

Nama & NPM	Topik:	Tanggal:
Donna Zelvia Gizella Alvera Deadora G1F024018	Tipe Data	17 September 2024

[No. 1] Identifikasi Masalah:

- 1) Uraikan permasalahan dan variabel
 1. Apabila diketahui kelas induk adalah Mahasiswa dan Kelas anak adalah turunan dari mahasiswa maka:
 - (a) Analisa atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!

Penjelasan :

Atribut :

nama
npm
prodi
sukaMembaca

Method :

void tampilkanInfo(): Menampilkan informasi mahasiswa.

Constructor :

Mahasiswa(String nama, String npm, String prodi, String sukaMembaca). Constructor digunakan untuk menginisialisasi atribut kelas mahasiswa saat objek dibuat.

 - (b) Evaluasi perbedaan kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!
- Penjelasan :
- Kelas induk memiliki atribut, constructor dan method dasar yang berlaku untuk semua mahasiswa. Kelas induk dan kelas anak memiliki atribut, constructor dan method yang sama. Kelas anak dapat menambah atribut tambahan yang sesuai dengan ketentuan yang dibutuhkan. Sebagai contoh Mahasiswa1 bisa menambah atribut khusus seperti tahunMasuk.
- (c) Rekomendasi atribut, method, dan constructor yang bisa digunakan bersama kelas induk dan kelas anak!
- Penjelasan :
- Atribut yang bisa digunakan bersama kelas induk dan anak adalah nama, npm dan prodi. Bisa juga menambahkan atribut lain seperti Angkatan, asal universitas, tinggi badan, dan lain-lain. Method nya yaitu dapat berupa sukaMembaca atau menambah yang lain seperti sukaMenonton, sukaMendaki, dan lain-lain. Constructor di kelas induk dapat digunakan untuk menginisialisasi atribut dasar, dan kelas anak hanya perlu memanggilnya.
- (d) Desain kode program Java yang berisi atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!

- 2) Rincikan sumber informasi yang

<https://www.youtube.com/watch?v=60IdOc8m8Es>

[No.1] Analisis dan Argumentasi

Kami mengusulkan pada program ini untuk memperjelas hubungan antara kelas induk dan kelas anak, serta menambahkan atribut dan metode sesuai kebutuhan. Memanfaatkan pewarisan memungkinkan kelas anak untuk mewarisi semua fungsi dari kelas induk, sehingga meminimalkan pengulangan kode.

[No.1] Penyusunan Algoritma dan Kode Program

- 1) Algoritma
 - (a) Mulai
 - (b) Membuat kelas induk
 - (c) Mendeklarasikan atribut
 - (d) Membuat constructor
 - (e) Membuat method
 - (f) Definisikan Kelas Anak Mahasiswa1 dan Mahasiswa2
 - (g) Tampilkan hasil
 - (h) Selesai
- 2) Kode program dan luaran
 - a) Screenshot/ Capture potongan kode dan hasil luaranBeri komentar pada kode yang di Screenshot

```
1 package kelaskelompok;
2
3 public class Mahasiswa {
4     void nama (String a){ //method induk spesifik
5         System.out.println("Nama: " + a);
6     }
7     void npm(String a){ //method induk umum bisa diubah anak
8         System.out.println("NPM: " + a);
9     }
10    void prodi(String a){ //method induk umum bisa diubah anak
11        System.out.println("Prodi: " + a);
12    }
13    void sukaMembaca(String a){ //method induk umum bisa diubah anak
14        System.out.println("Suka Membaca: " + a);
15    }
16    public static void main(String [] args) {
17        System.out.println();
18        Mahasiswa objek1 = new Mahasiswa (); //memanggil objek induk
19        objek1.nama("Donna Zelvia Gizella Alvera Deadora"); //memanggil sifat spesifik induk
20        objek1.npm("GIF024018"); //memanggil method dengan variabel dapat diubah
21        objek1.prodi("Sistem Informasi"); //memanggil method dengan variabel dapat diubah
22        objek1.sukaMembaca("Novel"); //memanggil method dengan variabel dapat diubah
23
24        System.out.println();
25        Mahasiswa objek2 = new Mahasiswa (); //memanggil objek induk
26        objek2.nama("Khusnul Hidayati"); //memanggil sifat spesifik induk
27        objek2.npm("GIF024032"); //memanggil method dengan variabel dapat diubah
28        objek2.prodi("Sistem Informasi"); //memanggil method dengan variabel dapat diubah
29        objek2.sukaMembaca("Komik"); //memanggil method dengan variabel dapat diubah
30
31        System.out.println();
32        Mahasiswa objek3 = new Mahasiswa (); //memanggil objek induk
33        objek3.nama("Vivilia Regita Pramesti"); //memanggil sifat spesifik induk
34        objek3.npm("GIF024060"); //memanggil method dengan variabel dapat diubah
35        objek3.prodi("Sistem Informasi"); //memanggil method dengan variabel dapat diubah
36        objek3.sukaMembaca("Novel"); //memanggil method dengan variabel dapat diubah
37    }
38
39    class Mahasiswal extends Mahasiswa { //membuat kelas anak
40        void nama (String D){ //method kelas anak
41            System.out.println("Nama : " + D);
42        }
43        void npm (String D) { //method kelas anak
44            System.out.println("NPM : " + D);
45        }
46        void prodi (String D){ //method kelas anak
47            System.out.println("prodi : " + D);
48        }
49        void sukaMembaca (String D){ //method kelas anak
50            System.out.println("Suka Membaca : " + D);
51        }
52    }
53    class Mahasiswa2 extends Mahasiswa { //membuat kelas anak
54        void nama (String E){ //method kelas anak
55            System.out.println("Nama : " + E);
56        }
57        void npm (String E) { //method kelas anak
58            System.out.println("NPM : " + E);
59        }
60        void prodi (String E) { //method kelas anak
61            System.out.println("prodi : " + E);
62        }
63        void sukaMembaca (String E){ //method kelas anak
64            System.out.println("Suka Membaca : " + E);
65        }
66    }
67 }
```

Output Console:

```
<terminated> Mahasiswa [Java Application] C:\Users\ACER\AppData\Local\Temp\pool
Nama: Donna Zelvia Gizella Alvera Deadora
NPM: GIF024018
Prodi: Sistem Informasi
Suka Membaca: Novel

Nama: Khusnul Hidayati
NPM: GIF024032
Prodi: Sistem Informasi
Suka Membaca: Komik

Nama: Vivilia Regita Pramesti
NPM: GIF024060
Prodi: Sistem Informasi
Suka Membaca: Novel
```

Kamis, 19 September 2024

b) Analisa luaran yang dihasilkan

Luaran yang dihasilkan sudah sesuai dan tepat dengan kode program yang disusun. Program menampilkan informasi berupa objek dan metod dari kelas.

[No.1] Kesimpulan

1) Analisa

Program ini bertujuan untuk menunjukkan hubungan antara kelas induk mahasiswa dan kelas anaknya, seperti Mahasiswa1 dan Mahasiswa2. Kelas induk berfungsi sebagai acuan dengan atribut dan metode dasar yang berlaku untuk semua mahasiswa. Kelas anak dapat menambah atau memodifikasi atribut dan metode sesuai kebutuhan, memungkinkan penyesuaian lebih lanjut.

Refleksi

Pada minggu ini, pengalaman belajar yang kami alami yaitu pemahaman mengenai kelas, objek, method, dan extends. Dalam menyelesaikan tugas ini, Kami mengalami banyak kesulitan karena bingung bagaimana konsep kode programnya. Tetapi, berkat usaha Kami yang terus mencoba juga dukungan dari teman-teman yang membantu mengajarkan Kami merancang program tersebut, tugas ini dapat terselesaikan.