

Template Lembar Kerja Individu dan Kelompok

Nama & NPM	Topik:	Tanggal:
1. Attiya Dianti Fadli (G1A022002) 2. Akram Analis (G1A022004) 3. Julia Mayang Sari (G1A022010)	Class, Objek, Method dan extend	15-09-2022
[No.1] Identifikasi Masalah:		
<p>1) Uraikan permasalahan dan variabel</p> <pre>//deklarsi konstruktor mahasiswa kegiatan = "Unit Kegiatan Mahasiswa"; belajar = "Mencari Sumber Informasi yang Relevan "; materi = " Sesuai Jurusan yang diambil"; pengajar = "Dosen"; //deklarasi konstruktor siswa kegiatan = "Ekstrakurikuler"; belajar = "Diberi penjelasan oleh guru di Sekolah "; materi = " Mata Pelajaran Umum "; pengajar = "Guru";</pre> <p>2) Rincikan sumber informasi yang relevan</p> <p>Sumber informasi yang saya dapat yakni dari Video pembelajaran yang dapat diakses pada Chanel Youtube Ruamh Ilmu Raflesia</p> <ul style="list-style-type: none"> • Video Materi 1 tentang Kelas, Objek, Method – https://www.youtube.com/watch?v=60ldOc8m8Es • Video Materi 2 tentang – https://www.youtube.com/watch?v=6qULMlcv-eg <p>3) Uraikan rancangan solusi yang diusulkan.</p> <p>Pada Permasalahan ini sebaiknya menggunakan uraian kode program pada contoh dibawah ini</p> <pre>public class Mahasiswa { //deklarasi atribut mahasiswa dalam variabel String kegiatan , belajar, materi, pengajar ;</pre>		



```

//deklarasi constructor
public Mahasiswa () {
    kegiatan = "Unit Kegiatan Mahasiswa";
    belajar = "Mencari Sumber Informasi yang Relevan ";
    materi = " Sesuai Jurusan yang diambil";
    pengajar = "Dosen";

    System.out.println("=====> Constructor
<=====");

    System.out.println(" Mahasiswa :"+
        "\n kegiatan yang sering di Ikuti : "+ kegiatan +
        "\n Belajar dengan cara : " + belajar +
        "\n Materi yang dipelajari : " + materi +
        "\n pengajar : " + pengajar);

    System.out.println("=====> Method
<=====");
}

//deklarasi method
void sukaBelajar(String a) // method induk spesifik
    System.out.println("Suka Belajar : " + a);
}

void sukaMembaca(String a) { // method induk umum bisa diubah siswa
    System.out.println("Suka Baca " + a);
}

//deklarasi method utama
public static void main( String[] args) {
    Mahasiswa ObjekO = new Mahasiswa();
    System.out.println("Sifat Mahasiswa :");
    ObjekO.sukaBelajar(" Bahasa Java "); // memanggil method dengan variabel dapat
    ObjekO.sukaMembaca("Buku Jurnal"); // memanggil method dengan variabel

    Siswa objekA = new Siswa(); //memanggil objek Siswa
    System.out.println("\nSifat Siswa :");

```

```

        objekA.sukaBelajar("Bahasa Inggris", 1, "jam"); //memanggil sifat spesifik anak
yang diturunkan induk

        objekA.sukaMembaca("Buku LKS"); //memanggil method ke induk yang otomatis
diturunkan tanpa deklarasi ulang di anak

```

```

class Siswa extends Mahasiswa {

    // deklarasi method

    void sukaBelajar(String a,int b, String c) { // method induk umum bisa diubah
siswa

        System.out.println("Suka Belajar " + a + " selama " + b + c);

    }

    void sukaMembaca(String a) { // method induk umum bisa diubah siswa

        System.out.println("Suka Membaca" + a + " Untuk Memahami materi");

    }
}

```

kode program yang dibuat diatas akan menampilkan hasil luaran sebagai berikut:

=====> Constructor <=====

Mahasiswa :

kegiatan yang sering di Ikuti : Unit Kegiatan Mahasiswa

Belajar dengan cara : Mencari Kegiatan Mahasiswa

Materi yang dipelajari : Sesuai Jurusan yang diambil

pengajar : Dosen

=====> Method <=====

Sifat Mahasiswa :

Suka Belajar : Bahasa Java

Suka Baca Buku Jurnal

=====> Constructor <=====

Siswa :

kegiatan yang sering di Ikuti : Ekstrakurikuler

Belajar dengan cara : Diberi penjelasan oleh guru di Sekolah

Materi yang dipelajari : Mata Pelajaran Umum



pengajar : Guru

=====> Method <=====

Sifat Siswa :

Suka Belajar Bahasa Inggris selama 1 jam

Suka MembacaBuku LKS Untuk Memahami materi

[No. 1] Analisis dan Argumentasi

- 1) Saya mengusulkan permasalahan ini dapat diatasi dengan cara Menggunakan Class Konstruktor dan Method yang menurunkan Mahasiswa ke Siswa.
- 2) Alasan solusi ini karena Class adalah entitas yang menggambarkan keadaan dan perilaku dari objek. Sehingga, kelas memiliki kumpulan objek dengan sifat yang umum. Class akan menjalankan data baru sesuai model dari objek yang dibuat berdasarkan karakter objek tersebut. Dan Method adalah Method adalah perilaku yang dilakukan objek.

Deklarasi method: `void/tipeData NamaMethod (tipeData namaVariabel) { ...`

[No.1] Penyusunan Algoritma dan Kode Program

1) Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.

- (a) Buka Eclipse/jdoodle pada komputer atau laptop.
- (b) Kemudian buat package untuk membuat kode program
- (c) Buatlah Kode program yang telah dirancang dan solusi nya:

```
public class Mahasiswa {
```

```
    //deklarasi atribut mahasiswa dalam variabel
```

```
    String kegiatan , belajar, materi, pengajar ;
```

```
    //deklarasi constructor
```

```
    public Mahasiswa () {
```

```
        kegiatan = "Unit Kegiatan Mahasiswa";
```

```
        belajar = "Mencari Sumber Informasi yang Relevan ";
```

```
        materi = " Sesuai Jurusan yang diambil";
```

```
        pengajar = "Dosen";
```

```
        System.out.println("=====> Constructor  
<=====");
```

```
        System.out.println(" Mahasiswa :"+
```

```
            "\n kegiatan yang sering di Ikuti : "+ kegiatan +
```



```

        "\n Belajar dengan cara : " + belajar +
        "\n Materi yang dipelajari : " + materi +
        "\n pengajar : " + pengajar);

        System.out.println("=====> Method
<=====");

    }

    //deklarasi method

    void sukaBelajar(String a) // method induk spesifik

        System.out.println("Suka Belajar : " + a);

    }

    void sukaMembaca(String a) { // method induk umum bisa diubah siswa

        System.out.println("Suka Baca " + a);

    }

    //deklarasi method utama

    public static void main( String[] args) {

        Mahasiswa ObjekO = new Mahasiswa();

        System.out.println("Sifat Mahasiswa :");

        ObjekO.sukaBelajar(" Bahasa Java "); // memanggil method dengan variabel dapat
diubah

        ObjekO.sukaMembaca("Buku Jurnal"); // memanggil method dengan variabel
dapat diubah


        Siswa objekA = new Siswa(); //memanggil objek Siswa

        System.out.println("\nSifat Siswa :");

        objekA.sukaBelajar("Bahasa Inggris", 1, " jam"); //memanggil sifat spesifik anak
yang diturunkan induk

        objekA.sukaMembaca("Buku LKS"); //memanggil method ke induk yang otomatis
diturunkan tanpa deklarasi ulang di anak


    }

}

class Siswa extends Mahasiswa {

    // deklarasi method

    void sukaBelajar(String a,int b, String c) { // method induk umum bisa diubah

```

siswa

```
System.out.println("Suka Belajar " + a + " selama "+ b + c);
```

```
void sukaMembaca(String a) { // method induk umum bisa diubah siswa
```

```
System.out.println("Suka Membaca" + a + " Untuk Memahami materi");
```

```
}}
```

- d) Jika kode program sudah selesai semua, run kan program tersebut
Hasil luaran sesuai dengan program yang telah disusun.

Hasil luaran:

=====> Constructor <=====

Mahasiswa :

kegiatan yang sering di Ikuti : Unit Kegiatan Mahasiswa

Belajar dengan cara : Mencari Kegiatan Mahasiswa

Materi yang dipelajari : Sesuai Jurusan yang diambil

pengajar : Dosen

=====> Method <=====

Sifat Mahasiswa :

Suka Belajar : Bahasa Java

Suka Baca Buku Jurnal

=====> Constructor <=====

Siswa :

kegiatan yang sering di Ikuti : Ekstrakurikuler

Belajar dengan cara : Diberi penjelasan oleh guru di Sekolah

Materi yang dipelajari : Mata Pelajaran Umum

pengajar : Guru

=====> Method <=====

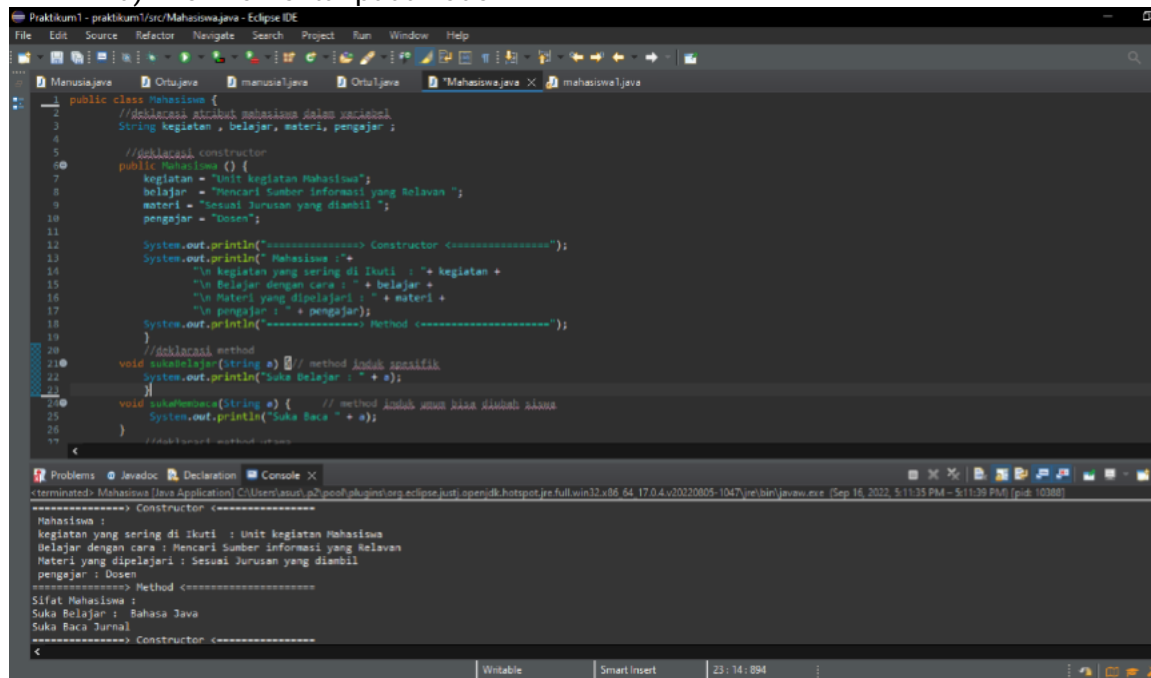
Sifat Siswa :

Suka Belajar Bahasa Inggris selama 1 jam

Suka MembacaBuku LKS Untuk Memahami materi



- 1) Tuliskan kode program dan luaran
- a) Beri komentar pada kode



```
public class Mahasiswa {
    //deklarasikan atribut mahasiswa dalam variabel
    String kegiatan, belajar, materi, pengajar;

    //deklarasikan constructor
    public Mahasiswa() {
        kegiatan = "Unit kegiatan Mahasiswa";
        belajar = "Mencari Sumber Informasi yang Relevan";
        materi = "Sesuai Jurusan yang diambil";
        pengajar = "Dosen";

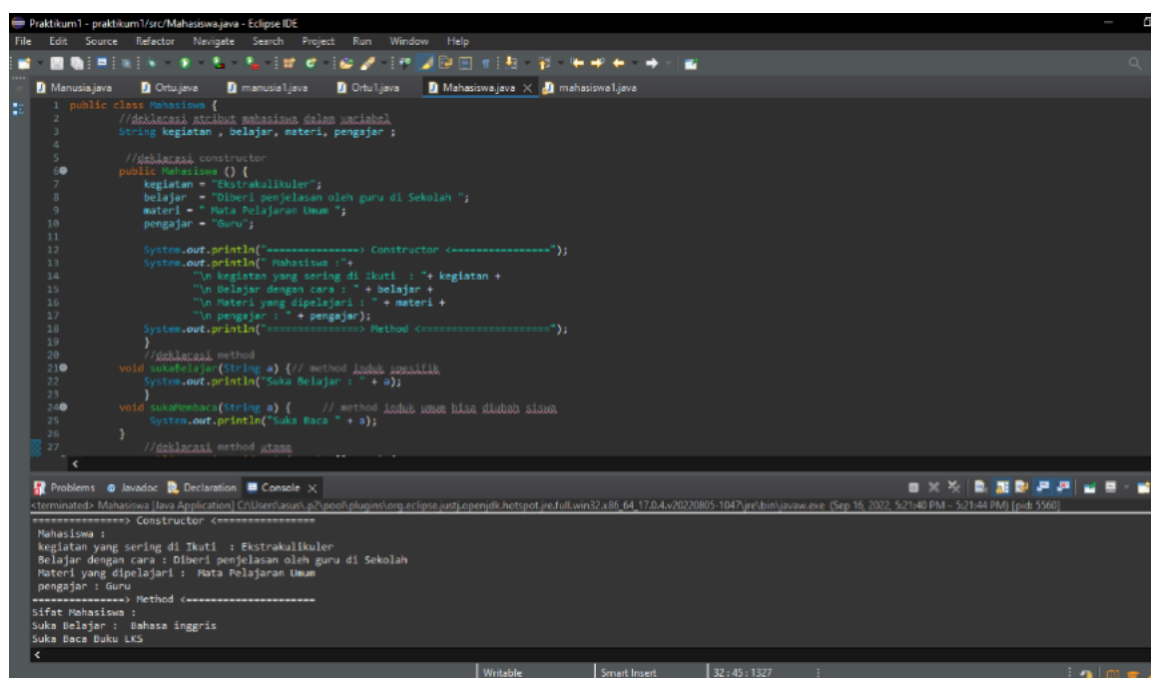
        System.out.println("===== Constructor =====");
        System.out.println("Mahasiswa : "+
            "\n kegiatan yang sering di ikuti : " + kegiatan +
            "\n Belajar dengan cara : " + belajar +
            "\n Materi yang dipelajari : " + materi +
            "\n pengajar : " + pengajar);
        System.out.println("===== Method =====");
    }

    //deklarasikan method
    void sukaBelajar(String a) { // method tidak spesifik
        System.out.println("Suka Belajar : " + a);
    }

    void sukaMembaca(String a) { // method tidak umum bisa diubah siswa
        System.out.println("Suka Baca " + a);
    }
}
```

Output Console:

```
===== Constructor =====
Mahasiswa :
kegiatan yang sering di ikuti : Unit kegiatan Mahasiswa
Belajar dengan cara : Mencari Sumber Informasi yang Relevan
Materi yang dipelajari : Sesuai Jurusan yang diambil
pengajar : Dosen
===== Method =====
Sifat Mahasiswa :
Suka Belajar : Bahasa Java
Suka Baca Jurnal
===== Constructor =====
```



```
public class Mahasiswa {
    //deklarasikan atribut mahasiswa dalam variabel
    String kegiatan, belajar, materi, pengajar;

    //deklarasikan constructor
    public Mahasiswa() {
        kegiatan = "Ekstrakurikuler";
        belajar = "Diberi penjelasan oleh guru di Sekolah";
        materi = "Mata Pelajaran Umum";
        pengajar = "Guru";

        System.out.println("===== Constructor =====");
        System.out.println("Mahasiswa : "+
            "\n kegiatan yang sering di ikuti : " + kegiatan +
            "\n Belajar dengan cara : " + belajar +
            "\n Materi yang dipelajari : " + materi +
            "\n pengajar : " + pengajar);
        System.out.println("===== Method =====");
    }

    //deklarasikan method
    void sukaBelajar(String a) { // method tidak spesifik
        System.out.println("Suka Belajar : " + a);
    }

    void sukaMembaca(String a) { // method tidak umum bisa diubah siswa
        System.out.println("Suka Baca " + a);
    }

    //deklarasikan method umum
}

//aplikasi method umum
```

Output Console:

```
===== Constructor =====
Mahasiswa :
kegiatan yang sering di ikuti : Ekstrakurikuler
Belajar dengan cara : Diberi penjelasan oleh guru di Sekolah
Materi yang dipelajari : Mata Pelajaran Umum
pengajar : Guru
===== Method =====
Sifat Mahasiswa :
Suka Belajar : Bahasa Inggris
Suka Baca Buku LKS
```

- b) Uraikanluaran yang dihasilkan
- Program ini dibuat menggunakan Class, Konstruktor dan Method yang menurunkan Mahasiswa ke Siswa. Dengan demikian menghasilkan luaran sebagai berikut:

=====> Constructor <=====

Mahasiswa :

kegiatan yang sering di Ikuti : Unit Kegiatan Mahasisa

Belajar dengan cara : Mencari Kegiatan Mahasiswa

Materi yang dipelajari : Sesuai Jurusan yang diambil

pengajar : Dosen

=====> Method <=====

Sifat Mahasiswa :

Suka Belajar : Bahasa Java

Suka Baca Buku Jurnal

=====> Constructor <=====

Siswa :

kegiatan yang sering di Ikuti : Ekstrakurikuler

Belajar dengan cara : Diberi penjelasan oleh guru di Sekolah

Materi yang dipelajari : Mata Pelajaran Umum

pengajar : Guru

=====> Method <=====

Sifat Siswa :

Suka Belajar Bahasa Inggris selama 1 jam

Suka MembacaBuku LKS Untuk Memahami materi

hasil luaran yang ditampilkan telah sesuai dengan kebutuhan dan permintaan kode program.

[No.1] Kesimpulan

1) Analisa

- Susunlahkesimpulanberdasarkanpermasalahan, algoritma, dan kode program!
- Apakahdasaralasanpengambilankeputusan Andauntukkasusini?

Pada Kode Program ini Saya mengusulkan permasalahan ini dapat diatasi dengan cara Menggunakan Class Konstruktor dan Method yang menurunkan Mahasiswa ke Siswa.

Alasan solusi ini karena Class adalah entitas yang menggambarkan keadaan dan perilaku dari objek. Sehingga, kelas memiliki kumpulan objek dengan sifat yang umum. Class akan menjalankan data baru sesuai model dari objek yang dibuat berdasarkan karakter objek tersebut. Dan Method adalah Method adalah perilaku yang dilakukan objek.

Deklarasi method: `void/tipeData NamaMethod (tipedata namaVariabel) { ...`

Maka hasil kode yang telah disusun sesuai rancangan dan solusi adalah:

```
public class Mahasiswa {
```

```
    //deklarasi atribut mahasiswa dalam variabel
```

```
    String kegiatan , belajar, materi, pengajar ;
```




```

//deklarasi constructor
public Mahasiswa () {
    kegiatan = "Unit Kegiatan Mahasiswa";
    belajar = "Mencari Sumber Informasi yang Relevan ";
    materi = " Sesuai Jurusan yang diambil";
    pengajar = "Dosen";

    System.out.println("=====> Constructor
<=====");

    System.out.println(" Mahasiswa :"+
        "\n kegiatan yang sering di Ikuti : "+ kegiatan +
        "\n Belajar dengan cara : " + belajar +
        "\n Materi yang dipelajari : " + materi +
        "\n pengajar : " + pengajar);

    System.out.println("=====> Method
<=====");
}

//deklarasi method
void sukaBelajar(String a) // method induk spesifik
    System.out.println("Suka Belajar : " + a);
}

void sukaMembaca(String a) { // method induk umum bisa diubah siswa
    System.out.println("Suka Baca " + a);
}

//deklarasi method utama
public static void main( String[] args) {
    Mahasiswa ObjekO = new Mahasiswa();
    System.out.println("Sifat Mahasiswa :");
    ObjekO.sukaBelajar(" Bahasa Java "); // memanggil method dengan variabel dapat
    ObjekO.sukaMembaca("Buku Jurnal"); // memanggil method dengan variabel

    Siswa objekA = new Siswa(); //memanggil objek Siswa
    System.out.println("\nSifat Siswa :");

```

```

        objekA.sukaBelajar("Bahasa Inggris", 1, "jam"); //memanggil sifat spesifik anak
yang diturunkan induk

        objekA.sukaMembaca("Buku LKS"); //memanggil method ke induk yang otomatis
diturunkan tanpa deklarasi ulang di anak

```

```

class Siswa extends Mahasiswa {

    // deklarasi method

    void sukaBelajar(String a,int b, String c) { // method induk umum bisa diubah
siswa

        System.out.println("Suka Belajar " + a + " selama "+ b + c);

    }

    void sukaMembaca(String a) { // method induk umum bisa diubah siswa

        System.out.println("Suka Membaca" + a + " Untuk Memahami materi");

    }
}

```

Dengan Hasil Luaran

=====> Constructor <=====

Mahasiswa :

kegiatan yang sering di Ikuti : Unit Kegiatan Mahasiswa

Belajar dengan cara : Mencari Kegiatan Mahasiswa

Materi yang dipelajari : Sesuai Jurusan yang diambil

pengajar : Dosen

=====> Method <=====

Sifat Mahasiswa :

Suka Belajar : Bahasa Java

Suka Baca Buku Jurnal

=====> Constructor <=====

Siswa :

kegiatan yang sering di Ikuti : Ekstrakurikuler

Belajar dengan cara : Diberi penjelasan oleh guru di Sekolah

Materi yang dipelajari : Mata Pelajaran Umum

pengajar : Guru

=====> Method <=====



Sifat Siswa :

Suka Belajar Bahasa Inggris selama 1 jam

Suka Membaca Buku LKS Untuk Memahami materi

Refleksi

Pada tugas kelompok ini kami mempelajari tentang apa itu class, method, objek dan extend dan bagaimana cara mengkonversikan nya. Pada materi ini kami belajar dari sumber dan relavan yang telah tersedia di laporan , dengan ada nya sumber tersebut kami dengan dapat memahami kode program ini.

