

PRAKTIKUM ALGORITMA DAN STRUKTUR DATA
JOBSHEET PERTEMUAN KE-3



NAMA : NABEEL NIZAM

KELAS : 1A

NO. ABSEN : 08

NIM : 2341720169

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2024

PERCOBAAN 1

Soal

1. Berdasarkan uji coba 3.2, apakah class yang akan dibuat array of object harus selalu memiliki atribut dan sekaligus method?Jelaskan!
2. Apakah class PersegiPanjang memiliki konstruktor?Jika tidak, kenapa dilakukan pemanggilan konstruktor pada baris program berikut :

```
ppArray[1] = new PersegiPanjang();
```

3. Apa yang dimaksud dengan kode berikut ini:

```
PersegiPanjang[] ppArray = new PersegiPanjang[3];
```

4. Apa yang dimaksud dengan kode berikut ini:

```
ppArray[1] = new PersegiPanjang();  
ppArray[1].panjang = 80;  
ppArray[1].lebar = 40;
```

5. Mengapa class main dan juga class PersegiPanjang dipisahkan pada uji coba 3.2?

Code main :

```
public class ArrayObject {  
    public static void main(String[] args) {  
        PersegiPanjang[] ppArray = new PersegiPanjang[3];  
  
        ppArray[0] = new PersegiPanjang();  
        ppArray[0].panjang = 110;  
        ppArray[0].lebar = 30;  
  
        ppArray[1] = new PersegiPanjang();  
        ppArray[1].panjang = 80;  
        ppArray[1].lebar = 40;  
  
        ppArray[2] = new PersegiPanjang();  
        ppArray[2].panjang = 100;  
        ppArray[2].lebar = 20;  
  
        System.out.println("Persegi Panjang ke-0 , Panjang : " +  
ppArray[0].panjang + " , Lebar : " + ppArray[0].lebar );  
        System.out.println("Persegi Panjang ke-1 , Panjang : " +  
ppArray[1].panjang + " , Lebar : " + ppArray[1].lebar );  
        System.out.println("Persegi Panjang ke-2 , Panjang : " +  
ppArray[2].panjang + " , Lebar : " + ppArray[2].lebar );  
    }  
}
```

Code class PersegiPanjang :

```
public class PersegiPanjang {  
    public int panjang;  
    public int lebar;  
}
```

Hasil :

```
Persegi Panjang ke-0 , Panjang : 110 , Lebar : 30
Persegi Panjang ke-0 , Panjang : 80 , Lebar : 40
Persegi Panjang ke-0 , Panjang : 100 , Lebar : 20
```

Jawab

1. Tidak selalu. Pada uji coba 3.2, kita hanya mengisi atribut tanpa menambahkan method khusus di dalam class PersegiPanjang. Meskipun demikian, program tetap berjalan dengan baik dan menampilkan output yang diharapkan. Jadi, sebuah class yang akan dibuat sebagai array of object tidak selalu harus memiliki atribut dan method secara bersamaan, tergantung pada kebutuhan dan desain program yang kita buat.
2. Tidak, pemanggilan konstruktor tersebut digunakan untuk menginisialisasi object
3. Kode tersebut digunakan untuk mendeklarasi *array of object*, kode diatas akan membuat array ppArray yang dapat menampung 3 objek PersegiPanjang.
4. Untuk mengisi elemen array, kode ini digunakan, kita perlu menginstansiasi objek persegi panjang pada array dan membuat objek tersebut terlebih dahulu.
5. Uji coba 3.2 memisahkan class main dan class PersegiPanjang untuk membedakan logika pengujian (testing) dari implementasi objek PersegiPanjang. Melakukan ini memungkinkan kita untuk lebih fokus pada pengujian dan memastikan bahwa objek PersegiPanjang beroperasi dengan benar tanpa mengganggu logika lain yang mungkin ada di class main. dengan demikian , pemisahan ini membuat kita lebih mudah untuk mengelola kode program.

PERCOBAAN 2

Soal

1. Apakah array of object dapat diimplementasikan pada array 2 Dimensi?
2. Jika jawaban soal no satu iya, berikan contohnya! Jika tidak, jelaskan!
3. Jika diketahui terdapat class Persegi yang memiliki atribut sisi bertipe integer, maka kode dibawah ini akan memunculkan error saat dijalankan. Mengapa?
4. Modifikasi kode program pada praktikum 3.3 agar length array menjadi inputan dengan Scanner!
5. Apakah boleh Jika terjadi duplikasi instansiasi array of objek, misalkan saja instansiasi dilakukan pada ppArray[i] sekaligus ppArray[0]?Jelaskan !

Code :

```
import java.util.Scanner;

public class ArrayObject {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        PersegiPanjang[] ppArray = new PersegiPanjang[3];

        for (int i = 0; i < 3; i++) {
            ppArray[i] = new PersegiPanjang();
            System.out.println("Persegi Panjang ke- " + i);
            System.out.print("Masukan Panjang : " );
```

```

        ppArray[i].panjang = sc.nextInt();
        System.out.print("Masukan Lebar : " );
        ppArray[i].lebar = sc.nextInt();
    }

    for (int i = 0; i < 3; i++) {
        System.out.println("Persegi Panjang ke- " + i);
        System.out.println("Panjang : " + ppArray[i].panjang + " ,
Lebar : " + ppArray[i].lebar );
    }
}
}

```

Hasil :

```

Persegi Panjang ke- 0
Masukan Panjang : 50
Masukan Lebar : 100
Persegi Panjang ke- 1
Masukan Panjang : 200
Masukan Lebar : 30
Persegi Panjang ke- 2
Masukan Panjang : 50
Masukan Lebar : 40
Persegi Panjang ke- 0
Panjang : 50 , Lebar : 100
Persegi Panjang ke- 1
Panjang : 200 , Lebar : 30
Persegi Panjang ke- 2
Panjang : 50 , Lebar : 40

```

Jawaban

1. Array of object dapat diimplementasikan pada array 2 dimensi. Dalam hal ini, kita dapat membuat array of object yang memiliki dua dimensi (baris dan kolom) untuk menyimpan objek-objek dengan lebih terstruktur.

Berikut adalah contoh implementasi array of object pada array 2 dimensi

Code class mahasiswa :

```

public class Mahasiswa {
    private String nama;
    private String nim;

    public Mahasiswa(String nama, String nim) {
        this.nama = nama;
        this.nim = nim;
    }

    public String getNama() {
        return nama;
    }

    public String getNim() {
        return nim;
    }
}

```

Code class main :

```
package jobsheet3;
public class mainMahasiswa {
    public static void main(String[] args) {
        // Buat array 2 dimensi yang berisi objek Mahasiswa
        Mahasiswa[][] mahasiswaArray = {
            {new Mahasiswa("dimas", "23417"), new Mahasiswa("samid",
"52201")},
            {new Mahasiswa("adi", "20169"), new Mahasiswa("bayu",
"7881")}
        };

        // Akses objek dalam array 2 dimensi
        System.out.println("Nama Mahasiswa : " +
mahasiswaArray[0][0].getNama() + ", Nim Mahasiswa : " +
mahasiswaArray[0][0].getNim());
        System.out.println("Nama Mahasiswa : " +
mahasiswaArray[0][1].getNama() + ", Nim Mahasiswa : " +
mahasiswaArray[0][1].getNim());
        System.out.println("Nama Mahasiswa : " +
mahasiswaArray[1][0].getNama() + ", Nim Mahasiswa : " +
mahasiswaArray[1][0].getNim());
        System.out.println("Nama Mahasiswa : " +
mahasiswaArray[1][1].getNama() + ", Nim Mahasiswa : " +
mahasiswaArray[1][1].getNim());
    }
}
```

Hasil :

```
Nama Mahasiswa : dimas, Nim Mahasiswa : 23417
Nama Mahasiswa : samid, Nim Mahasiswa : 52201
Nama Mahasiswa : adi, Nim Mahasiswa : 20169
Nama Mahasiswa : bayu, Nim Mahasiswa : 7881
```

3.

```
Persegi[] pgArray = new persegi [100];
pgArray[5].sisi=20;
```

Pada baris pgArray[5].sisi=20;, terdapat kemungkinan error NullPointerException. Hal ini karena array pgArray belum diinisialisasi dengan objek Persegi.

4.

```
Scanner sc = new Scanner(System.in);
System.out.println("Masukkan Panjang Array: ");
int j= sc.nextInt();
PersegiPanjang[] ppArray = new PersegiPanjang[j];
```

5. Tidak boleh terjadi duplikasi instansiasi pada array of objects, karena hal ini menyebabkan akan dibuatnya dua object pada index yang sama

PERCOBAAN 3

Soal

1. Dapatkah konstruktor berjumlah lebih dalam satu kelas? Jelaskan dengan contoh!
2. Jika diketahui terdapat class Segitiga seperti berikut ini: Tambahkan konstruktor pada class Segitiga tersebut yang berisi parameter int a, int t yang masing-masing digunakan untuk mengisi atribut alas dan tinggi.
3. Tambahkan method hitungLuas() dan hitungKeliling() pada class Segitiga tersebut. Asumsi segitiga adalah segitiga siku-siku. (Hint: Anda dapat menggunakan bantuan library Math pada Java untuk mengkalkulasi sisi miring)
4. Pada fungsi main, buat array Segitiga sgArray yang berisi 4 elemen, isikan masing-masing atributnya sebagai berikut:

sgArray ke-0 alas: 10, tinggi: 4

sgArray ke-1 alas: 20, tinggi: 10

sgArray ke-2 alas: 15, tinggi: 6

sgArray ke-3 alas: 25, tinggi: 10

5. Kemudian menggunakan looping, cetak luas dan keliling dengan cara memanggil method hitungLuas() dan hitungKeliling().

Code :

```
package jobsheet3.ArrayBalok;
public class Balok {
    public int panjang;
    public int lebar;
    public int tinggi;

    public Balok(int p, int l, int t){
        panjang = p ;
        lebar = l;
        tinggi = t;
    }
    public int hitungVolume(){
        return panjang * lebar * tinggi;
    }
    public static void main(String[] args) {
        Balok[] blArray = new Balok[3];

        blArray[0] = new Balok(100, 30, 12);
        blArray[1] = new Balok(120, 40, 15);
        blArray[2] = new Balok(210, 50, 25);

        for (int i = 0; i < 3; i++) {
            System.out.println("Volume Balok ke- " + i + " : " +
            blArray[i].hitungVolume());
        }
    }
}
```

Hasil :

```
Volume Balok ke- 0 : 36000
Volume Balok ke- 1 : 72000
Volume Balok ke- 2 : 262500
```

Jawab :

1. Ya, sebuah kelas dalam bahasa pemrograman dapat memiliki lebih dari satu konstruktor.

```
public class Persegi {

    private int sisi;
    private String warna;

    public Persegi(int sisi) {
        this.sisi = sisi;
        this.warna = "Putih"; // Warna default
    }

}
```

2.

```
public class Segitiga {

    public int alas;
    public int tinggi;

    public Segitiga(int a, int t) {
        this.alas = a;
        this.tinggi = t;
    }

}
```

3.

```
public class Segitiga {
    public int alas;
    public int tinggi;

    public Segitiga(int a, int t) {
        this.alas = a;
        this.tinggi = t;
    }

    public int hitungLuas() {
        return (alas * tinggi) / 2;
    }

    public double hitungKeliling() {
        double sisiMiring = Math.sqrt(Math.pow(alas, 2) +
Math.pow(tinggi, 2));
        return alas + tinggi + sisiMiring;
    }

}
```

4.

```
public class Main {  
  
    public static void main(String[] args) {  
        Segitiga[] sgArray = new Segitiga[4];  
  
        sgArray[0] = new Segitiga(10, 4);  
        sgArray[1] = new Segitiga(20, 10);  
        sgArray[2] = new Segitiga(15, 6);  
        sgArray[3] = new Segitiga(25, 10);  
  
        for (int i = 0; i < sgArray.length; i++) {  
            System.out.println("Segitiga ke-" + (i + 1));  
            System.out.println("Alas: " + sgArray[i].alas);  
            System.out.println("Tinggi: " + sgArray[i].tinggi);  
            System.out.println("Luas: " + sgArray[i].hitungLuas());  
            System.out.println("Keliling: " +  
sgArray[i].hitungKeliling());  
            System.out.println();  
        }  
    }  
}
```

5.

```
for (Segitiga s : sgArray) {  
    System.out.println("Luas: " + s.hitungLuas());  
    System.out.println("Keliling: " + s.hitungKeliling());  
    System.out.println();  
}
```


LATIHAN PRAKTIKUM

Soal

1. Buatlah program yang dapat menghitung luas permukaan dan volume bangun ruang kerucut, limas segi empat sama sisi, dan bola. Buatlah 3 (tiga) class sesuai dengan jumlah jenis bangun ruang. Buatlah satu main class untuk membuat array of objects yang menginputkan atribut-atribut yang ada menggunakan konstruktor semua bangun ruang tersebut. Dengan ketentuan,
 - a. Buat looping untuk menginputkan masing-masing atributnya, kemudian tampilkan luas permukaan dan volume dari tiap jenis bangun ruang tersebut.
 - b. Pada kerucut, inputan untuk atribut hanya jari-jari dan sisi miring
 - c. Pada limas segi empat sama sisi, inputan untuk atribut hanya panjang sisi alas dan tinggi limas
 - d. Pada bola, inputan untuk atribut hanya jari-jari
2. Sebuah kampus membutuhkan program untuk menampilkan informasi mahasiswa berupa nama, nim, jenis kelamin dan juga IPK mahasiswa. Program dapat menerima input semua informasi tersebut, kemudian menampilkannya kembali ke user. Implementasikan program tersebut jika dimisalkan terdapat 3 data mahasiswa yang tersedia. Contoh output program:

```
Masukkan data mahasiswa ke- 1
Masukkan nama :Rina
Masukkan nim :1234567
Masukkan jenis kelamin :P
Masukkan IPK :3.5
Masukkan data mahasiswa ke- 2
Masukkan nama :Rio
Masukkan nim :7654321
Masukkan jenis kelamin :L
Masukkan IPK :4.0
Masukkan data mahasiswa ke- 3
Masukkan nama :Reza
Masukkan nim :8765398
Masukkan jenis kelamin :L
Masukkan IPK :3.8
```

```
Data Mahasiswa ke-1
nama : Rina
nim : 1234567
Jenis kelamin : P
Nilai IPK : 3.5
Data Mahasiswa ke-2
nama : Rio
nim : 7654321
Jenis kelamin : L
Nilai IPK : 4.0
Data Mahasiswa ke-3
nama : Reza
nim : 8765398
Jenis kelamin : L
Nilai IPK : 3.8
```

3. Modifikasi program Latihan no.2 di atas, sehingga bisa digunakan untuk menghitung rata-rata IPK, serta menampilkan data mahasiswa dengan IPK terbesar! (gunakan method untuk masing-masing proses tersebut)

Jawaban :

1.

Code Class Kerucut

```
package jobsheet3.Latihan1;

import java.util.Scanner;

class Kerucut {
    private double jariJari;
    private double sisiMiringKerucut;

    public Kerucut(double r, double sM) {
        jariJari = r;
        sisiMiringKerucut = sM;
    }

    public double hitungLuasPermukaan() {

        double luasSelimut = Math.PI * jariJari * sisiMiringKerucut;
        double luasAlas = Math.PI * Math.pow(jariJari, 2);
        return luasSelimut + luasAlas;
    }

    public double hitungVolume() {
        double tinggiKerucut ;
        tinggiKerucut = Math.pow(sisiMiringKerucut, 2) -
Math.pow(jariJari, 2);
        tinggiKerucut = Math.sqrt(2);
        return (1.0 / 3.0) * Math.PI * Math.pow(jariJari, 2) *
tinggiKerucut;
    }
}
```

Code Class Limas

```
package jobsheet3.Latihan1;

import java.util.Scanner;

class Kerucut {
    private double jariJari;
    private double sisiMiringKerucut;

    public Kerucut(double r, double sM) {
        jariJari = r;
        sisiMiringKerucut = sM;
    }

    public double hitungLuasPermukaan() {

        double luasSelimut = Math.PI * jariJari * sisiMiringKerucut;
        double luasAlas = Math.PI * Math.pow(jariJari, 2);
        return luasSelimut + luasAlas;
    }
}
```

```

        public double hitungVolume() {
            double tinggiKerucut ;
            tinggiKerucut = Math.pow(sisiMiringKerucut, 2) -
Math.pow(jariJari, 2);
            tinggiKerucut = Math.sqrt(2);
            return (1.0 / 3.0) * Math.PI * Math.pow(jariJari, 2) *
tinggiKerucut;
        }
    }
}

```

Code Class Bola

```

package jobsheet3.Latihan1;

class Bola {
    private double jariJari;

    public Bola(double r) {
        jariJari = r;
    }

    public double hitungLuasPermukaan() {
        return 4 * Math.PI * Math.pow(jariJari, 2);
    }

    public double hitungVolume() {
        return (4.0 / 3.0) * Math.PI * Math.pow(jariJari, 3);
    }
}

```

Code MainLatihan1

```

package jobsheet3.Latihan1;

import java.util.Scanner;

public class mainLatihan1 {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);
        int jmlBangunRuang;
        System.out.println("1. Kerucut ");
        System.out.println("2. Limas Segi Empat Sama Sisi ");
        System.out.println("3. Bola ");
        System.out.println("Bangun ruang apa yang ingin anda hitung : ");
        int bangunRuang = scanner.nextInt();

        System.out.println("Masukan Jumlah bangun ruang yang ingin anda
hitung : ");
        jmlBangunRuang = scanner.nextInt();

        if (bangunRuang == 1) {
            Kerucut[] kerucutArray = new Kerucut[jmlBangunRuang];
            for (int i = 0; i < jmlBangunRuang; i++) {
                System.out.println("Bangun Ruang Limas Segi Empat ke-" +
(i + 1));
            }
        }
    }
}

```

```

        System.out.print("Masukkan jari-jari kerucut ke- " +
(i+1) + " : ");
        double jariJariKerucut = scanner.nextDouble();
        System.out.print("Masukkan sisi miring kerucut ke- " +
(i+1) + " : ");
        double sisiMiringKerucut = scanner.nextDouble();
        kerucutArray[i] = new Kerucut(jariJariKerucut,
sisiMiringKerucut);
        System.out.println("Luas Permukaan Bangun Kerucut ke-" +
(i + 1) + ": " + kerucutArray[i].hitungLuasPermukaan());
        System.out.println("Volume Bangun Kerucut ke-" + (i + 1)
+ ": " + kerucutArray[i].hitungVolume());
    }
    } else if (bangunRuang == 2) {
        LimasSegiEmpatSamaSisi[] limasArray = new
LimasSegiEmpatSamaSisi[jmlBangunRuang];
        for (int i = 0; i < jmlBangunRuang; i++) {
            System.out.println("Bangun Ruang Limas Segi Empat ke-" +
(i + 1));
            System.out.print("Masukkan panjang sisi alas limas ke- "
+ (i+1) + " : ");
            double panjangSisiAlasLimas = scanner.nextDouble();
            System.out.print("Masukkan tinggi limas ke- " + (i+1) + "
: ");
            double tinggiLimas = scanner.nextDouble();
            limasArray[i] = new
LimasSegiEmpatSamaSisi(panjangSisiAlasLimas, tinggiLimas);
            System.out.println("Luas Permukaan Bangun Kerucut ke-" +
(i + 1) + ": " + limasArray[i].hitungLuasPermukaan());
            System.out.println("Volume Bangun Kerucut ke-" + (i + 1)
+ ": " + limasArray[i].hitungVolume());
        }
    } else if (bangunRuang == 3) {
        Bola[] bola = new Bola[jmlBangunRuang];
        for (int i = 0; i < jmlBangunRuang; i++) {
            System.out.println("Bangun Ruang Limas Segi Empat ke-" +
(i + 1));
            System.out.print("Masukkan jari-jari bola ke- " + (i+1) +
" : ");
            double jariJariBola = scanner.nextDouble();
            bola[i] = new Bola(jariJariBola);
            System.out.println("Luas Permukaan Bangun Kerucut ke-" +
(i + 1) + ": " + bola[i].hitungLuasPermukaan());
            System.out.println("Volume Bangun Kerucut ke-" + (i + 1)
+ ": " + bola[i].hitungVolume());
        }
    } else {
        System.out.println("Pilihan tidak valid");
        return;
    }
}
}

```

Hasil Run :

Kerucut :

```
1. Kerucut
2. Limas Segi Empat Sama Sisi
3. Bola
Bangun ruang apa yang ingin anda hitung :
1
Masukan Jumlah bangun ruang yang ingin anda hitung :
2
Bangun Ruang Limas Segi Empat ke-1
Masukkan jari-jari kerucut ke- 1 : 5
Masukkan sisi miring kerucut ke- 1 : 5
Luas Permukaan Bangun Kerucut ke-1: 157.07963267948966
Volume Bangun Kerucut ke-1: 37.024024484653054
Bangun Ruang Limas Segi Empat ke-2
Masukkan jari-jari kerucut ke- 2 : 10
Masukkan sisi miring kerucut ke- 2 : 10
Luas Permukaan Bangun Kerucut ke-2: 628.3185307179587
Volume Bangun Kerucut ke-2: 148.09609793861222
```

Limas :

```
1. Kerucut
2. Limas Segi Empat Sama Sisi
3. Bola
Bangun ruang apa yang ingin anda hitung :
2
Masukan Jumlah bangun ruang yang ingin anda hitung :
2
Bangun Ruang Limas Segi Empat ke-1
Masukkan panjang sisi alas limas ke- 1 : 5
Masukkan tinggi limas ke- 1 : 5
Luas Permukaan Bangun Kerucut ke-1: 75.0
Volume Bangun Kerucut ke-1: 41.666666666666666
Bangun Ruang Limas Segi Empat ke-2
Masukkan panjang sisi alas limas ke- 2 : 10
Masukkan tinggi limas ke- 2 : 10
Luas Permukaan Bangun Kerucut ke-2: 300.0
Volume Bangun Kerucut ke-2: 333.33333333333326
```

Bola :

```
1. Kerucut
2. Limas Segi Empat Sama Sisi
3. Bola
Bangun ruang apa yang ingin anda hitung :
3
Masukan Jumlah bangun ruang yang ingin anda hitung :
2
Bangun Ruang Limas Segi Empat ke-1
Masukkan jari-jari bola ke- 1 : 5
Luas Permukaan Bangun Kerucut ke-1: 314.1592653589793
Volume Bangun Kerucut ke-1: 523.5987755982989
Bangun Ruang Limas Segi Empat ke-2
Masukkan jari-jari bola ke- 2 : 5
Luas Permukaan Bangun Kerucut ke-2: 314.1592653589793
Volume Bangun Kerucut ke-2: 523.5987755982989
```

Latihan 2 :

Code Class Mahasiswa :

```
package jobsheet3.Latihan2;

class Mahasiswa {

    private String nama;
    private String nim;
    private String jenisKelamin;
    private double ipk;

    public Mahasiswa(String nama, String nim, String jenisKelamin, double ipk) {
        this.nama = nama;
        this.nim = nim;
        this.jenisKelamin = jenisKelamin;
        this.ipk = ipk;
    }

    public String Nama() {
        return nama;
    }

    public String Nim() {
        return nim;
    }

    public String JenisKelamin() {
        return jenisKelamin;
    }

    public double Ip() {
        return ipk;
    }
}
```

Code Class MainMahasiswa :

```
package jobsheet3.Latihan2;

import java.util.Scanner;

public class mainMahasiswa {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        Mahasiswa[] mahasiswa = new Mahasiswa[3];

        for (int i = 0; i < mahasiswa.length; i++) {
            System.out.println("Mahasiswa ke-" + (i + 1));
            System.out.print("Masukkan nama: ");
            String nama = scanner.next();
            System.out.print("Masukkan NIM: ");
            String nim = scanner.next();
            System.out.print("Masukkan jenis kelamin (L/P): ");
            String jenisKelamin = scanner.next();
        }
    }
}
```

```

        System.out.print("Masukkan IPK: ");
        double ipk = scanner.nextDouble();

        Mahasiswa mhs = new Mahasiswa(nama, nim, jenisKelamin,
ipk);
        mahasiswa[i] = mhs;
    }

    for (Mahasiswa mhs : mahasiswa) {
        System.out.println();
        System.out.println("Nama: " + mhs.Nama());
        System.out.println("NIM: " + mhs.Nim());
        System.out.println("Jenis Kelamin: " +
mhs.JenisKelamin());
        System.out.println("IPK: " + mhs.Ip());
    }
}

```

Hasil Run :

```

Mahasiswa ke-1
Masukkan nama: samid
Masukkan NIM: 54321
Masukkan jenis kelamin (L/P): l
Masukkan IPK: 4
Mahasiswa ke-2
Masukkan nama: dimas
Masukkan NIM: 12345
Masukkan jenis kelamin (L/P): l
Masukkan IPK: 3.7
Mahasiswa ke-3
Masukkan nama: adi
Masukkan NIM: 53263
Masukkan jenis kelamin (L/P): l
Masukkan IPK: 3

Nama: samid
NIM: 54321
Jenis Kelamin: l
IPK: 4.0

Nama: dimas
NIM: 12345
Jenis Kelamin: l
IPK: 3.7

Nama: adi
NIM: 53263
Jenis Kelamin: l
IPK: 3.0

```

Latihan Praktikum 3 :

Code Class Mahasiswa :

```
package jobsheet3.Latihan2;

class Mahasiswa {

    private String nama;
    private String nim;
    private String jenisKelamin;
    private double ipk;

    public Mahasiswa(String nama, String nim, String jenisKelamin, double ipk) {
        this.nama = nama;
        this.nim = nim;
        this.jenisKelamin = jenisKelamin;
        this.ipk = ipk;
    }

    public String Nama() {
        return nama;
    }

    public String Nim() {
        return nim;
    }

    public String JenisKelamin() {
        return jenisKelamin;
    }

    public double Ip() {
        return ipk;
    }
}
```

Code Class MainMahasiswa :

```
package jobsheet3.Latihan3;

import java.util.Scanner;

public class mainMahasiswa {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        Mahasiswa[] mahasiswa = new Mahasiswa[3];

        for (int i = 0; i < mahasiswa.length; i++) {
            System.out.println("Mahasiswa ke-" + (i + 1));
            System.out.print("Masukkan nama: ");
            String nama = scanner.next();
            System.out.print("Masukkan NIM: ");
            String nim = scanner.next();
            System.out.print("Masukkan jenis kelamin (L/P): ");
```



```

        String jenisKelamin = scanner.next();
        System.out.print("Masukkan IPK: ");
        double ipk = scanner.nextDouble();

        Mahasiswa mhs = new Mahasiswa(nama, nim, jenisKelamin, ipk);
        mahasiswa[i] = mhs;
    }

    System.out.println("\nData Mahasiswa:");
    for (Mahasiswa mhs : mahasiswa) {
        System.out.println("Nama: " + mhs>Nama());
        System.out.println("NIM: " + mhs.Nim());
        System.out.println("Jenis Kelamin: " + mhs.JenisKelamin());
        System.out.println("IPK: " + mhs.Ip());
        System.out.println();
    }

    double totalIpk = 0;
    for (Mahasiswa mhs : mahasiswa) {
        totalIpk += mhs.Ip();
    }
    double rataIpk = totalIpk / mahasiswa.length;
    System.out.println("Rata-rata IPK: " + rataIpk);

    double maxIpk = mahasiswa[0].Ip();
    Mahasiswa mahasiswaTerbaik = mahasiswa[0];
    for (int i = 1; i < mahasiswa.length; i++) {
        if (mahasiswa[i].Ip() > maxIpk) {
            maxIpk = mahasiswa[i].Ip();
            mahasiswaTerbaik = mahasiswa[i];
        }
    }
    System.out.println("\nMahasiswa dengan IPK terbesar:");
    System.out.println("Nama: " + mahasiswaTerbaik>Nama());
    System.out.println("NIM: " + mahasiswaTerbaik.Nim());
    System.out.println("Jenis Kelamin: " +
mahasiswaTerbaik.JenisKelamin());
    System.out.println("IPK: " + mahasiswaTerbaik.Ip());
    }
}

```

Hasil run Code :

```
Mahasiswa ke-1
Masukkan nama: Samid
Masukkan NIM: 12342
Masukkan jenis kelamin (L/P): L
Masukkan IPK: 4
Mahasiswa ke-2
Masukkan nama: Dimas
Masukkan NIM: 12345
Masukkan jenis kelamin (L/P): L
Masukkan IPK: 3.5
Mahasiswa ke-3
Masukkan nama: Dimsut
Masukkan NIM: 45678
Masukkan jenis kelamin (L/P): L
Masukkan IPK: 2.5

Data Mahasiswa:
Nama: Samid
NIM: 12342
Jenis Kelamin: L
IPK: 4.0

Nama: Dimas
NIM: 12345
Jenis Kelamin: L
IPK: 3.5

Nama: Dimsut
NIM: 45678
Jenis Kelamin: L
IPK: 2.5

Rata-rata IPK: 3.3333333333333335

Mahasiswa dengan IPK terbesar:
Nama: Samid
NIM: 12342
Jenis Kelamin: L
IPK: 4.0
```