

Nama & NPM	Topik:	Tanggal:
<p><b>Nama : Ari Setiawan</b> <b>Npm : G1A022092</b></p> <p><b>Nama : Ahmad Radesta</b> <b>Npm : G1A022086</b></p> <p><b>Nama : Dian Ardiyanti Saputri</b> <b>Npm : G1A022084</b></p>	<b>Kelas, Objek, dan Method</b>	<b>16 September 2022</b>
<b>[No.1] Identifikasi Masalah:</b>		
<ol style="list-style-type: none"> <li>1. Apabila diketahui kelas induk adalah Mahasiswa dan Kelas anak adalah turunan dari mahasiswa maka: <ol style="list-style-type: none"> <li>(a) Analisa atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!</li> <li>(b) Evaluasi perbedaan kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!</li> <li>(c) Rekomendasi atribut, method, dan constructor yang bisa digunakan bersama kelas induk dan kelas anak!</li> <li>(d) Desain kode program Java yang berisi atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!</li> </ol> </li> <li>2. Rincikan sumber informasi yang relevan (buku/webpage) <ol style="list-style-type: none"> <li>a) Video materi 1 tentang Kelas dan Object – <a href="https://youtu.be/60ldOc8m8Es">https://youtu.be/60ldOc8m8Es</a></li> <li>b) Video materi 2 tentang Method dan Extends - <a href="https://youtu.be/6qULMlcv-eg">https://youtu.be/6qULMlcv-eg</a></li> </ol> </li> </ol>		
<b>[No.1] Analisis dan Argumentasi</b>		
<ol style="list-style-type: none"> <li>1) Uraikan rancangan solusi yang diusulkan. Kami mengusulkan untuk menggunakan nama constructor Mahasiswa yang akan diisi dengan nama atribut sebagai pelengkap variabel constructor. Pada kode program juga kami akan mengusulkan untuk menggunakan method tanpa return value. Kode program ini akan kami jadikan sebagai kelas induk untuk diwariskan kepada kelas turunannya atau kelas anak.</li> <li>2) Analisis solusi, kaitkan dengan permasalahan. Hal ini karena kami akan menyusun sebuah kode program tentang mahasiswa yang akan menghasilkan turunan siswa. Oleh karena itu, kami mengusulkan atribut, method, dan constructor yang tepat dan cocok untuk dipasangkan pada kode program yang disusun.</li> </ol>		
<b>[No.1] Penyusunan Algoritma dan Kode Program</b>		
<ol style="list-style-type: none"> <li>1) Rancang desain solusi atau algoritma <ol style="list-style-type: none"> <li>a) Package</li> <li>b) Public class</li> <li>c) String</li> <li>d) Int</li> <li>e) Char</li> <li>f) Void</li> <li>g) System.out.println</li> <li>h) Public static void</li> </ol> </li> </ol>		

- i) New
  - j) Class
  - k) Extends
- 2) Tuliskan kode program dan luaran

### Kode Program

```
1 package tgsKelompok;
2 public class Mahasiswa { //Kelas Induk
3     String nama, rambut, kulit; //Deklarasi atribut mahasiswa dalam variabel
4     int berat, tinggi;
5     char GolDar;
6
7     public Mahasiswa() { //Deklarasi constructor
8         nama = "Bjorka";
9         rambut = "pirang";
10        kulit = "putih langsung";
11        berat = 65;
12        tinggi = 170;
13        GolDar = 'o';
14    }
15    //Deklarasi Method
16    void turunan() {
17        System.out.println("===BIODATA MAHASISWA===");
18        System.out.println("Nama Mahasiswa = "+nama
19            + "\nWarna Rambut = " + rambut
20            + "\nWarna Kulit = " + kulit
21            + "\nBerat Badan = " + berat
22            + "\nTinggi badan = " + tinggi
23            + "\nGolongan Darah = " + GolDar);
24        System.out.println("===BIODATA MAHASISWA===");
25        System.out.println("\n");
26    }
27    void SukaMemancing(String Memancing) { //method induk spesifikasi
28        System.out.println("Mahasiswa Suka " + Memancing);
29    }
30    }
31    void SukaTraveling(String a) { //Method induk umum bisa di ubah anak
32        System.out.println("Suka jalan-jalan " + a);
33    }
34    }
```

```
30    }
31    void SukaTraveling(String a) { //Method induk umum bisa di ubah anak
32        System.out.println("Suka jalan-jalan " + a);
33    }
34    void kepribadian(String sifat) { //Method induk umum bisa di ubah anak
35        System.out.println(" " + sifat);
36    }
37    public static void main(String [] args) {
38        System.out.println("===Tugas praktikum 2===");
39        Mahasiswa objek0 = new Mahasiswa(); //Memanggil objek induk
40        objek0.turunan ();
41        System.out.println("Sifat Mahasiswa = ");
42        objek0.SukaMemancing("Memancing"); //Memanggil Method dengan variabel dapat di ubah
43        objek0.SukaTraveling ("ke Eropa"); //Memanggil Method dengan variabel dapat di ubah
44        objek0.kepribadian("introvert"); //Memanggil Method dengan variabel dapat di ubah
45
46        System.out.println("\n sifat Siswa : ");
47        Siswa objekA = new Siswa(); //Memanggil Sifat spesifikasi anak yang diturunkan dari induk
48        objekA.Kesukaan("Main Game ",1,"jam"); //Memanggil method induk yang diturunkan tanpa mendeklarasikan lagi
49        objekA.SukaTraveling ("ke Jepang"); //
50        objekA.kepribadian("ekstrovert");
51
52    }
53    class Siswa extends Mahasiswa{ //Membuat kelas siswa
54        void Kesukaan(String b,int jam,String x) { // Method Kelas siswa
55            System.out.println("Siswa Suka memancing " + b + jam + x);
56        }
57    }
58    }
59
60
61
```

### Luaran

```

<terminated> Mahasiswa [Java Application] C:\Users\Asus VivoBook\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.4.v20220805-1047\jre\bin\javaw.exe (Sep 16, 2022, 1:12:45 PM - 1:12:45 PM) [pid: 7596]

====Tugas praktikum 2====
====BIODATA MAHASISWA====
Nama Mahasiswa = Bjorka
Warna Rambut = pirang
Warna Kulit = putih langsung
Berat Badan = 65
Tinggi badan = 170
Golongan Darah = o
====BIODATA MAHASISWA====

Sifat Mahasiswa =
Mahasiswa Suka Memancing
Suka jalan-jalan ke Eropa
introvert

sifat Siswa : |
Siswa Suka memancing Main Game Ijam
Suka jalan-jalan ke Jepang
ekatrovert

```

### Analisa Luaran yang Dihasilkan

Luaran yang dihasilkan sudah sesuai dengan kode program yang disusun. Kode program dapat berjalan dengan baik, yang mengartikan kode program sudah sesuai dengan ketentuan yang ada dengan menggunakan atribut, method, dan constructor yang benar.

### Atribut, Method, dan Constructor yang Digunakan

Atribut yang digunakan terdiri dari nama, rambut, kulit, berat, tinggi, dan golongan darah. Method yang digunakan adalah method tanpa return value dengan nama method yang digunakan, yaitu turunan, sukaMemancing, kesukaan, sukaTraveling, dan kepribadian. Constructor yang digunakan adalah constrctor dengan nama Mahasiswa.

### Perbedaan Kelas Induk Mahasiswa dan Kelas Anak (Turunan)

Kelas induk adalah wadah atau tempat penyimpanan data pewarisan yang akan diturunkan pada anak atau turunannya nanti. Sedangkan, kelas anak adalah kelas yang mewarisi sifat dari kelas induk atau kelas utama.

### Rekomendasi Atribut, Method, dan Constructor yang Dapat Dipasangkan Dengan Kelas Induk dan Kelas Anak

Pada kode program ini kami merekomendasikan beberapa algoritma yang dapat dipasangkan dengan kelas induk dan kelas anak, yaitu public class, string, int, char, public, void, system.out.println, public static void, new, class, dan extends.

### [No.1] Kesimpulan

Pada permasalahan ini kami menyusun sebuah kode program tentang pewarisan atau inheritance dengan kelas induk dan kelas anak. Kelas induk yang digunakan adalah Mahasiswa dan kelas anak yang digunakan adalah Siswa. Mahasiswa merupakan nama kelas dan constructor utamanya, sedangkan Siswa merupakan turunan yang mewarisi sifat dari Mahasiswa. Kode program ini menggunakan jenis atribut, method, dan constructor yang beragam, agar mendapatkan sebuah kode program dengan pewarisan yang ditandai dengan kata kunci extends. Kelas induk pada pewarisan atau inheritance merupakan suatu tempat atau wadah yang digunakan untuk menyimpan data utama yang akan diwarisi oleh turunannya atau kelas anak. Sedangkan, kelas anak pada pewarisan merupakan kelas yang akan mewarisi sifat dari kelas induk atau kelas utama.

### Refleksi

Pada materi ini kelompok kami mempelajari tentang atribut, method, dan constructor. Kami menganalisa atribut, method, dan constructor, yang akan dievaluasi juga menjadi sebuah kode program dengan pewarisan sifat dari kelas induk yang diturunkan kepada kelas anak. Kami juga dapat menentukan algoritma yang dapat dipasangkan pada atribut, method, dan constructor. Kode program yang didesain juga merupakan kode program dari data kelas induk dan kelas anak yang memiliki beberapa perbedaan yang sudah kami jelaskan di atas.