**Nama : Febby Novanti Azhara**

**NIM : 1182002002**

**Mata Kuliah : Pemrograman Berorientasi Objek**

Soal

Buatlah sebuah program berisi contoh:

1. Kelas-kelas pewarisan (yaitu ada kelas orangtua dan kelas anak).

2. Kelas orangtua yang menerapkan konsep *overloading.*

3. Kelas yang menerapkan konsep *overriding*.

Jawab :

1. Kelas pewarisan atau Inheritance

*Inheritance* atau pewarisan atau penurunan adalah konsep pemrograman dimana sebuah *class* dapat menurunkan *property* dan *method* yang dimilikinya kepada sebuah *class* lain. Konsep *inheritance* digunakan untuk memanfaatkan fitur ‘*code reuse’* untuk menghindari duplikasi kode program. *Inheritance* membuat sebuah struktur atau ‘*hierarchy’* *class* dalam kose program. *Class* yang akan diturunkan bisa disebut sebagai *class* induk atau *parent class*, *super class*, atau *base class.* Sedangkan *class* yang menerima penurunan bisa disebut sebagai *class* anak atau *child class,* *sub class, derived class* atau *heir class*.

Berikut contoh program :

* Karyawan.java

package app.coba;

public class Karyawan {

    private String nama;

    private String nip;

    private String golongan;

    private double 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=5500000;

        break;

      case '2':this.gaji=4000000;

        break;

      case '3':this.gaji=3000000;

        break;

      case '4':this.gaji=1500000;

        break;

      case '5':this.gaji=850000;

        break;

     }

    }

    public void setGaji(double gaji)

    {

     this.gaji=gaji;

    }

    public String getNama()

    {

     return nama;

    }

    public String getNip()

    {

     return nip;

    }

    public String getGolongan()

    {

     return golongan;

    }

    public double getGaji()

    {

     return gaji;

    }

   }

* Staff.java

package app.coba;

public class Staff extends Karyawan {

    private int lembur;

    private double gajiLembur;

    public void setLembur(int lembur)

    {

     this.lembur=lembur;

    }

    public int getLembur()

    {

     return lembur;

    }

    public void setGajiLembur(double gajiLembur)

    {

     this.gajiLembur=gajiLembur;

    }

    public double getGajiLembur()

    {

     return gajiLembur;

    }

    public double getGaji(int lembur,double gajiLembur)

    {

     return super.getGaji()+lembur\*gajiLembur;

    }

    public double getGaji()

    {

     return super.getGaji()+lembur\*gajiLembur;

    }

    public void lihatInfo()

    {

    System.out.println("--------------------");

     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());

     System.out.println("--------------------");

    }

   }

* Manager.java

package app.coba;

public class Manager extends Karyawan {

    private double tunjangan;

    private String bagian;

    private Staff st[];

    public void setTunjangan(double tunjangan)

    {

     this.tunjangan=tunjangan;

    }

    public double getTunjangan()

    {

     return tunjangan;

    }

    public void setBagian(String bagian)

    {

     this.bagian=bagian;

    }

    public String getBagian()

    {

     return bagian;

    }

    public void setStaff(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("Manager  :"+this.getBagian());

     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());

     this.viewStaff();

    }

    public double getGaji()

    {

     return super.getGaji()+tunjangan;

    }

   }

* Main.java

package app.coba;

public class Main {

    public static void main(String[] args)

    {

     System.out.println("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();

     //Febby Novanti Azhara

     man[0].setNama("Febby Novanti Azhara");

     man[0].setNip("101");

     man[0].setGolongan("1");

     man[0].setTunjangan(5500000);

     man[0].setBagian("Administrasi");

     man[1]=new Manager();

     //Rizky Maudy Aryanti

     man[1].setNama("Rizky Maudy Aryanti");

     man[1].setNip("102");

     man[1].setGolongan("1");

     man[1].setTunjangan(3000000);

     man[1].setBagian("Pemasaran");

     // staff Febby Novanti Azhara

     staff1[0]=new Staff();

     staff1[0].setNama("Siti Rahila Azahra");

     staff1[0].setNip("0003");

     staff1[0].setGolongan("2");

     staff1[0].setLembur(10);

     staff1[0].setGajiLembur(10000);

     staff1[1]=new Staff();

     staff1[1].setNama("Fanri Ochadicta");

     staff1[1].setNip("0005");

     staff1[1].setGolongan("2");

     staff1[1].setLembur(10);

     staff1[1].setGajiLembur(55000);

     man[0].setStaff(staff1);

     //staff Rizky Maudy Aryanti

     staff2[0]=new Staff();

     staff2[0].setNama("Mega Listia Kuswani");

     staff2[0].setNip("0004");

     staff2[0].setGolongan("3");

     staff2[0].setLembur(15);

     staff2[0].setGajiLembur(5500);

     staff2[1]=new Staff();

     staff2[1].setNama("Cikal Aliftia");

     staff2[1].setNip("0006");

     staff2[1].setGolongan("4");

     staff2[1].setLembur(5);

     staff2[1].setGajiLembur(100000);

     staff2[2]=new Staff();

     staff2[2].setNama("Rafiqa Amanda Putri");

     staff2[2].setNip("0007");

     staff2[2].setGolongan("3");

     staff2[2].setLembur(6);

     staff2[2].setGajiLembur(20000);

     man[1].setStaff(staff2);

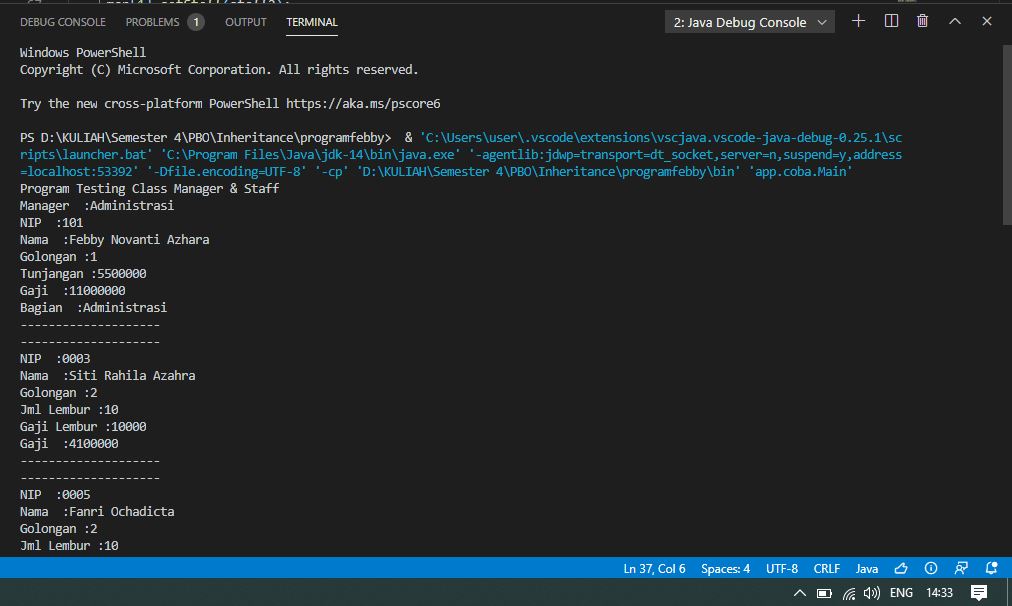
     //cetak informasi dari manager + staffnya

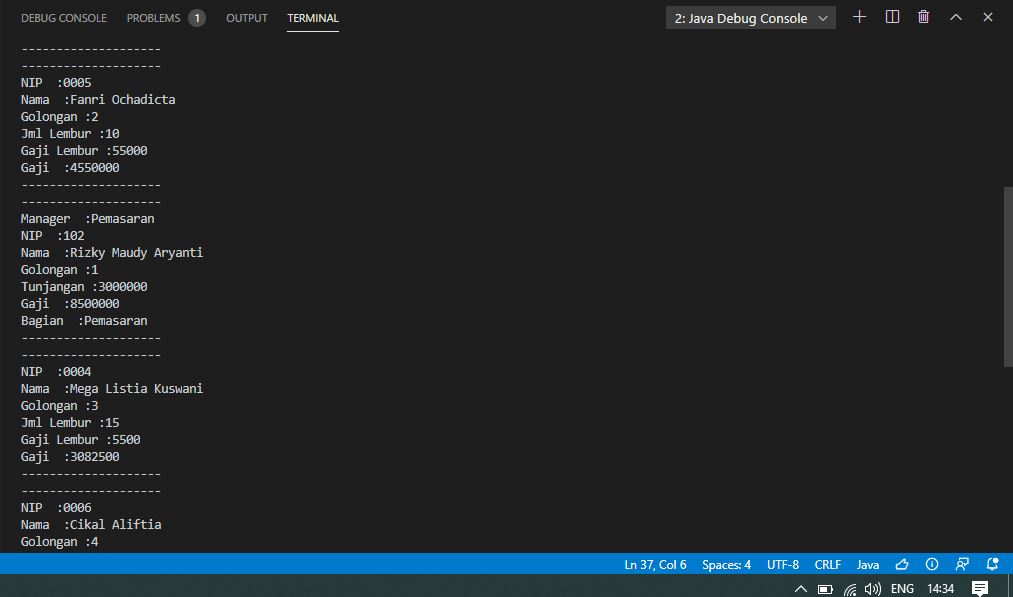
     man[0].lihatInfo();

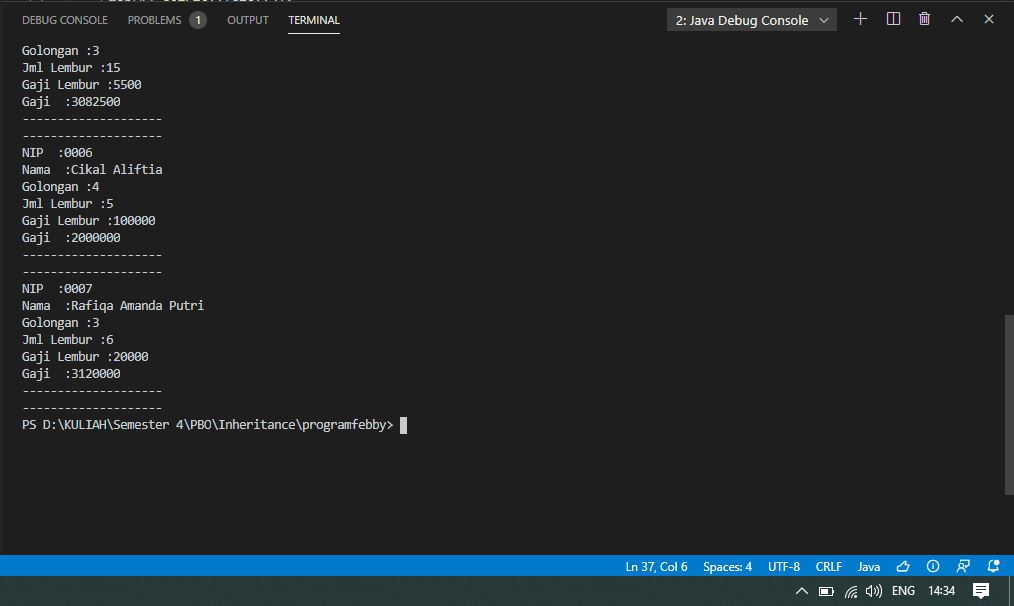
     man[1].lihatInfo();

     }

   }

Berikut hasil dari kodingan diatas :





1. Overloading

Overloading merupakan sebuah method yang terjadi ketika dua atau lebih metode dalam satu kelas memiliki nama metode yang sama namun memiliki parameter yang berbeda.

Berikut contoh program overloading :

package app.overloding;

   public class perkalianku {

    void perkalian(int a, int b){

    System.out.println("Hasil Perkalian adalah :");

    System.out.println(a \* b);

    }

    void perkalian(double a, double b){

    System.out.println("Hasil Perkalian adalah :");

     System.out.println(a \* b);

    }

    public static void main(String args []){

     perkalianku objek = new perkalianku();

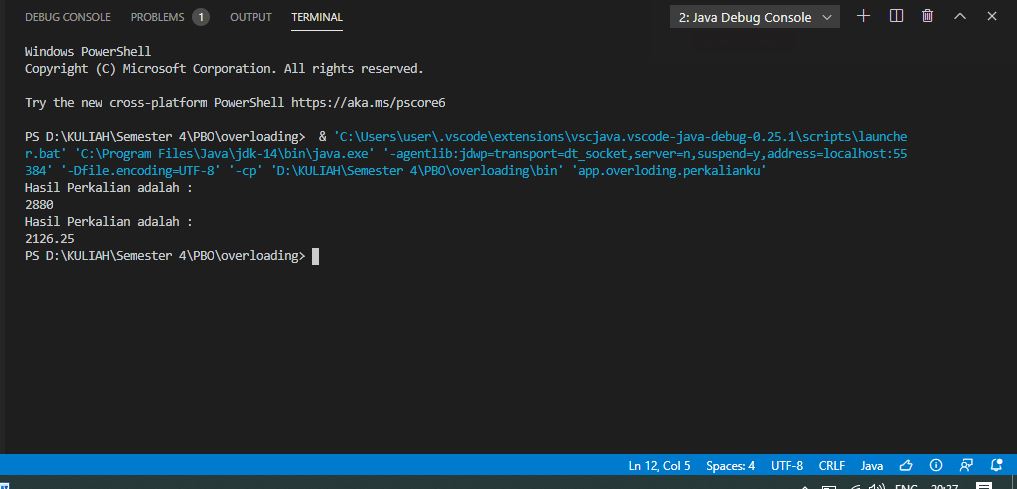
     objek.perkalian(80, 36);

     objek.perkalian(37.5, 56.7);

    }

   }

Berikut hasil nya :



1. Overidding

Overidding berarti memiliki dua buah metode dengan nama dan parameter yang sama berada pada superclass dan subclass pada sebuah program. Overidding memungkinkan subclass untuk merubah dan memodifikasi statement – statement pada metode yang dimiliki oleh superclass.

package app.telpon;

import java.util.Scanner;

public class cekhp {

public static void main(String[] args) {

      String phoneStr;

      System.out.print("Masukan nomor hp anda: ");

      Scanner s = new Scanner(System.in);

      phoneStr = s.next();

      System.out.println(" ");

      System.out.print("No : " + phoneStr);

      char checkPola1 = phoneStr.charAt(3);

      char checkPola2 = phoneStr.charAt(7);

      char checkPola3 = phoneStr.charAt(8);

      int panjang = phoneStr.length();

      int syarat=0;

      for(int x=0;x<panjang;x++){

          char angka = phoneStr.charAt(x);

          if(angka == '1' || angka == '2' ||angka == '3' ||angka == '4' ||angka == '5' ||angka == '6' ||angka == '7' ||angka == '8' ||angka == '9' ||angka == '0' ||angka == '-'){

              syarat=1;

          }

          else{

              syarat=0;

              System.out.println(" TIDAK VALID!! ");

              break;

          }

        }

      if(syarat==1){

          if(checkPola1=='-'&&checkPola2=='-'){

                 System.out.println(" VALID!! ");

                }

          else if(checkPola1=='-'&&checkPola3=='-'){

                 System.out.println(" VALID!! ");

                }

          else{

                 System.out.println(" TIDAK VALID!! ");

                }

      }

    }

}

Berikut hasilnya :

