

Template Lembar Kerja Individu dan Kelompok

Nama & NPM	Topik:	Tanggal:
Muhammad Arya Nugraha_G1F024002 Muhammad Burhanudin Sulaiman Abbas_G1F024038 Naufal Wafiq Syafri_G1F024054	Kelas, Method, dan Constructor	19 September 2024
[Nomor 1] Identifikasi Masalah:		
<p>1. Apabila diketahui kelas induk adalah Mahasiswa dan Kelas anak adalah turunan dari mahasiswa maka:</p> <p>(a) Analisa atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!</p> <p>(b) Evaluasi perbedaan kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!</p> <p>(c) Rekomendasi atribut, method, dan constructor yang bisa digunakan bersama kelas induk dan kelas anak!</p> <p>(d) Desain kode program Java yang berisi atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!</p>		
[Nomor 1] Analisis dan Argumentasi		
<p>1) Rancangan solusi melibatkan pembuatan dua kelas: Mahasiswa sebagai kelas induk yang berisi atribut umum mahasiswa dan kelas anak MahasiswaS1 yang menambahkan atribut khusus untuk mahasiswa program S1.</p> <p>2) Solusi ini mencakup penggunaan konsep inheritance untuk menyederhanakan struktur kode dan menghindari duplikasi dengan memanfaatkan kelas induk dan turunan. Ini memungkinkan pengelolaan atribut dan method yang spesifik pada setiap tipe mahasiswa dengan cara yang efisien.</p>		
[Nomor 1] Penyusunan Algoritma dan Kode Program		
<p>1) Algoritma:</p> <p>a) Mulai program.</p> <p>b) Buat kelas induk Mahasiswa:</p> <ul style="list-style-type: none"> Definisikan atribut: nama, npm, programStudi, dan tahunMasuk. Buat constructor untuk menginisialisasi atribut. Definisikan getter dan setter untuk masing-masing atribut. Buat method displayInfo() untuk menampilkan informasi atribut. <p>c) Buat kelas anak MahasiswaS1 yang merupakan turunan dari kelas Mahasiswa:</p> <ul style="list-style-type: none"> Tambahkan atribut baru: ipk dan jumlahSKS. Buat constructor yang menginisialisasi atribut dari kelas induk dan atribut tambahan. Definisikan getter dan setter untuk atribut tambahan (ipk dan jumlahSKS). Override method displayInfo() untuk menampilkan informasi dari kelas induk dan atribut tambahan. <p>d) Dalam kelas utama (Main):</p> <p>Buat objek MahasiswaS1 untuk mahasiswa:</p> <ul style="list-style-type: none"> Muhammad Arya Nugraha, dengan atribut seperti nama, npm, program 		

studi, tahun masuk, IPK, dan jumlah SKS.

- Muhammad Burhanudin Sulaiman Abbas, dengan atribut seperti nama, npm, program studi, tahun masuk, IPK, dan jumlah SKS.
- Naufal Wafiq Syafri, dengan atribut seperti nama, npm, program studi, tahun masuk, IPK, dan jumlah SKS.

e) Tampilkan informasi mahasiswa:

- Untuk setiap objek mahasiswa, panggil method `displayInfo()` untuk menampilkan data.
- Sisipkan baris kosong setelah menampilkan informasi tiap mahasiswa untuk memisahkan output.

f) Akhiri program.

2) Tuliskan kode program dan luaran

```
1 // Kelas Utama
2 class Mahasiswa {
3     private String nama;
4     private String npm;
5     private String programStudi;
6     private int tahunMasuk;
7
8     // Constructor
9     public Mahasiswa(String nama, String npm, String programStudi, int tahunMasuk) {
10         this.nama = nama;
11         this.npm = npm;
12         this.programStudi = programStudi;
13         this.tahunMasuk = tahunMasuk;
14     }
15
16     // Getter dan Setter
17     public String getNama() { return nama; }
18     public void setNama(String nama) { this.nama = nama; }
19
20     public String getNPM() { return npm; }
21     public void setNPM(String npm) { this.npm = npm; }
22
23     public String getProgramStudi() { return programStudi; }
24     public void setProgramStudi(String programStudi) { this.programStudi = programStudi; }
25
26     public int getTahunMasuk() { return tahunMasuk; }
27     public void setTahunMasuk(int tahunMasuk) { this.tahunMasuk = tahunMasuk; }
28
29     // Method untuk menampilkan data Mahasiswa
30     public void displayInfo() {
31         System.out.println("Nama: " + nama);
32         System.out.println("NPM: " + npm);
33         System.out.println("Program Studi: " + programStudi);
34         System.out.println("Tahun Masuk: " + tahunMasuk);
35     }
36 }
37
38 // Kelas Anak
39 class Mahasiswa01 extends Mahasiswa {
40     private double ipk;
41     private int jumlahSKS;
42
43     // Constructor
44     public Mahasiswa01(String nama, String npm, String programStudi, int tahunMasuk, double ipk, int jumlahSKS) {
45         super(nama, npm, programStudi, tahunMasuk);
46         this.ipk = ipk;
47         this.jumlahSKS = jumlahSKS;
48     }
49
50     // Getter dan Setter
51     public double getIPK() { return ipk; }
52     public void setIPK(double ipk) { this.ipk = ipk; }
53
54     public int getJumlahSKS() { return jumlahSKS; }
55     public void setJumlahSKS(int jumlahSKS) { this.jumlahSKS = jumlahSKS; }
56
57     // Method untuk menampilkan data Mahasiswa 01
58     @Override
59     public void displayInfo() {
60         super.displayInfo(); // Memanggil method dari kelas induk
61         System.out.println("IPK: " + ipk);
62         System.out.println("Jumlah SKS: " + jumlahSKS);
63     }
64 }
65
66 // Kelas Utama
67 public class Main {
68     public static void main(String[] args) {
69         // Membuat objek dari kelas Mahasiswa
70         Mahasiswa01 arya = new Mahasiswa01("Muhammad Arya Nugraha", "G1P024002", "Sistem Informasi", 2024, 3.95, 150);
71         Mahasiswa01 burhan = new Mahasiswa01("Muhammad Burhanudin Sulaiman Abbas", "G1P024003", "Sistem Informasi", 2024, 3.95, 150);
72         Mahasiswa01 naufal = new Mahasiswa01("Naufal Wafiq Syafri", "G1P024004", "Sistem Informasi", 2024, 3.95, 150);
73
74         // Menampilkan data
75         System.out.println("Data Mahasiswa:");
76         arya.displayInfo();
77         burhan.displayInfo();
78         naufal.displayInfo();
79     }
80 }
```

```
1 // Output:
2 Data Mahasiswa:
3 Nama: Muhammad Arya Nugraha
4 NPM: G1P024002
5 Program Studi: Sistem Informasi
6 Tahun Masuk: 2024
7 IPK: 3.95
8 Jumlah SKS: 150
9
10 Nama: Muhammad Burhanudin Sulaiman Abbas
11 NPM: G1P024003
12 Program Studi: Sistem Informasi
13 Tahun Masuk: 2024
14 IPK: 3.95
15 Jumlah SKS: 150
16
17 Nama: Naufal Wafiq Syafri
18 NPM: G1P024004
19 Program Studi: Sistem Informasi
20 Tahun Masuk: 2024
21 IPK: 3.95
22 Jumlah SKS: 150
23
24 --- Code Execution Successful ---
```

[Nomor 1] Kesimpulan

1) Analisa

Program sudah dapat Penggunaan inheritance pada kode ini mempermudah pengelolaan data mahasiswa dengan memisahkan atribut dan method umum dan spesifik. Inheritance memungkinkan penggunaan kembali kode yang sudah ada dan penambahan fitur khusus pada subclass tanpa perlu menulis ulang kode.

Refleksi

(Kami belajar tentang class, method, dan constructor)