

Template Lembar Kerja Individu dan Kelompok

Nama & NPM	Topik:	Tanggal:
Natasya Salsabilla (G1A022023) Weko Abbror (G1A022025) Diodo Arrahman (G1A022027) Zoni Aryantoni Albab (G1A022043)	Kelas, Objek, Method	16 September 2022

[No. 1] Identifikasi Masalah

Diketahui : Kelas Induk : Mahasiswa

 Kelas Anak : Turunan Mahasiswa

Pada soal, kami diminta untuk menganalisa atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa). Kemudian kami diminta mengevaluasi perbedaan kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa). Selanjutnya kami diminta menampilkan rekomendasi atribut, method, dan constructor yang bisa digunakan bersama kelas induk dan kelas anak. Terakhir, kami diminta membuat desain kode program Java yang berisi atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa).

[No. 1] Analisis dan Argumentasi

Pada soal ini, kami akan menyusun kode program mulai dari classs, objek beserta constructor, method serta extends nya.

[No. 1] Penyusunan Algoritma dan Kode Program**1) Algoritma**

1. Buat kelas Mahasiswa sebagai kelas induk
2. Buat method induk
3. Deklarasikan method utama
4. Deklarasikan variabel sebagai tipe objek dari kelas
5. Deklarasikan Instansiasi untuk menciptakan sebuah objek
6. Buat inisialisasi pemanggilan konstruktor dengan mengisi nilai variabel
7. Deklarasikan extends sebagai kelas anak (turunan dari Mahasiswa)
8. Deklarasikan method pada kelas anak

2) Kode Program dan Luaran

```
1 1/*Diketahui : Kelas Induk : Mahasiswa
2 2      *      Kelas Anak : Turunan Mahasiswa
3 3      */
4 package Materi3;
5 5/* Turunan Mahasiswa :
6 6      */
7
8 public class Mahasiswa { //kelas induk
9
10     void a1(String a, String b, String c) { //method induk
11         System.out.println("Mengerjakan " + a +
12                             "\nBelajar Matkul "+b+
13                             "\nOrganisasi "+c);
14     }
15
16     public static void main(String [] args) {
17
18         System.out.println("Sifat Mahasiswa :");
19         FakultasTeknik objekA = new FakultasTeknik();
20         objekA.a1("Tugas","Umum","Universitas");
21
22
23         System.out.println("\nSifat Mahasiswa Fakultas Teknik :");
24         objekA.a1("Laporan Praktikum ",2,"Teknik","Fakultas");
25     }
26 }
27 class FakultasTeknik extends Mahasiswa {
28
29     //membuat kelas anak
30     void a1(String a, int j, String b, String c) {
31         System.out.println("Mengerjakan " + a+ "jam " +j + " malam"+
32                             "\nBelajar Matkul "+b+
33                             "\nOrganisasi "+c);
34     }
35 }
36
37 }
```

Luaran :

```
<terminated> Mahasiswa.java Application) C:\Users\acer\AppData\Local\Temp\plugins\org.ecj
Sifat Mahasiswa :
Mengerjakan Tugas
Belajar Matkul Umum
Organisasi Universitas

Sifat Mahasiswa Fakultas Teknik :
Mengerjakan Laporan Praktikum jam 2 malam
Belajar Matkul Teknik
Organisasi Fakultas
```

Luaran yang dihasilkan sudah sesuai dengan kode program. Pada kode program ini kami menggunakan `void a1(String a, String b, String c)`

Method induk karena kami ingin menampilkan 3 string dengan variabel atau parameter yang berbeda. Untuk variabel a akan menampilkan Jenis Tugas, variabel b menampilkan Jenis Matkul, dan variabel menampilkan jenis organisasi yang diikuti Mahasiswa. ObjekA pada baris 20 digunakan untuk memanggil sifat spesifik induk. ObjekA pada baris 24 digunakan untuk memanggil sifat spesifik anak yang diturunkan induk. Kami menggunakan Extends sebagai pewarisan dari semua fungsi, variabel, dan method dari kelas induk. Pada kode program kami kelas anak (Mahasiswa Fakultas Teknik) mendapatkan warisan sifat dari induk (Mahasiswa).

[No. 1] Kesimpulan

- a. Analisa atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!
Pada kode program ini atribut kami adalah variabel String a, String b, String c. Method kami void a1 yang merupakan Method tanpa pengembalian nilai (return value). Sedangkan untuk constructor, kami tidak menggunakan constructor karena kami sudah dapat menampilkan luanan cukup dengan method.
- b. Evaluasi perbedaan kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!
Kelas induk Mahasiswa menampilkan hal-hal yang dikerjakan Mahasiswa secara umum, yaitu Mengerjakan Tugas, Belajar Matkul Umum, dan Organisasi Universitas. Sedangkan, untuk kelas anak (Fakultas Teknik) menampilkan hal-hal yang dikerjakan Mahasiswa Teknik yaitu Mengerjakan Laporan Praktikum jam 2 malam, Belajar Matkul Teknik, dan Organisasi Fakultas. Adapun perbedaan pada kode program, untuk kelas anak menambahkan tampilan tipe data int untuk menampilkan keterangan jam 2 malam.
- c. Rekomendasi atribut, method, dan constructor yang bisa digunakan bersama kelas induk dan kelas anak!
Atribut, method, dan constructor yang bisa digunakan bersama kelas induk dan kelas anak adalah Jenis tugas, Jenis matkul, dan Jenis organisasi.

[Refleksi]

Menurut kami materi minggu ini cukup rumit dan *advance*. Kami telah melakukan berbagai *trial and error* dan masih belum benar-benar menguasai materi minggu ini. Karena pada materi ini banyak hal yang perlu diperhatikan agar kode dapat berjalan sesuai dengan perintah. Kami sudah mencoba membuat extends dari constructor walau tidak pernah berhasil.

Berikut ini adalah salah satu percobaan kami

```
package Materi3;
public class Mahasiswa1 {

    // deklarasi Kelas Induk
    // deklarasi atribut Manusia dalam variabel

    // deklarasi constructor
    Mahasiswa1 (char no, String nama, String fakultas, char jenisKelamin, int ukt, char nilai ) {
        System.out.println(no+" "
            +nama+" "
            +fakultas+" "
            +jenisKelamin+" "
            +ukt+" "
            +nilai);
    }
    // deklarasi method utama
    public static void main( String[] args) { //Method Utama
        System.out.println("No "
            + "Nama "
            + "Fakultas "
            + "Jenis Kelamin "
            + "UKT "
            + "Nilai");

        Mahasiswa1 satu = new Mahasiswa1('1',"Weko", "Teknik", 'L', 300000,'A' );
        Mahasiswa1 dua = new Mahasiswa1('2',"Diodo", "Teknik", 'L', 300000,'A' );
        Mahasiswa1 tiga = new Mahasiswa1('3',"Weko", "Teknik", 'L', 300000,'A' );
```

```

        Mahasiswa1 empat = new Mahasiswa1('4',"Weko", "Teknik", 'L', 300000,'A' );
    }
}

class Mahasiswa2 {

    void sukaMenonton(String a) { //method induk spesifik
        System.out.println("Nonton " + a);
    }
    void sukaMembaca(String a)
    { //method induk umum bisa diubah Mahasiswa3
        System.out.println("Suka Baca " + a);
    }
    public static void main(String [] args) {
        System.out.println("Sifat Mahasiswa :");
        Mahasiswa2 objekO = new Mahasiswa2(); //memanggil objek induk
        objekO.sukaMenonton("Berita"); //memanggil sifat spesifik induk
        objekO.sukaMembaca("Koran"); //memanggil method dengan variabel dapat diubah
        System.out.println("\n Sifat Siswa :");
        Mahasiswa3 objekA = new Mahasiswa3(); //memanggil objek Mahasiswa3
        objekA.sukaMenonton(9, "Film Drakor"); //memanggil sifat spesifik Mahasiswa3 yang
diturunkan induk
        objekA.sukaMembaca("Komik One Piece"); //memanggil method induk otomatis diturunkan
ke Mahasiswa3
    }
}
class Mahasiswa3 extends Mahasiswa2 { //membuat kelas Mahasiswa3
    void sukaMenonton(int a, String b) { //method kelas Mahasiswa3
        System.out.println("Nonton Jam " + a + " Malam " + b);
    }
}

```

Kami sudah mencari berbagai referensi, salah satu jawaban nya adalah menggunakan fungsi *super* untuk memanggil konstruktor pada kelas induk. Tapi, kami juga masih belum terlalu mengerti sehingga kami cukup menggunakan method untuk meng-*extends* kelas induk. Kami juga mencoba membuat kelas baru yang dapat di-*extends* dan tetap menampilkan konstruktor, tetapi luaran dari kelas baru tidak muncul, hanya kelas induk yang muncul. Harapan kami untuk materi selanjutnya kami dapat lebih memahami dari dasar sampai alur algoritma program materi pada minggu tersebut.