

Nama & NPM	Topik:	Tanggal:
<b>M. Hafidz Ashshidiqi</b> <b>G1A022079</b> <b>Revan Avaruz Mufid</b> <b>G1A022065</b> <b>Rengga Anggarah</b> <b>G1A022069</b>	Kelas, Objek, Method,	16 September 2022
<b>[Nomor 1] Identifikasi Masalah:</b>		
<p>1. Apabila diketahui kelas induk adalah Mahasiswa dan Kelas anak adalah turunan dari mahasiswa maka:</p> <ul 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> </ul> <p>1) Rincikan sumber informasi yang relevan (buku / webpage)  Sumber informasi didapat dari :  <a href="https://youtu.be/60ldOc8m8Es">https://youtu.be/60ldOc8m8Es</a>  <a href="https://www.youtube.com/watch?v=6qULMlcV-eg">https://www.youtube.com/watch?v=6qULMlcV-eg</a></p>		
<b>[No.1] Analisis dan Argumentasi</b>		
<p>1. Analisa atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!</p> <p>Atribut, method dan constructor kelas induk mahasiswa dan anak yaitu Nama, Jenis Kelamin, Npm, Prodi, Fakultas, dan Matkul</p> <p>2. Evaluasi perbedaan kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!</p> <p>Perbedaan Kelas Induk Mahasiswa dan Kelas Anak(turunan dari mahasiswa) yaitu Matkul Favorit dari kelas induk Mahasiswa dan kelas anak(turunan dari Mahasiswa)</p> <p>3. Rekomendasi atribut, method, dan constructor yang bisa digunakan bersama kelas induk dan kelas anak!</p> <p>Atribut yang bisa digunakan kelas induk dan anak yaitu Nama, Jenis Kelamin, Npm, Prodi, Fakultas, dan Matkul</p>		
<b>[Nomor 1] Penyusunan Algoritma dan Kode Program</b>		
<p>1) Rancang desain solusi atau algoritma</p> <p>a) susunan algoritma</p> <ol style="list-style-type: none"> <li>1) package</li> <li>2) deklarasi class</li> <li>3) deklarasi constructor</li> <li>4) deklarasi method utama</li> <li>5) menampilkan hasil luaran</li> </ol>		

## 2) Kode program dan luaran

### a) Screenshot/ Capture potongan kode dan hasil luaran

**Gambar Kode 1.1**

```
public class Mahasiswa {  
  
    String nama;  
    char jk;  
    String npm;  
    String prodi;  
    String fakultas;  
  
    public void Mahasiswa (String nama, char jk, String npm, String prodi, String fakultas){  
        System.out.println("Nama Mahasiswa      : "+nama +  
            "\nJenis Kelamin      : "+ jk +  
            "\nNPM      : " + npm +  
            "\nProgram Studi      : "+ prodi +  
            "\nFakultas      : "+ fakultas + " " + "Universitas Bengkulu"+  
            "\n");  
    }  
  
    public void Mahasiswa1 (String nama, char jk, String npm, String prodi, String fakultas){  
        System.out.println(""  
            + ""+  
            "\nNama Mahasiswa      : "+nama +  
            "\nJenis Kelamin      : "+ jk +  
            "\nNPM      : " + npm +  
            "\nProgram Studi      : "+ prodi +  
            "\nFakultas      : "+ fakultas + " " + "Universitas Bengkulu");  
    }  
  
    //deklarasi method tanpa return value  
    void mataKuliah (String matkul) { //method tanpa return value, tambah parameter String film  
        System.out.println("Mata Kuliah Favorit : " + matkul);  
    }  
  
    //deklarasi method tanpa return value, overloading  
    void belajar (String hari, String tempat) {  
        System.out.print("Mata kuliah tersebut dilaksanakan di " + tempat +  
            " pada hari " + hari);  
    }  
  
    //deklarasi method utama  
    public static void main (String[] args) {  
        System.out.println("==== DATA MAHASISWA =====");  
        Mahasiswa satu = new Mahasiswa();  
        satu.Mahasiswa("Revan Averuz Mufid", 'L', "G1A022065", "Informatika", "Teknik" );  
        Mahasiswa dua = new Mahasiswa();  
        satu.Mahasiswa("Rengga Anggarah", 'L', "G1A022069", "Informatika", "Teknik" );  
        Mahasiswa tiga = new Mahasiswa();  
        tiga.Mahasiswa("Muhammad Hafidz A", 'L', "G1A022079", "Informatika", "Teknik" );  
        // memanggil method  
        satu.mataKuliah("Komputer dan Pemograman");  
        satu.belajar("Selasa", "ruang 1.05 GB V");  
  
        System.out.println("\n==== TURUNAN MAHASISWA =====");  
        Matkul objekA = new Matkul(); //memanggil objek anak  
        objekA.mataKuliah("Multimedia", "Standar kompresi Oudio", 2); /*memanggil sifat spesifik anak  
            yang diturunkan induk*/  
        objekA.belajar("Rabu", "ruang 3.07 GB V"); /*memanggil method ke induk yang otomatis  
            diturunkan tanpa deklarasi ulang di anak */  
    }  
}  
  
class Matkul extends Mahasiswa {  
    void mataKuliah(String a, String c, int b) {  
        System.out.println("Mata kuliah favorit : " + a + " saat materi " + c + " Sebanyak " + b + " SKS")  
    }  
}
```

**Gambar Kode 1.2**

```
        System.out.print("Mata kuliah tersebut dilaksanakan di " + tempat +  
            " pada hari " + hari);  
    }  
  
    //deklarasi method utama  
    public static void main (String[] args) {  
        System.out.println("==== DATA MAHASISWA =====");  
        Mahasiswa satu = new Mahasiswa();  
        satu.Mahasiswa("Revan Averuz Mufid", 'L', "G1A022065", "Informatika", "Teknik" );  
        Mahasiswa dua = new Mahasiswa();  
        satu.Mahasiswa("Rengga Anggarah", 'L', "G1A022069", "Informatika", "Teknik" );  
        Mahasiswa tiga = new Mahasiswa();  
        tiga.Mahasiswa("Muhammad Hafidz A", 'L', "G1A022079", "Informatika", "Teknik" );  
        // memanggil method  
        satu.mataKuliah("Komputer dan Pemograman");  
        satu.belajar("Selasa", "ruang 1.05 GB V");  
  
        System.out.println("\n==== TURUNAN MAHASISWA =====");  
        Matkul objekA = new Matkul(); //memanggil objek anak  
        objekA.mataKuliah("Multimedia", "Standar kompresi Oudio", 2); /*memanggil sifat spesifik anak  
            yang diturunkan induk*/  
        objekA.belajar("Rabu", "ruang 3.07 GB V"); /*memanggil method ke induk yang otomatis  
            diturunkan tanpa deklarasi ulang di anak */  
    }  
}  
  
class Matkul extends Mahasiswa {  
    void mataKuliah(String a, String c, int b) {  
        System.out.println("Mata kuliah favorit : " + a + " saat materi " + c + " Sebanyak " + b + " SKS")  
    }  
}
```

**Gambar Luaran 1.3**

```
===== DATA MAHASISWA =====
Nama Mahasiswa      : Revan Averuz Mufid
Jenis Kelamin       : L
NPM                 : G1A022065
Program Studi       : Informatika
Fakultas            : Teknik Universitas Bengkulu

Nama Mahasiswa      : Rengga Anggarah
Jenis Kelamin       : L
NPM                 : G1A022069
Program Studi       : Informatika
Fakultas            : Teknik Universitas Bengkulu

Nama Mahasiswa      : Muhammad Hafidz A
Jenis Kelamin       : L
NPM                 : G1A022079
Program Studi       : Informatika
Fakultas            : Teknik Universitas Bengkulu

Mata Kuliah Favorit : Komputer dan Pemograman
Mata kuliah tersebut dilaksanakan di ruang 1.05 GB V pada hari Selasa
===== TURUNAN MAHASISWA =====
Mata kuliah favorit : Multimedia saat materi Standar Kompresi Oudio Sebanyak 2 SKS
Mata kuliah tersebut dilaksanakan di ruang 3.07 GB V pada hari Rabu
```

a) Analisa luaran yang dihasilkan

Luaran sudah sesuai dengan program yang disusun. Dan tidak mengalami eror dan sesuai dengan kelas Induk dan kelas anak (turunan dari induk).

#### **[Nomor 1] Kesimpulan**

1) Analisa

- a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
- b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Berdasarkan pengamatan kami program yang kami susun pada kelas Induk Mahasiswa dan Kelas Anak (turunan dari Mahasiswa) memiliki kesamaan atribut method dan constructor maka dapat disimpulkan pada kelas anak nanti akan serupa dengan kelas induk.

#### **Refleksi**

Pada materi kelas, objek, method, kami bisa memahami apa itu kelas Induk dan kelas anak (extend). Dan cara penggunaan pada program java.