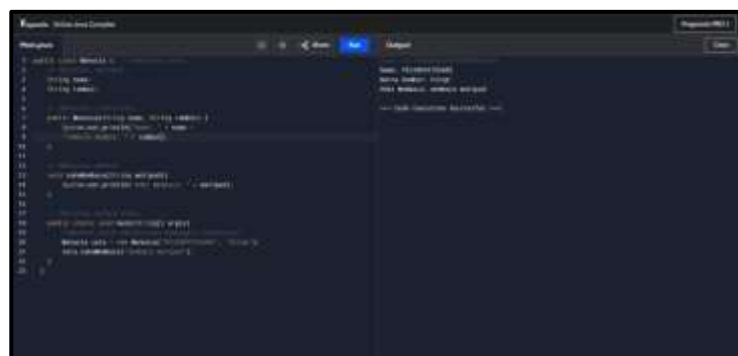
Pada soal diatas kita di suruh membuat program yang outputnya menampilkan nama, warna rambut dan hobi. Kita membuat publi clas manusia untuk menampilkan nama dan warna rambut. Membuat void suka membaca dan membuat void static untuk memasukkan data nama dan warna rambut.

### [No.1 ] Penyusunan Algoritma dan Kode Program

#### 1) Algoritma

- (a).Membuat public class manusia
- (b).Menambahkan string nama dan string rambut.
- (c).Membuat public class manusia untuk mendeklarasikan nama dan warna rambut.
- (d).Membuat void sukaMembaca untuk mendeklarasikan hobi.
- (e).Membuat public static void main untuk menampilkan data nama , warna rambut, dan hobi.
- (f).Menambahkan system.out.println untuk menampilkan hasil outputnya.

#### 2) Kode program dan luaran



- a) Screenshot/ Capture potongan kode dan hasil luaran  
Berikan komentar pada kode yang di Screenshot

- b) Analisa luaran yang dihasilkan

Luaran yang dihasilkan sudah sesuai dengan permintaan soal yang program yang dijalankan sudah benar.

### [No.1] Kesimpulan

#### Analisa

Pada program diatas membuat public class manusia dengan menggunakan tipe data string untuk mendeklarasikan data nama dan rambut. Menambahkan void suka membaca untuk mendeklarasikan hobi. Dan menambahkan public static void main untuk memasukkan data nama, warna rambut dan hobi.

## LATIHAN 2

### [No. 2] Identifikasi Masalah:

- 1) Uraikan permasalahan dan variabel

**Contoh 2:** Salin dan tempel kode program berikut ke Eclipse atau JDoodle.

```
public class Ortu {  
    //deklarasi constructor  
    public Ortu(String nama, String rambut) {  
        //nama dan rambut adalah variabel constructor  
        System.out.println(" Nama saya : "+ nama +  
            "\n Warna Rambut : " + rambut);  
    }  
    public static void main (String[] args) {  
        Ortu satu = new Ortu("Putri", "hitam");  
    }  
}
```

#### Luaran 2:

```
Nama saya : Putri  
Warna Rambut : hitam
```

#### Latihan 2:

- 2.1. Susun kembali kode di contoh 2 dengan menambahkan data ciri-ciri Anda di dalam variabel constructor!  
2.2. Apabila nanti Anda akan memiliki keturunan, analisa sifat (atribut), constructor, dan perilaku positif (behavior) apa yang akan diturunkan?

### [No.2] Analisis dan Argumentasi

Membuat program class ortu dengan menggunakan tipe data string untuk mendeklarasikan data nama, rambut, warna kulit, warna mata dan bentuk rambut. Menambahkan system.out.println untuk menampilkan data nama, warna rambut, warna kulit, warna mata, dan bentuk rambut. Dan gunakan + untuk memanggil data tersebut. Menambahkan public static void untuk menampilkan isi dari data nama, warna rambut, warna kulit, warna mata, dan bentuk rambut.

### [No.2 ] Penyusunan Algoritma dan Kode Program

- 1) Algoritma  
(a).Membuat public class ortu

- (b).Mendeklarasikan data nama, rambut, warna kulit, warna mata, dan bentuk rambut dengan menggubajan tipe data string.
- (c).System.out.println nama saya dan tambahkan + nama untuk memanggil variable nama, System.out.println warna rambut dan tambahkan + rambut untuk memanggil variable rambut, System.out.println warna kulit dan tambahkan + warna kulit untuk memanggil variable warna kulit, System.out.println warna mata dan tambahkan = warna mata untuk memanggil variable warna mata, System.out.println bentuk rambut dan tambah + bentuk rambut untuk memanggil variable bentuk rambut.
- (d).Menampilkan data dari nama, warna rambut, warna kulit, warna mata, dan bentuk rambut.

## 2) Kode program dan luaran

The screenshot shows a Java IDE with a file named 'Main.java'. The code defines a public class 'Ortu' with attributes 'nama', 'warnaRambut', 'warnaKulit', 'warnaMata', and 'bentukRambut'. It includes a constructor and a static method 'main' that creates an instance of 'Ortu' and prints its details. The output window on the right shows the execution results: 'Nama saya: PELENTILISIAI', 'warna rambut: hitam', 'warna kulit: sawo matang', 'warna mata: hitam', and 'bentuk rambut: lurus'.

```

1 public class Ortu {
2     //atribut/variabel, untuk menunjukkan data yang akan digunakan
3     public String nama, String rambut, String warnaKulit, String warnaMata, String
4     bentukRambut;
5     //konstruktor, untuk menginisialisasi variabel-variabel yang akan digunakan
6     //dalam program
7     //konstruktor
8     public Ortu(String nama, String rambut, String warnaKulit, String warnaMata, String
9     bentukRambut) {
10        this.nama = nama;
11        this.rambut = rambut;
12        this.warnaKulit = warnaKulit;
13        this.warnaMata = warnaMata;
14        this.bentukRambut = bentukRambut;
15    }
16
17    //metode, untuk melakukan aksi yang akan dilakukan
18    public static void main(String[] args) {
19        //membuat objek Ortu
20        Ortu oru = new Ortu("PELENTILISIAI", "hitam", "sawo matang", "hitam", "lurus");
21        //menampilkan data
22        System.out.println("Nama saya: " + oru.nama);
23        System.out.println("warna rambut: " + oru.rambut);
24        System.out.println("warna kulit: " + oru.warnaKulit);
25        System.out.println("warna mata: " + oru.warnaMata);
26        System.out.println("bentuk rambut: " + oru.bentukRambut);
27    }
28 }

```

Output:

```

Nama saya: PELENTILISIAI
warna rambut: hitam
warna kulit: sawo matang
warna mata: hitam
bentuk rambut: lurus

```

a)Screenshot/ Capture potongan kode dan hasil luaran  
 Beri komentar pada kode yang di Screenshot

b)Analisa luaran yang dihasilkan  
 Luaran yang dihasilkan sudah sesuai dengan permintaan soal yang program yang dijalankan sudah benar.

## [No.2] Kesimpulan

### Analisa

Pada program diatas Membuat public class ortu Mendeklarasikan data nama, rambut, warna kulit, warna mata, dan bentuk rambut dengan menggubajan tipe data string.

System.out.println nama saya dan tambahkan + nama untuk memanggil variable nama, System.out.println warna rambut dan tambahkan + rambut untuk memanggil variable rambut, System.out.println warna kulit dan tambahkan + warna kulit untuk memanggil variable warna kulit, System.out.println warna mata dan tambahkan = warna mata untuk memanggil variable warna mata, System.out.println bentuk rambut dan tambah + bentuk rambut untuk memanggil variable bentuk rambut. Menampilkan data dari nama, warna rambut, warna kulit, warna mata, dan bentuk rambut.

## LATIHAN 3

### [No. 3] Identifikasi Masalah:

Uraikan permasalahan dan variable

**Contoh 3:** Salin dan tempel kode program berikut ke Eclipse atau JDoodle.

```
public class Manusia {  
    //deklarasi atribut Manusia dalam variabel  
    String nama, rambut;  
  
    //deklarasi constructor  
    public Manusia1(String nama, String rambut) {  
        System.out.println(" Nama saya : "+ nama +  
            "\n Warna Rambut : " + rambut);  
    }  
  
    //deklarasi method  
    void sukaNonton(String film) {  
        System.out.println(" Hobi Menonton : " + film);  
    }  
  
    //deklarasi method utama  
    public static void main( String[] args) {  
        Manusia satu = new Manusia("Putri", "hitam");  
        satu.sukaNonton("Drakor");  
    }  
}
```

**Luaran 3:**

```
Nama saya : Putri  
Warna Rambut : hitam  
Hobi Menonton : Drakor
```

**Latihan 3:**

- 3.1. Analisa perbedaan deklarasi constructor, method, dan method utama!
- 3.2. Tentukan kapan Anda perlu menggunakan constructor dan method?
- 3.3. Uraikan perbedaan berikut:
  - a) constructor overloading dan overriding
  - b) method overloading, dan method overriding
  - c) method yang mengembalikan nilai dan method tidak mengembalikan nilai

### [No.3] Analisis dan Argumentasi

Perbedaan antara deklarasi constructor, method, dan method utama yaitu constructor digunakan untuk menginisialisasi objek, nama dengan nama kelas dan tidak memiliki tipe data kembalian(return type). Method berfungsi untuk melakukan suatu tindakan atau menghitung nilai dan memiliki tipe data kembalian yang berupa nilai seperti ( int, string) atau void( tidak mengembalikan nilai) . Sedangkan method utama yang merupakan titik program java nya dan selalu memiliki signature public static void main(String[] args).

Constructor digunakan saat ingin memastikan bahwa objek diinisialisasi dengan nilai tertentu saat dibuat. Sedangkan method digunakan untuk melakukan tindakan tertentu pada objek, seperti memproses data atau memberikan informasi. Method bisa dipanggil kapan saja setelah objek diciptakan.

a). Constructor overloading dan overriding

Overloading memiliki beberapa constructor dengan nama sama tetapi parameter berbeda dalam satu kelas

Overriding tidak berlaku untuk constructor, karena constructor tidak dapat diwarisi. Overriding biasa digunakan untuk method dalam subclass yang memiliki implementasi berbeda dari superclass.

b). Method overloading dan overriding

Overloading memiliki method yang sama tetapi parameter berbeda

Overriding mengganti implementasi method dari superclass dalam subclass.

c). Method yang mengendalikan nilai dan method yang tidak mengembalikan nilai

Method mengembalikan nilai memiliki tipe kembalian yang menggunakan kata kunci return untuk mengembalikan nilai.

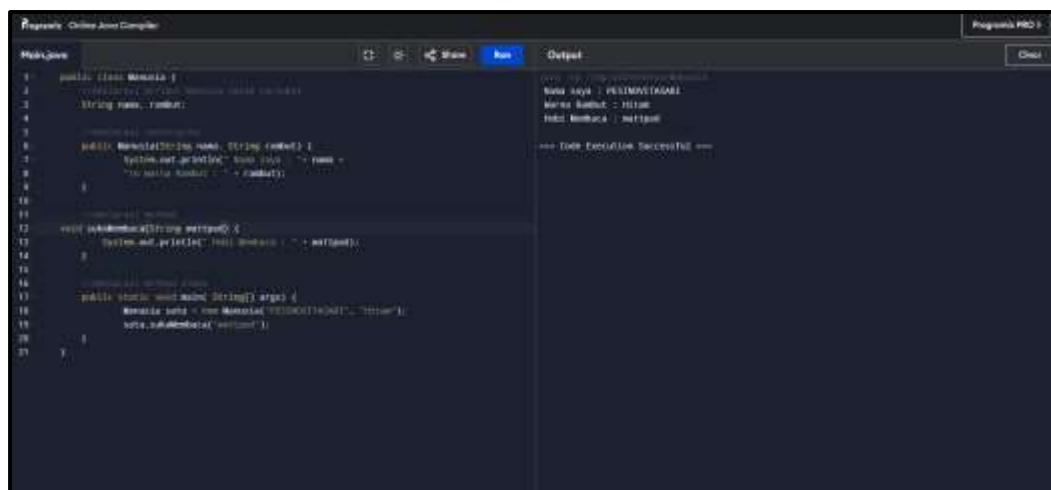
Method yang tidak mengembalikan nilai dideklarasikan dengan tipe kembalian void, tidak menggunakan return untuk mengembalikan nilai, hanya melakukan tindakan.

### [No.3 ] Penyusunan Algoritma dan Kode Program

#### 1) Algoritma

- (a).Membuat program public class manusia.
- (b).Menambahkan tipe data string untuk mendeklarasikan nama, rambut.
- (c).Mendeklarasikan constructor kelas manusia dengan menampilkan data nama, dan rambut.
- (d).Menambahkan system.out.print untuk mendeklarasikan nama dan tambahkan +nama untuk memanggil data nama dan warna rambut.
- (e).Menambahkan void hobi untuk menampilkan hobi dan tambahkan + wattpad untuk memanggil variable hobi.
- (f).Menambahkan static void main untuk menampilkan data nama, dan warna rambut.
- (g).Menambahkan hobi suka membaca wattpad.

#### 2)Kode program dan luaran



The screenshot shows a Java IDE with a code editor on the left and an output window on the right. The code defines a class 'Manusia' with attributes 'nama' and 'rambut', a constructor, and methods 'salam', 'hobi', and 'main'. The output window displays the results of running the program, showing the class name, attributes, and the output of the methods.

```
1 public class Manusia {
2     //atribut/variabel manusia
3     String nama, rambut;
4
5     //konstruktor
6     public Manusia(String nama, String rambut) {
7         System.out.println("Nama saya : " + nama +
8             "\nRambut saya : " + rambut);
9     }
10
11     //metode salam
12     void salam(String wattpad) {
13         System.out.println("Hai Manusia : " + wattpad);
14     }
15
16     //metode hobi
17     public static void main(String[] args) {
18         Manusia saya = new Manusia("RIZKIYATI", "Hitam");
19         saya.salam("wattpad");
20     }
21 }
```

Output:

```
Manusia
Nama saya : RIZKIYATI
Rambut saya : Hitam
Hai Manusia : wattpad
--- End Execution Successful ---
```

a)Screenshot/ Capture potongan kode dan hasil luaran  
Berikan komentar pada kode yang di Screenshot

b)Analisa luaran yang dihasilkan

Luaran yang dihasilkan sudah sesuai dengan permintaan soal yang program yang dijalankan sudah benar.

### [No.3] Kesimpulan

Analisa

Pada program diatas membuat public clas untuk menampilkan tipe data string nama, dan rambut. Membuat public class manusia string nama dan string nama rambut. Membuat void suka membaca dan menambahkan system.out.println untuk menampilkan hobi membaca dan tambahkan + wattpad untuk memanggil data variable hobi. Membuat objek manusia satu untuk menampilkan data nama, warna rambut dan hobi.

## LATIHAN 4

### [No. 4] Identifikasi Masalah:

Uraikan permasalahan dan variable

Contoh 4: Salin dan tempel kode program berikut ke JDoodle. Kemudian catat waktu eksekusinya.

```
public class Ortu { // membuat kelas induk
    void sukaMenonton(String a) { // method induk spesifik
        System.out.println("Nonton " + a);
    }
    void sukaMembaca(String a) { // method induk umum bisa diubah anak
        System.out.println("Suka Baca " + a);
    }
}

public static void main(String [] args) {
    System.out.println("Sifat Orang Tua :");
    Ortu objek0 = new Ortu(); // memanggil objek induk
    objek0.sukaMenonton("Berita"); // memanggil sifat spesifik induk
    objek0.sukaMembaca("Koran"); // memanggil method dengan variabel dapat diubah

    System.out.println("\n Sifat Anak :");
    Anak objekA = new Anak(); //memanggil objek anak
    objekA.sukaMenonton(9, "Film Drakor"); //memanggil sifat spesifik anak yang
    diturunkan induk
    objekA.sukaMembaca("Komik One Piece"); //memanggil method ke induk yang otomatis
    diturunkan tanpa deklarasi ulang di anak
}

class Anak extends Ortu {
    void sukaMenonton(int a, String b) {
        System.out.println("Nonton Jam " + a + " Malam " + b);
    }
    void sukaMenonton(String a) { // method induk spesifik
        System.out.println("Nonton " + a);
    }
    void sukaMembaca(String a) { // method induk umum bisa diubah anak
        System.out.println("Suka Baca " + a);
    }
}

public static void main(String [] args) {
    System.out.println("Sifat Orang Tua :");
    Ortu objek0 = new Ortu(); // memanggil objek induk
    objek0.sukaMenonton("Berita"); // memanggil sifat spesifik induk
    objek0.sukaMembaca("Koran"); // memanggil method dengan variabel dapat diubah

    System.out.println("\n Sifat Anak :");
    Anak objekA = new Anak(); //memanggil objek anak
    objekA.sukaMenonton(9, "Film Drakor"); //memanggil sifat spesifik anak yang
    diturunkan induk
    objekA.sukaMembaca("Komik One Piece"); //memanggil method ke induk yang otomatis
    diturunkan tanpa deklarasi ulang di anak
}
```

```
}
}
```

Luaran 4:  
Sifat Orang Tua :  
Nonton Berita  
Suka Baca Koran

Sifat Anak :  
Nonton Jam 9 Malam Film Drakor  
Suka Baca Komik One Piece

#### Latihan 4:

- 4.1. Bandingkan method yang dimiliki `class Anak extends Ortu` dengan method di `class Ortu`!
- 4.2. Ubahlah Contoh 4 dengan menambahkan objek anak dengan method yang berbeda!

#### [No.4] Analisis dan Argumentasi

Method class anak extends dengan method class ortu memiliki perbedaan yang mana pada method class ortu sifat ortunya suka menonton berita dan suka membaca Koran. Sedangkan sifat anaknya yaitu nonton drakor jam 9 malam, suka baca wattpad dan suka banget baca wattpad.

#### [No.4] Penyusunan Algoritma dan Kode Program

##### 1) Algoritma

- (a).Membuat program class ortu dengan menggunakan tipe data string untuk variable suka menonton, dan system.out.println kata nonton untuk menampilkan variable nontonnya.
- (b).Menambahkan void suka membaca dengan menggunakan tipe data string, dan system.out.println untuk menampilkan variable suka baca.
- (c).Membuat public class static untuk menampilkan isi dari hobi diatas.
- (d).Tambahkan system.out.println sifat ortu untuk menampilkan data dari sifat ortu tersebut.
- (e).Tambahkan system.out.println sifat anak untuk menampilkan data dari sifat anak.

##### 2)Kode program dan luaran

```

1 // class Ortu
2 void nontonBerita() {
3     System.out.println("Nonton " + hobi);
4 }
5
6 void sukaMembaca() {
7     System.out.println("Suka Baca " + hobi);
8 }
9
10 // class Anak extends Ortu
11 void nontonFilm() {
12     System.out.println("Nonton jam " + hobi + " Malam");
13 }
14
15 // Main
16 void main() {
17     Ortu ortu = new Ortu();
18     ortu.nontonBerita();
19     System.out.println("Suka banget baca " + hobi);
20 }
21
22 // class Main
23 public static void main(String[] args) {
24     Ortu ortu = new Ortu();
25     ortu.nontonBerita();
26     ortu.sukaMembaca();
27
28     Anak anak = new Anak();
29     anak.nontonFilm();
30     anak.sukaMembaca();
31 }

```

Output:

```

Sifat ortu:
Nonton Berita
Suka Baca Koran

Sifat anak:
Nonton jam 9 Malam Drakor
Suka Baca Wattpad
Suka banget baca Wattpad

```

- a)Screenshot/ Capture potongan kode dan hasil luaran  
Berikan komentar pada kode yang di Screenshot

b)Analisa luaran yang dihasilkan

Luaran yang dihasilkan sudah sesuai dengan perminttan soal yang program yang dijalankan sudah benar.

#### **[No.4] Kesimpulan**

##### **Analisa**

Pada program diatas membandingkan class anak extends ortu dan class ortu, yang mana perbandingan nya adalah jika kelas anak extend ortu ia adalah sifat anak yang dari ortu tetapi berbeda dan jika class ortu itu adalah sifat ortunya. Membuat program diatas yaitu dengan menggunakan tipe data string dan tipe data int untuk jam 9 malam nya.