Object Oriented Programming

Jobsheet 9

Overloading and Overriding

BY :

D4 INFORMATICS ENGINEERING (2I)
Shofa Yasmin Fauziah (21)

State Polytechnic of Malang

Soekarno Hatta street No.9, Malang, East Java 65141

2023/2024

```java
package percobaan1;
public class Karyawan {
    private String nama;
    private String nip;
    private String golongan;
    private double gaji;

    public double getGaji() {
        return gaji;
    }

    public void setNama(String nama) {
        this.nama = nama;
    }

    public void setNip(String nip) {
        this.nip = nip;
    }

    public void setGolongan(String golongan) {
        this.golongan = golongan;

        switch (golongan.charAt(0)){
            case '1':this.gaji=5000000;
                break;
            case '2':this.gaji=3000000;
                break;
            case '3':this.gaji=2000000;
                break;
            case '4':this.gaji=1000000;
                break;
            case '5':this.gaji=7500000;
                break;
        }
    }

    public void setGaji(double gaji){
        this.gaji = gaji;
    }

    public String getNama() {
        return nama;
    }

    public String getNip() {
        return nip;
    }

    public String getGolongan() {
        return golongan;
    }
}
```

```java
package percobaan1;
public class Manager extends Karyawan{
    private double tunjangan;
    private String bagian;
    private Staff st[];

    public double getTunjangan() {
        return tunjangan;
    }

    public void setTunjangan(double tunjangan) {
        this.tunjangan = tunjangan;
    }

    public String getBagian() {
        return bagian;
    }

    public void setBagian(String bagian) {
        this.bagian = bagian;
    }

    public Staff[] getSt() {
        return st;
    }

    public void setSt(Staff[] st) {
        this.st = st;
    }

    public void viewStaff(){
        int i;
        System.out.println("-----------------");
        for(i=0; i<st.length;i++){
            st[i].lihatInfo();
        }
        System.out.println("-----------------");
    }

    public void lihatInfo(){
        System.out.println("NIP :"+this.getNip());
        System.out.println("Nama :"+this.getNama());
        System.out.println("Golongan
:"+this.getGolongan());
        System.out.printf("Tunjangan :%.0f\n",
this.getTunjangan());
        System.out.printf("Gaji :%.0f\n", this.getGaji());
        System.out.println("Bagian :"+this.getBagian());
    }

    public double getGaji(){
        return super.getGaji()+tunjangan;
    }
}
```

```java
package percobaan1;
public class Staff extends Karyawan{
    private int lembur;
    private double gajiLembur;

    public void setLembur(int lembur) {
        this.lembur = lembur;
    }

    public void setGajiLembur(double gajiLembur) {
        this.gajiLembur = gajiLembur;
    }

    public int getLembur() {
        return lembur;
    }

    public double getGajiLembur() {
        return gajiLembur;
    }

    //overloding
    public double getGaji(int lembur, double gajiLembur){
        return super.getGaji()+lembur*gajiLembur;
    }

    @Override
    public double getGaji(){
        return super.getGaji()+lembur*gajiLembur;
    }

    public void lihatInfo(){
        System.out.println("NIP :"+this.getNip());
        System.out.println("Nama :"+this.getNama());
        System.out.println("Golongan
:"+this.getGolongan());
        System.out.println("Jml Lembur
:"+this.getLembur());
        System.out.printf("Gaji Lembur:%.0f\n",
this.getGajiLembur());
        System.out.printf("Gaji :%.0f\n", this.getGaji());
    }
}
```

```
package percobaanl;
public class Utama {
    public static void main(String[] args) {
        System.out.println( x: "Program Testing Class Manager & Staff");
        Manager man[] = new Manager[2];
        Staff staff1[] = new Staff[2];
        Staff staff2[] = new Staff[3];


        //pembuatan manager

        man[0]=new Manager();
        man[0].setNama( nama: "Tedjo");
        man[0].setNip( nip: "101");
        man[0].setGolongan( golongan: "1");
        man[0].setTunjangan( tunjangan: 5000000);
        man[0].setBagian( bagian: "Administrasi");

        man[1]=new Manager();
        man[1].setNama( nama: "Atika");
        man[1].setNip( nip: "102");
        man[1].setGolongan( golongan: "1");
        man[1].setTunjangan( tunjangan: 2500000);
        man[1].setBagian( bagian: "Pemasaran");

        staff1[0]=new Staff();
        staff1[0].setNama( nama: "Usman");
        staff1[0].setNip( nip: "0003");
        staff1[0].setGolongan( golongan: "2");
        staff1[0].setLembur( lembur: 10);
        staff1[0].setGajiLembur( gajiLembur: 10000);

        staff1[1]=new Staff();
        staff1[1].setNama( nama: "Anugrah");
        staff1[1].setNip( nip: "0005");
        staff1[1].setGolongan( golongan: "2");
        staff1[1].setLembur( lembur: 10);
        staff1[1].setGajiLembur( gajiLembur: 55000);
        man[0].setSt( st: staff1);

        staff2[0]=new Staff();
        staff2[0].setNama( nama: "Hendra");
        staff2[0].setNip( nip: "0004");
        staff2[0].setGolongan( golongan: "3");
        staff2[0].setLembur( lembur: 15);
        staff2[0].setGajiLembur( gajiLembur: 5500);

```

```
47
48              staff2[1]=new Staff();
49              staff2[1].setNama( nama: "Arie");
50              staff2[1].setNip( nip: "0006");
51              staff2[1].setGolongan( golongan: "4");
52              staff2[1].setLembur( lembur: 5);
53              staff2[1].setGajiLembur( gajiLembur: 100000);
54
55              staff2[2]=new Staff();
56              staff2[2].setNama( nama: "Mentari");
57              staff2[2].setNip( nip: "0007");
58              staff2[2].setGolongan( golongan: "3");
59              staff2[2].setLembur( lembur: 6);
60              staff2[2].setGajiLembur( gajiLembur: 20000);
61              man[1].setSt( st: staff2);
62
63              //cetak informasi dari manager + staffnya
64              man[0].lihatInfo();
65              man[1].lihatInfo();
66          }
67      }
```

```
Program Testing Class Manager & Staff
NIP :101
Nama :Tedjo
Golongan :1
Tunjangan :5000000
Gaji :10000000
Bagian :Administrasi
NIP :102
Nama :Atika
Golongan :1
Tunjangan :2500000
Gaji :7500000
Bagian :Pemasaran
```

Latihan

```
public class PerkalianKu {

 void perkalian(int a, int b){

  System.out.println(a * b);

 }

 void perkalian(int a, int b, int c){

  System.out.println(a * b * c);

 }

 public static void main(String args []){

  PerkalianKu objek = new PerkalianKu();

  objek.perkalian(25, 43);
  objek.perkalian(34, 23, 56);
 }
}
```

4.1 Dari source coding diatas terletak dimanakah overloading?

```
void perkalian(int a, int b, int c){

 System.out.println(a * b * c);

}
```

4.2 Jika terdapat overloading ada berapa jumlah parameter yang berbeda
Satu parameter yaitu parameter c dengan data type int

```
public class PerkalianKu {

 void perkalian(int a, int b){

  System.out.println(a * b);

 }

 void perkalian(double a, double b){

  System.out.println(a * b);

 }

 public static void main(String args []){

  PerkalianKu objek = new PerkalianKu();

  objek.perkalian(25, 43);
  objek.perkalian(34.56, 23.7);
 }
}
```

4.3 Dari source coding diatas terletak dimanakah overloading?

```
void perkalian(double a, double b){

 System.out.println(a * b);

}
```

4.4 Jika terdapat overloading ada berapa tipe parameter yang berbeda?
Terdapat 2 parameter yang memiliki data type berbeda, yaitu a dan b yang memiliki data type int dan double

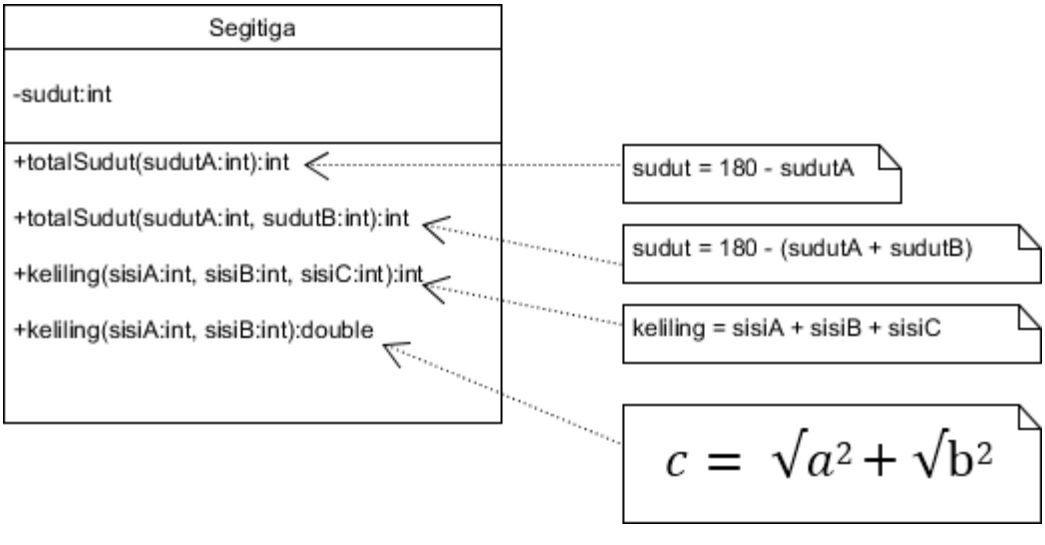
```
class Ikan{
 public void swim(){
    System.out.println("Ikan bisa berenang");
 }
}
class Piranha extends Ikan{
 public void swim(){
    System.out.println("Piranha bisa makan daging");
 }
}
public class Fish {
    public static void main(String[] args) {
       Ikan a = new Ikan();
      Ikan b = new Piranha();
      a.swim();
      b.swim();
    }
 }
```

4.5 Dari source coding diatas terletak dimanakah overriding?

```
class Piranha extends Ikan{
  public void swim(){
    System.out.println("Piranha bisa makan daging");
  }
}
```

4.6 Jabarkanlah apabila sourcoding diatas jika terdapat overriding?
Pada class Piranha terdapat method swim, yang merupakan merupakan overriding dari method swim dari class Ikan

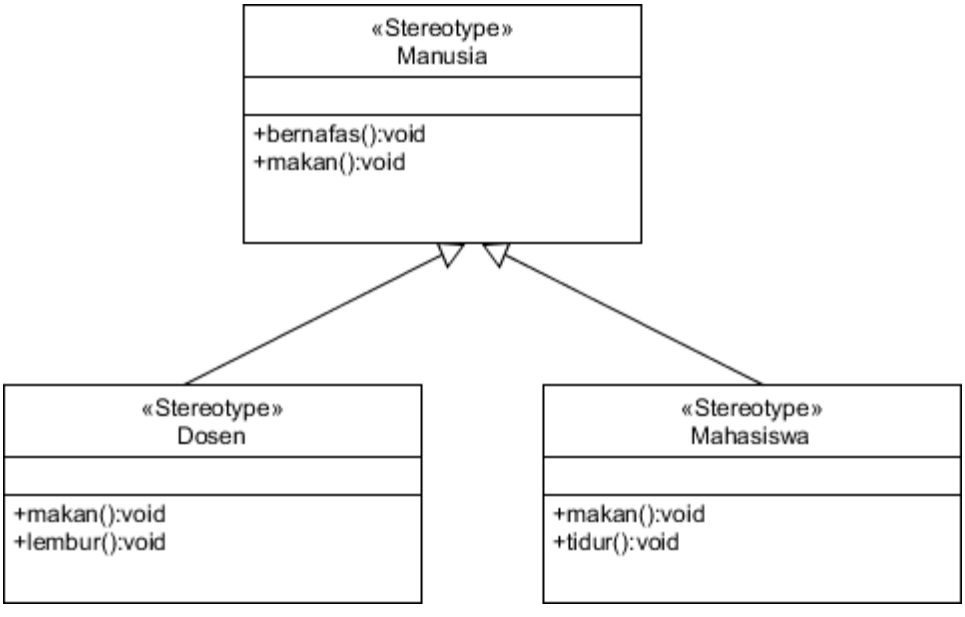
Tugas
5.1 Overloading



```java
package overloading;
public class Segitiga {
    private int sudut;

    public int totalSudut(int sudutA){
        return this.sudut = 180 - sudutA;
    }
    public int totalSudut(int sudutA, int sudutB){
        return this.sudut = 180 - (sudutA+sudutB);
    }
    public int keliling (int sisiA, int sisiB,  int sisiC){
        return sisiA+sisiB+sisiC;
    }
    public double keliling (int sisiA, int sisiB){
        double c = Math.sqrt(Math.pow(sisiA, 2)+Math.pow(sisiB, 2));
        return c;
    }

    public static void main(String[] args) {
        Segitiga segitiga1 =  new Segitiga();
        int sudutA = 60;
        int sudutB = 70;
        int sisiA = 5;
        int sisiB = 7;
        int sisiC = 8;

        System.out.println("Diketahui: \nsudut A = " + sudutA +
                "\nsudut B = " + sudutB +
                "\nsisi A = " + sisiA +
                "\nsisi B = " + sisiB +
                "\nsisi C = " + sisiC);
        System.out.println("Total sudut dengan satu sudut (" + sudutA + "): " + segitiga1.totalSudut(sudutA));
        System.out.println("Total sudut dengan dua sudut (" + sudutA + ", " +sudutB+ "): "+segitiga1.totalSudut(sudutA, sudutB));
        System.out.println("Keliling dengan dua sisi ("+sisiA+", "+sisiB+ "): "+ segitiga1.keliling(sisiA, sisiB));
        System.out.println("Keliling dengan tiga sisi ("+sisiA+", "+sisiB+", "+sisiC+ "): "+ segitiga1.keliling(sisiA, sisiB, sisiC));
    }
}
```

```
Diketahui:
sudut A = 60
sudut B = 70
sisi A = 5
sisi B = 7
sisi C = 8
Total sudut dengan satu sudut (60): 120
Total sudut dengan dua sudut (60, 70): 50
Keliling dengan dua sisi (5, 7): 8.602325267042627
Keliling dengan tiga sisi (5, 7, 8): 20
------------------------------------------------------
```

5.2 Overriding




```java
package overriding;
public class Manusia {

    public void bernafas(){
        System.out.println( x: "Manusia sedang bernafas");
    }

    public void makan(){
        System.out.println( x: "Manusia sedang makan");
    }
}
```

```java
package overriding;
public class Dosen extends Manusia{
    @Override
    public void makan(){
        System.out.println( x: "Dosen sedang makan");
    }

    public void lembur(){
        System.out.println( x: "Dosen sedang lembur");
    }
}

```

```java
package overriding;
public class Mahasiswa extends Manusia{
    @Override
    public void makan(){
        System.out.println( x: "Mahasiswa sedang makan");
    }

    public void tidur(){
        System.out.println( x: "Mahasiswa sedang tidur");
    }
}

```

```java
package overriding;
public class MainClass {
    public static void main(String[] args) {
        Manusia dosen = new Dosen();
        dosen.makan();
        dosen.bernafas();

        System.out.println();

        Manusia mahasiswa = new Mahasiswa();
        mahasiswa.makan();
        mahasiswa.bernafas();
    }

}
```

```
Dosen sedang makan
Manusia sedang bernafas

Mahasiswa sedang makan
Manusia sedang bernafas
```