**PEMROGRAMAN BERBASIS OBJEK**

**(Tugas 7)**

****

**Disusun Oleh:**

Prames Ray Lapian – 140810210059

**PROGRAM STUDI S-1 TEKNIK INFORMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS PADJADJARAN**

**JATINANGOR**

**2022**

1. Soal 1:
   1. Source Code:
      1. C++

*/\**

*Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*Kelas         : A*

*Tanggal       : 19 Oktober 2022*

*Nama Program  : Soal1\_ArrOOP.cpp*

*Deskripsi     : program untuk mencari nilai mahasiswa*

*\*/*

*#include*<iostream>

class Waktu{

    private:

        int jam,menit,detik;

    public:

        Waktu(int *jam*,int *menit*,int *detik*){

            this->jam = *jam*;

            this->menit = *menit*;

            this->detik = *detik*;

        }

        Waktu(){

            this->jam = 0;

            this->menit =0;

            this->detik=0;

        }

        void setJam(int *jam*){

            this->jam = *jam*;

        }

        void setMenit(int *menit*){

            this->menit = *menit*;

        }

        void setDetik(int *detik*){

            this->detik = *detik*;

        }

        void inputWaktu(){

            std::cout<<"Masukkan jam : ";

            std::cin>>this->jam;

            std::cout<<"Masukkan menit : ";

            std::cin>>this->menit;

            std::cout<<"Masukkan detik : ";

            std::cin>>this->detik;

        }

        int getJam(){

*return* this->jam;

        }

        int getDetik(){

*return* this->detik;

        }

        std::string getWaktu(){

            std::string nolJam ="";

            std::string nolMenit="";

            std::string nolDetik="";

*if*(this->jam<10){

                nolJam="0";

            }

*if*(this->menit<10){

                nolMenit="0";

            }

*if*(this->detik<10){

                nolDetik="0";

            }

*return* nolJam + std::to\_string(this->jam) + ":" + nolMenit+ std::to\_string(this->menit) + ":" +nolDetik+ std::to\_string(this->detik);

        }

        int convertToSecond(){

            int hasil = this->detik + this->menit\*60 + this->jam\*3600;

*return* hasil;

        }

        void secondToClock(int *second*){

            this->menit=*second*/60;

            this->detik=*second*%60;

            this->jam=this->menit/60;

            this->menit=this->menit%60;

        }

        Waktu cariDurasi(Waktu *akhir*){

             Waktu temp;

            int detikAwal = this->convertToSecond();

            int detikAkhir = *akhir*.convertToSecond();

*if*(detikAkhir<detikAwal){

                detikAkhir+=86400;

            }

            int detikHasil = detikAkhir - detikAwal;

            temp.secondToClock(detikHasil);

*return* temp;

        }

};

class Ujian{

    private:

        float ulangan;

        Waktu mulai;

        Waktu selesai;

    public:

        Ujian(){

            this->ulangan = 0;

        }

*//Input*

        void setUlangan(int *ulangan*){

            this->ulangan = *ulangan*;

        }

        void setWaktuMulai(Waktu *mulai*){

            this->mulai = *mulai*;

        }

         void setWaktuSelesai(Waktu *selesai*){

            this->selesai = *selesai*;

        }

        void inputUlangan(){

            std::cout<<"Nilai Ulangan : ";

            std::cin>>this->ulangan;

            std::cout<<"\n---Waktu Mulai Ujian ---\n";

            mulai.inputWaktu();

             std::cout<<"\n---Waktu Selesai Ujian ---\n";

            selesai.inputWaktu();

        }

*//Output*

        void outputUjian(){

            std::cout<<"Nilai Ujian : "<<this->ulangan<<"\n";

            std::cout<<"Waktu Mulai : "<<this->mulai.getWaktu();

            std::cout<<"Waktu Selesai : "<<this->selesai.getWaktu();

            std::cout<<"Huruf Mutu : "<<this->getMutuNilai();

        }

        float getUlangan(){

*return* this->ulangan;

        }

        Waktu getWaktuDatang(){

*return* this->mulai;

        }

        Waktu getWaktuPulang(){

*return* this->selesai;

        }

*//Proses*

        char getMutuNilai(){

            char mutuNilai;

*if*(this->ulangan>=0 && this->ulangan<45){

                mutuNilai='E';

            }

*else* *if*(this->ulangan>=45 && this->ulangan<55){

                mutuNilai='D';

            }

*else* *if*(this->ulangan>=55 && this->ulangan<68){

                mutuNilai='C';

            }

*else* *if*(this->ulangan>=68 && this->ulangan<80){

                mutuNilai='B';

            }

*else* *if*(this->ulangan>=80 && this->ulangan<=100){

                mutuNilai= 'A';

            }

*return* mutuNilai;

        }

};

class Mahasiswa{

    private:

        std::string NPM;

        std::string nama;

        int banyakUjian;

        Ujian kumpUjian[100];

    public:

    Mahasiswa(){

        this->NPM = " ";

        this->nama=" ";

        this->banyakUjian=1;

    }

    void setNama(std::string *nama*){

        this->nama = *nama*;

    }

    void setNPM(std::string *NPM*){

        this->NPM = *NPM*;

    }

    void setnilaiMahasiswa(float *nilaimahasiswa*,int *index*){

        this->kumpUjian[*index*].setUlangan(*nilaimahasiswa*);

    }

    void setBanyakUjian(int *banyakUjian*){

        this->banyakUjian = *banyakUjian*;

    }

    void inputKumpUjian(){

*for*(int i=0;i<this->banyakUjian;i++){

            std::cout<<"\nInput ujian  ke "<<(i+1) <<" atas nama "<<this->nama<<" dengan npm "<<this->NPM<<"\n";

            kumpUjian[i].inputUlangan();

        }

    }

    void inputMahasiswa(){

        std::cout<<"NPM Mahasiswa  : ";

        std::cin>>this->NPM;

        std::cout<<"Nama Mahasiswa : ";

        std::cin>>this->nama;

        std::cout<<"Banyak ujian  yang diikuti  : ";

        std::cin>>this->banyakUjian;

       inputKumpUjian();

    }

*//Output*

    std::string getNama(){

*return* this->nama;

    }

    std::string getNPM(){

*return* this->NPM;

    }

    int getBanyakUjian(){

*return* this->banyakUjian;

    }

    Ujian getUjianByIndex(int *index*){

*return* this->kumpUjian[*index*];

    }

    float getnilaiMahasiswaByIndex(int *index*){

*return* this->kumpUjian[*index*].getUlangan();

    }

    void outputKumpUjian(){

*for*(int i=0;i<this->banyakUjian;i++){

            std::cout<<"Ujian ke - "<<(i+1)<<"\n";

            kumpUjian[i].outputUjian();

        }

    }

    void outputMahasiswa(){

        std::cout<<"NPM         : " << this->NPM;

        std::cout<<"Nama        : " <<this->nama;

        std::cout<<"\n--- Nilai Ujian ---\n";

        outputKumpUjian();

        std::cout<<"Keterangan : "<<hitungKeterangan();

    }

*//Proses*

   float getRataRataUjian(){

    float rataRata = 0;

*for*(int i=0;i<this->banyakUjian;i++){

            rataRata += kumpUjian[i].getUlangan();

    }

      rataRata = rataRata / banyakUjian;

*return* rataRata;

   }

   float getNilaiTertinggi(){

    float tertinggi = -999;

*for*(int i=0;i<this->banyakUjian;i++){

*if*(tertinggi < this->kumpUjian[i].getUlangan()){

            tertinggi = this->kumpUjian[i].getUlangan();

        }

    }

*return* tertinggi;

   }

   float getNilaiTerendah(){

    float terendah = 999;

*for*(int i=0;i<this->banyakUjian;i++){

*if*(terendah > this->kumpUjian[i].getUlangan()){

            terendah = this->kumpUjian[i].getUlangan();

        }

    }

*return* terendah;

   }

    std::string hitungKeterangan(){

        std::string status;

*if* (getRataRataUjian() >= 55) {

            status = "LULUS";

        } *else* {

            status = "GAGAL";

        }

*return* status;

    }

};

class LarikMahasiswa{

    private:

        int ukuranMahasiswa;

        Mahasiswa mhs[100];

    public:

        LarikMahasiswa();

        void setUkuran(int *ukuranMahasiswa*);

        int getUkuran();

        float getNilaiTertinggiMahasiswa();

        float getNilaiTerendahMahasiswa();

        float getRataRataTertinggi();

        float getRataRataTerendah();

        void inputUkuranMahasiswa();

        void inputMahasiswa();

        void tampilkanTabelMahasiswa();

};

LarikMahasiswa::LarikMahasiswa(){

            this->ukuranMahasiswa = 0;

}

void LarikMahasiswa::setUkuran(int *ukuranMahasiswa*){

            this->ukuranMahasiswa = *ukuranMahasiswa*;

}

int LarikMahasiswa::getUkuran(){

*return* this->ukuranMahasiswa;

}

float LarikMahasiswa::getNilaiTertinggiMahasiswa(){

            float tinggi = -999;

            float temp = 0;

*for*(int i=0;i<this->ukuranMahasiswa;i++){

                temp = this->mhs[i].getNilaiTertinggi();

*if*(tinggi<temp){

                    tinggi = temp;

                }

            }

*return* tinggi;

}

float LarikMahasiswa::getNilaiTerendahMahasiswa(){

            float rendah = 999;

            float temp = 0;

*for*(int i=0;i<this->ukuranMahasiswa;i++){

                temp = this->mhs[i].getNilaiTerendah();

*if*(rendah>temp){

                    rendah = temp;

                }

            }

*return* rendah;

}

float LarikMahasiswa::getRataRataTertinggi(){

            float tinggi = -999;

            float temp = 0;

*for*(int i=0;i<this->ukuranMahasiswa;i++){

                temp = this->mhs[i].getRataRataUjian();

*if*(tinggi<temp){

                    tinggi = temp;

                }

            }

*return* tinggi;

}

float LarikMahasiswa::getRataRataTerendah(){

            float rendah = 999;

            float temp = 0;

*for*(int i=0;i<this->ukuranMahasiswa;i++){

                temp = this->mhs[i].getRataRataUjian();

*if*(rendah>temp){

                    rendah = temp;

                }

            }

*return* rendah;

}

void LarikMahasiswa::inputUkuranMahasiswa(){

            std::cout<<"Ukuran Mahasiswa : ";

            std::cin>>this->ukuranMahasiswa;

}

void LarikMahasiswa::inputMahasiswa(){

*for*(int i=0;i<this->ukuranMahasiswa;i++){

                std::cout<<"Input Mahasiswa ke - "<<(i+1) <<"\n\n";

                this->mhs[i].inputMahasiswa();

                system("cls");

            }

}

void LarikMahasiswa::tampilkanTabelMahasiswa(){

        int no = 1;

        std::cout<<"\t\t\t\t[DAFTAR NILAI UJIAN MAHASISWA TI]\n";

*if*(this->mhs[0].getNPM()==" "){

            std::cout<<"Data kosong ! \n";

        }

*else*{

            std::cout<<"================================================================================================================================\n";

            std::cout<<"No\tNPM\tNama\tStatus\t\tNilai Ujian\tHM\t\tMulai\t\tSelesai\t\tLama\n";

            std::cout<<"================================================================================================================================\n";

*for*(int i=0;i<this->ukuranMahasiswa;i++){

*if*(this->mhs[i].getNPM() == " "){

*break*;

                }

*else*{

                    std::cout<<

                        no << "\t" <<

                        this->mhs[i].getNPM() << "\t" <<

                        this->mhs[i].getNama() << "\t"<<

                        this->mhs[i].hitungKeterangan()<<"\t\t";

*for*(int j=0;j<this->mhs[i].getBanyakUjian();j++){

                            std::cout<<this->mhs[i].getUjianByIndex(j).getUlangan() << "\t\t"

                            << this->mhs[i].getUjianByIndex(j).getMutuNilai()<<"\t\t"

                            <<this->mhs[i].getUjianByIndex(j).getWaktuDatang().getWaktu()<<"\t"<<

                            this->mhs[i].getUjianByIndex(j).getWaktuPulang().getWaktu()<<"\t"<<

                            this->mhs[i].getUjianByIndex(j).getWaktuDatang().cariDurasi(this->mhs[i].getUjianByIndex(j).getWaktuPulang()).getWaktu()<<"\t";

                            std::cout<<"\n";

                            std::cout<<"\t\t\t\t\t";

                        }

                    std::cout<<"\n";

                    no++;

                }

            }

            std::cout<<"================================================================================================================================\n";

        }

}

int main(int *argc*, char const \**argv*[])

{

    LarikMahasiswa lm;

    lm.inputUkuranMahasiswa();

    system("cls");

    lm.inputMahasiswa();

    system("cls");

    lm.tampilkanTabelMahasiswa();

    std::cout<<"Nilai Tertinggi : "<<lm.getNilaiTertinggiMahasiswa()<<"\n";

    std::cout<<"Nilai Terendah : "<<lm.getNilaiTerendahMahasiswa()<<"\n";

    std::cout<<"Rata Rata Tertinggi : "<<lm.getRataRataTertinggi()<<"\n";

    std::cout<<"Rata Rata Terendah : "<<lm.getRataRataTerendah()<<"\n";

*return* 0;

}

* + 1. Python

*/\**

*Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*Kelas         : A*

*Tanggal       : 19 Oktober 2022*

*Nama Program  : Soal1\_ArrOOP.py*

*Deskripsi     : program untuk mencari nilai mahasiswa*

*\*/*

*from* os *import* system

class Waktu:

*#Attribute*

    \_\_jam=0

    \_\_menit=0

    \_\_detik=0

*#Constructor*

    def \_\_init\_\_(*self*, \**args*):

*if* (len(*args*) == 3):

*self*.\_\_jam = int(*args*[0])

*self*.\_\_menit = int(*args*[1])

*self*.\_\_detik = int(*args*[2])

*elif*(len(*args*)==0):

*self*.\_\_jam = int(0)

*self*.\_\_menit = int(0)

*self*.\_\_detik = int(0)

*else*:

            print("False number of argument in constructor")

*#Input Method*

    def setJam(*self*,*jam*):

*self*.\_\_jam = int(*jam*)

    def setMenit(*self*,*menit*):

*self*.\_\_menit = int(*menit*)

    def setDetik(*self*,*detik*):

*self*.\_\_detik = int(*detik*)

    def inputWaktu(*self*):

*self*.\_\_jam = int(input("Masukkan jam : "))

*self*.\_\_menit = int(input("Masukkan menit : "))

*self*.\_\_detik = int(input("Masukkan detik : "))

*#Output Method*

    def getJam(*self*):

*return* *self*.\_\_jam

    def getMenit(*self*):

*return* *self*.\_\_menit

    def getDetik(*self*):

*return* *self*.\_\_detik

    def getWaktu(*self*):

        nolJam =""

        nolMenit=""

        nolDetik=""

*if*(*self*.\_\_jam<10):

            nolJam="0"

*if*(*self*.\_\_menit<10):

            nolMenit="0"

*if*(*self*.\_\_detik<10):

            nolDetik="0"

*return* nolJam + str(*self*.\_\_jam) + ":" + str(nolMenit)+ str(*self*.\_\_menit) + ":" +nolDetik+ str(*self*.\_\_detik)

*#Proses*

    def convertToSecond(*self*):

        hasil = *self*.\_\_detik + (int(60) \* *self*.\_\_menit) + (int(3600) \* *self*.\_\_jam)

*return* hasil

    def secondToClock(*self*,*second*:int):

*self*.\_\_menit = int(*second*/60)

*self*.\_\_detik = int(*second*%60)

*self*.\_\_jam = int(*self*.\_\_menit/60)

*self*.\_\_menit = int(*self*.\_\_menit%60)

    def cariDurasi(*self*,*akhir*):

        temp = Waktu()

        detikAwal = *self*.convertToSecond()

        detikAkhir = *akhir*.convertToSecond()

*if*(detikAkhir<detikAwal):

            detikAkhir+=86400

        detikHasil = detikAkhir - detikAwal

        temp.secondToClock(detikHasil)

*return* temp

*#-----------------------------------------------------------------------------------------------------------------------------------*

class Ujian:

*#Attribute*

    \_\_ulangan = float(0.0)

    \_\_mulai = Waktu()

    \_\_selesai = Waktu()

*#Constructor*

    def \_\_init\_\_(*self*):

*self*.\_\_ulangan=0

*self*.\_\_mulai = Waktu(0,0,0)

*self*.\_\_selesai = Waktu(0,0,0)

*#Input*

    def setUlangan(*self*,*ulangan*):

*self*.\_\_ulangan = *ulangan*

    def setWaktuMulai(*self*,*mulai*):

*self*.\_\_mulai = *mulai*

    def setWaktuSelesai(*self*,*selesai*):

*self*.\_\_selesai = *selesai*;

    def inputUlangan(*self*):

*self*.\_\_ulangan = float(input("Masukkan Nilai Ulangan : "))

        print("\n---Waktu Mulai Ujian---")

*self*.\_\_mulai.inputWaktu()

        print("\n---Waktu Selesai Ujian---")

*self*.\_\_selesai.inputWaktu()

*#Output*

    def outputUjian(*self*):

        print("Nilai Ujian : " , *self*.\_\_ulangan)

        print("Waktu Mulai : " , *self*.\_\_mulai.getWaktu())

        print("Waktu Selesai : " , *self*.\_\_selesai.getWaktu())

        print("Huruf Mutu : " , *self*.\_getMutuNilai())

    def getUlangan(*self*):

*return* float(*self*.\_\_ulangan)

    def getWaktuDatang(*self*):

*return* *self*.\_\_mulai

    def getWaktuPulang(*self*):

*return* *self*.\_\_selesai

*#Proses*

    def getMutuNilai(*self*):

        mutuNilai = ' '

*if*(*self*.\_\_ulangan>=0 and *self*.\_\_ulangan<45):

            mutuNilai = 'E'

*elif*(*self*.\_\_ulangan>=45 and *self*.\_\_ulangan<55):

            mutuNilai='D'

*elif*(*self*.\_\_ulangan>=55 and *self*.\_\_ulangan<68):

            mutuNilai='C'

*elif*(*self*.\_\_ulangan>=68 and *self*.\_\_ulangan<80):

            mutuNilai='B'

*elif*(*self*.\_\_ulangan>=80 and *self*.\_\_ulangan<=100):

                mutuNilai= 'A';

*return* mutuNilai

*#-----------------------------------------------------------------------------------------------------------------------------------*

class Mahasiswa:

*#Attribute*

    \_\_npm = " "

    \_\_nama = " "

    \_\_banyakUjian = int(0)

    \_\_kumpUjian = []

*#Constructor*

    def \_\_init\_\_(*self*):

*self*.\_\_NPM = " "

*self*.\_\_nama = " "

*self*.\_\_banyakUjian = int(0)

*self*.\_kumpUjian = []

*#Input*

    def setNama(*self*,*nama*):

*self*.\_\_nama = *nama*;

    def setNPM(*self*,*NPM*):

*self*.\_\_NPM = *NPM*

    def setnilaiMahasiswa(*self*,*nilaimahasiswa*,*index*):

*self*.\_\_kumpUjian[*index*].setUlangan(*nilaimahasiswa*)

    def setBanyakUjian(*self*,*banyakUjian*):

*self*.\_\_banyakUjian = *banyakUjian*

    def inputKumpUjian(*self*):

        i = int(0)

*while*(i<*self*.\_\_banyakUjian):

            print("\nInput ujian  ke ",(i+1) ," atas nama ",*self*.\_\_nama," dengan npm ",*self*.\_\_npm,"\n")

            obj = Ujian()

            obj.inputUlangan()

*self*.\_\_kumpUjian.append(obj)

            i = i+1

    def inputMahasiswa(*self*):

*self*.\_\_NPM = input("NPM Mahasiswa  : ")

*self*.\_\_nama = input("Nama Mahasiswa : ")

*self*.\_\_banyakUjian = int(input("Banyak ujian  yang diikuti  : "))

*self*.inputKumpUjian()

*#Output*

    def getNama(*self*):

*return* *self*.\_\_nama

    def getNPM(*self*):

*return* *self*.\_\_NPM

    def getBanyakUjian(*self*):

*return* *self*.\_\_banyakUjian;

    def getUjianByIndex(*self*,*index*):

*return* *self*.\_\_kumpUjian[*index*]

    def getnilaiMahasiswaByIndex(*self*, *index*):

*return* *self*.\_\_kumpUjian[*index*].getUlangan()

    def outputKumpUjian(*self*):

        i = int(0)

*while*(i<*self*.\_\_banyakUjian):

            print("Ujian ke - ",(i+1))

*self*.\_\_kumpUjian[i].outputUjian()

            i = i+1

    def outputMahasiswa(*self*):

        print("NPM         : " , *self*.\_\_NPM)

        print("NPM         : " , *self*.\_\_nama)

        print("\n---Nilai Ujian---\n")

*self*.\_\_outputKumpUjian()

        print("Keterangan : " , *self*.\_\_hitungKeterangan())

*#Proses*

    def getRataRataUjian(*self*):

        rataRata = float(0)

        i = int(0)

*while*(i<*self*.\_\_banyakUjian):

            rataRata += *self*.\_\_kumpUjian[i].getUlangan()

            i = i+1

        rataRata = rataRata / *self*.\_\_banyakUjian

*return* rataRata;

    def getNilaiTertinggi(*self*):

        tertinggi = float(-999)

        i = int(0)

*while*(i<*self*.\_\_banyakUjian):

*if*(tertinggi < *self*.\_\_kumpUjian[i].getUlangan()):

                tertinggi = *self*.\_\_kumpUjian[i].getUlangan()

            i = i+1

*return* tertinggi;

    def getNilaiTerendah(*self*):

        terendah = float(999)

        i = int(0)

*while*(*self*.\_\_banyakUjian):

*if*(terendah > *self*.\_\_kumpUjian[i].getUlangan()):

                terendah = *self*.\_\_kumpUjian[i].getUlangan()

*return* terendah;

    def hitungKeterangan(*self*):

        status = " "

*if* (*self*.getRataRataUjian() >= 55) :

            status = "LULUS"

*else* :

            status = "GAGAL"

*return* status

*#-----------------------------------------------------------------------------------------------------------------------------------*

class LarikMahasiswa:

*#Attribute*

    \_\_ukuranMahasiswa = int(0)

    \_\_mhs = []

*#Constructor*

    def \_\_init\_\_(*self*):

*self*.\_\_ukuranMahasiswa = 2

*#Input*

    def setUkuran(*self*,*ukuranMahasiswa*):

*self*.\_\_ukuranMahasiswa = *ukuranMahasiswa*

    def inputUkuranMahasiswa(*self*):

*self*.\_\_ukuranMahasiswa = int(input("Ukuran Mahasiswa : "))

    def inputMahasiswa(*self*):

        i = int(0)

*while*(i<*self*.\_\_ukuranMahasiswa):

            print("Input Mahasiswa ke - ",(i+1) , "\n\n")

            obj = Mahasiswa()

            obj.inputMahasiswa()

*self*.\_\_mhs.append(obj)

            i = i +1

            system("cls")

    def getUkuran(*self*):

*return* *self*.\_\_ukuranMahasiswa

    def getNilaiTertinggiMahasiswa(*self*):

        tinggi = float(-999);

        temp = float(0);

        i = int(0)

*while*(i<*self*.\_\_ukuranMahasiswa):

            temp = *self*.\_\_mhs[i].getNilaiTertinggi();

*if*(tinggi<temp):

                tinggi = temp;

            i = i +1

*return* tinggi

    def getNilaiTerendahMahasiswa(*self*):

        rendah = float(999);

        temp = float(0);

        i = int(0)

*while*(i<*self*.\_\_ukuranMahasiswa):

            temp = *self*.\_\_mhs[i].getNilaiTertinggi();

*if*(rendah>temp):

                rendah= temp

            i = i +1

*return* rendah

    def getRataRataTertinggi(*self*):

        tinggi = float(-999);

        temp = float(0);

        i = int(0)

*while*(i<*self*.\_\_ukuranMahasiswa):

            temp = *self*.\_\_mhs[i].getRataRataUjian();

*if*(tinggi<temp):

                tinggi = temp

            i = i +1

*return* tinggi

    def getRataRataTerendah(*self*):

        rendah = float(999);

        temp = float(0);

        i = int(0)

*while*(i<*self*.\_\_ukuranMahasiswa):

            temp = *self*.\_\_mhs[i].getRataRataUjian()

*if*(rendah>temp):

                rendah= temp

            i = i +1

*return* rendah

    def tampilkanTabelMahasiswa(*self*):

        no =int(1);

        print("\t\t\t\t==== Daftar Gaji Harian PT Informatika ====")

*if*(*self*.\_\_mhs[0].getNPM()==" "):

            print("Data kosong ! ")

*else*:

            print("================================================================================================================================")

            print("No\tNPM\tNama\tStatus\t\tNilai Ujian\tHM\t\tMulai\t\tSelesai\t\tLama")

            print("================================================================================================================================")

            i = int(0)

*while*(i<*self*.\_\_ukuranMahasiswa):

*if*(*self*.\_\_mhs[i].getNPM() == " "):

*break*

*else*:

                    print ( no , "\t" , *self*.\_\_mhs[i].getNPM() , "\t" ,*self*.\_\_mhs[i].getNama() ,"\t",*self*.\_\_mhs[i].hitungKeterangan() , "\t\t" , *end*="" )

                    j = int(0)

*while*(j<*self*.\_\_mhs[i].getBanyakUjian()):

                        print(*self*.\_\_mhs[i].getUjianByIndex(j).getUlangan() , "\t\t"

                            , *self*.\_\_mhs[i].getUjianByIndex(j).getMutuNilai(),"\t\t"

                            ,*self*.\_\_mhs[i].getUjianByIndex(j).getWaktuDatang().getWaktu(),"\t",

*self*.\_\_mhs[i].getUjianByIndex(j).getWaktuPulang().getWaktu(),"\t",

*self*.\_\_mhs[i].getUjianByIndex(j).getWaktuDatang().cariDurasi(*self*.\_\_mhs[i].getUjianByIndex(j).getWaktuPulang()).getWaktu(),"\t")

                        print("\t\t\t\t\t",*end*="")

                        j= j +1

                    print("\n")

                    no = no+1

                i = i+1

            print("================================================================================================================================")

*#-----------------------------------------------------------------------------------------------------------------------------------*

lm = LarikMahasiswa()

lm.inputUkuranMahasiswa()

system("cls")

lm.inputMahasiswa()

system("cls")

lm.tampilkanTabelMahasiswa();

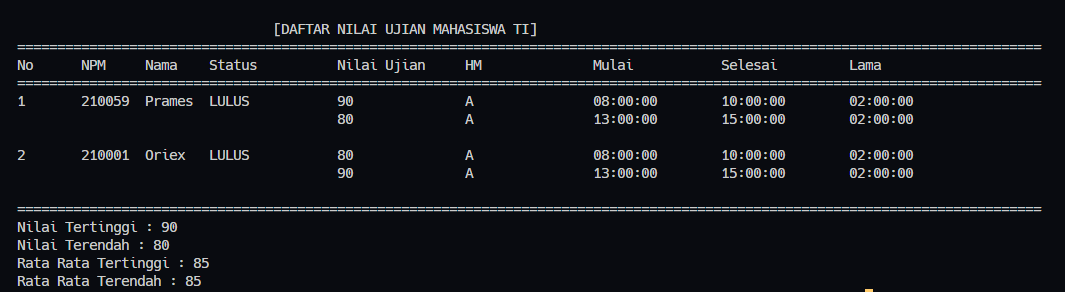
print("Nilai Tertinggi : ",lm.getNilaiTertinggiMahasiswa())

print("Nilai Terendah : ",lm.getNilaiTerendahMahasiswa())

print("Rata Rata Tertinggi : ",lm.getRataRataTertinggi())

print("Rata Rata Terendah : ",lm.getRataRataTerendah())

* 1. Screenshot:



1. Soal 2
   1. Source Code:
      1. C++

*/\**

*Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*Kelas         : A*

*Tanggal       : 19 Oktober 2022*

*Nama Program  : Soal2\_ArrOOP.cpp*

*Deskripsi     : program untuk mencari pegawai dan gajinya*

*\*/*

*#include*<iostream>

class Waktu{

    private:

        int jam,menit,detik;

    public:

        Waktu(int *jam*,int *menit*,int *detik*){

            this->jam = *jam*;

            this->menit = *menit*;

            this->detik = *detik*;

        }

        Waktu(){

            this->jam = 0;

            this->menit =0;

            this->detik=0;

        }

        void setJam(int *jam*){

            this->jam = *jam*;

        }

        void setMenit(int *menit*){

            this->menit = *menit*;

        }

        void setDetik(int *detik*){

            this->detik = *detik*;

        }

        void inputWaktu(){

            std::cout<<"Masukkan jam : ";

            std::cin>>this->jam;

            std::cout<<"Masukkan menit : ";

            std::cin>>this->menit;

            std::cout<<"Masukkan detik : ";

            std::cin>>this->detik;

        }

        int getJam(){

*return* this->jam;

        }

        int getDetik(){

*return* this->detik;

        }

        std::string getWaktu(){

            std::string nolJam ="";

            std::string nolMenit="";

            std::string nolDetik="";

*if*(this->jam<10){

                nolJam="0";

            }

*if*(this->menit<10){

                nolMenit="0";

            }

*if*(this->detik<10){

                nolDetik="0";

            }

*return* nolJam + std::to\_string(this->jam) + ":" + nolMenit+ std::to\_string(this->menit) + ":" +nolDetik+ std::to\_string(this->detik);

        }

        int convertToSecond(){

            int hasil = this->detik + this->menit\*60 + this->jam\*3600;

*return* hasil;

        }

        void secondToClock(int *second*){

            this->menit=*second*/60;

            this->detik=*second*%60;

            this->jam=this->menit/60;

            this->menit=this->menit%60;

        }

        Waktu cariDurasi(Waktu *akhir*){

             Waktu temp;

            int detikAwal = this->convertToSecond();

            int detikAkhir = *akhir*.convertToSecond();

*if*(detikAkhir<detikAwal){

                detikAkhir+=86400;

            }

            int detikHasil = detikAkhir - detikAwal;

            temp.secondToClock(detikHasil);

*return* temp;

        }

};

class Pegawai{

  private:

    std::string nama,nip;

    int gol;

    Waktu datang;

    Waktu pulang;

  public:

    Pegawai(){

        this->nip=" ";

        this->nama=" ";

        this->gol=0;

    }

    void setNama(std::string *nama*){

        this->nama = *nama*;

    }

    void setNIP(std::string *nip*){

        this->nip = *nip*;

    }

    void setGol(int *gol*){

        this->gol = *gol*;

    }

    void setWaktuDatang(Waktu *datang*){

        this->datang = *datang*;

    }

    void setWaktuPulang(Waktu *pulang*){

        this->pulang = *pulang*;

    }

    void inputPegawai(){

        std::cout<<"--- Input Pegawai ---\n";

        std::cout<<"NIP Pegawai : ";

        std::cin>>this->nip;

        std::cout<<"Nama Pegawai : ";

        std::cin>>this->nama;

        std::cout<<"Golongan gaji : ";

        std::cin>>this->gol;

        std::cout<<"\n--- Jam Mulai Kerja---\n";

        this->datang.inputWaktu();

        std::cout<<"\n--- Jam Selesai Kerja---\n";

        this->pulang.inputWaktu();

    }

    std::string getNama(){

*return* this->nama;

    }

    std::string getNIP(){

*return* this->nip;

    }

    int getGol(){

*return* this->gol;

    }

    Waktu getWaktuDatang(){

*return* this->datang;

    }

    Waktu getWaktuPulang(){

*return* this->pulang;

    }

    void outputPegawai(){

        std::cout<<"NIP : "<<this->nip<<"\n";

        std::cout<<"Nama : "<<this->nama<<"\n";

        std::cout<<"Golongan gaji : "<<this->nip<<"\n";

        std::cout<<"Waktu Datang : "<<this->datang.getWaktu();

        std::cout<<"Waktu Pulang : "<<this->pulang.getWaktu();

    }

    Waktu getLamaKerja(){

*return* this->datang.cariDurasi(this->pulang);

    }

    Waktu getWaktuLembur(){

        Waktu delJam(8,0,0);

        Waktu hasil(0,0,0);

*if*(getLamaKerja().getJam() >= 8){

            hasil=delJam.cariDurasi(getLamaKerja());

        }

*return* hasil;

    }

    int getTambahanLembur(){

        int tambahan = 0;

*if*(this->gol==1){

            tambahan = (50000\*getWaktuLembur().getJam());

        }

*else* *if*(this->gol==2){

            tambahan = (70000\*getWaktuLembur().getJam());

        }

*else* *if*(this->gol==3){

            tambahan = (150000\*getWaktuLembur().getJam());

        }

*else* *if*(this->gol==4){

            tambahan = (200000\*getWaktuLembur().getJam());

        }

*return* tambahan;

    }

     int getGajiPokok(){

        int gapok = 0;

*if*(this->gol == 1){

            gapok = 150000;

        }

*else* *if*(this->gol == 2){

            gapok = 200000;

        }

*else* *if*(this->gol == 3){

            gapok = 400000;

        }

*else* *if*(this->gol == 4){

            gapok = 500000;

        }

*return* gapok;

    }

    int getGajiHarian(){

        int tambahan = getTambahanLembur();

        int gapok = getGajiPokok();

        int gajiHarian = tambahan + gapok;

*return* gajiHarian;

    }

    std::string getPeringatan(){

        std::string hasil = "";

*if*(getLamaKerja().getJam()<8){

            hasil = "SP1";

        }

*return* hasil;

    }

};

class LarikPegawai{

    private:

        int ukuran;

        Pegawai larikp[1000];

    public:

        LarikPegawai(int *ukuran*);

        void setUkuran(int *ukuran*);

        void inputLarik();

        void printLarikTabel();

        int getUkuran();

        int getTotalGaji();

};

LarikPegawai::LarikPegawai(int *ukuran*){

    this->ukuran = *ukuran*;

}

void LarikPegawai::setUkuran(int *ukuran*){

    this->ukuran = *ukuran*;

}

void LarikPegawai::inputLarik(){

*for*(int i=0;i<this->ukuran;i++){

        std::cout<<"Pegawai ke - " << i+1<<"\n\n";

        larikp[i].inputPegawai();

        system("cls");

    }

}

int LarikPegawai::getUkuran(){

*return* this->ukuran;

}

void LarikPegawai::printLarikTabel(){

     int no = 1;

        std::cout<<"\t\t\t\t==== Daftar Gaji Harian PT Informatika ====\n";

*if*(this->larikp[0].getNIP()==" "){

            std::cout<<"Data kosong ! \n";

        }

*else*{

            std::cout<<"================================================================================================================================\n";

            std::cout<<"No\tNIP\tNama\tGolongan\tDatang\t\tPulang\t\tLama\t\tJamLembur\tGajiHarian\tStatus\n";

            std::cout<<"================================================================================================================================\n";

*for*(int i=0;i<this->ukuran;i++){

*if*(this->larikp[i].getNIP() == " "){

*break*;

                }

*else*{

                    std::cout<<

                        no << "\t" <<

                        this->larikp[i].getNIP() << "\t" <<

                        this->larikp[i].getNama() << "\t" <<

                        this->larikp[i].getGol() << "\t\t" <<

                        this->larikp[i].getWaktuDatang().getWaktu() << "\t" <<

                        this->larikp[i].getWaktuPulang().getWaktu() << "\t" <<

                        this->larikp[i].getLamaKerja().getWaktu() << "\t " <<

                        this->larikp[i].getWaktuLembur().getWaktu() << "\t" <<

                        this->larikp[i].getGajiHarian() << "\t         " <<

                        this->larikp[i].getPeringatan() << "\t"<<"\n";

                    no++;

                }

            }

            std::cout<<"================================================================================================================================\n";

        }

}

 int LarikPegawai::getTotalGaji(){

        int total = 0;

*for*(int i=0;i<this->ukuran;i++){

            total +=this->larikp[i].getGajiHarian();

        }

*return* total;

}

int main(int *argc*, char const \**argv*[])

{

    int ukuran;

    std::cout<<"Ukuran array : ";

    std::cin>>ukuran;

    system("cls");

    LarikPegawai lp(ukuran);

    lp.inputLarik();

    system("cls");

    lp.printLarikTabel();

    std::cout<<"\nTotal Gaji : "<<lp.getTotalGaji();

*return* 0;

}

* + 1. Python

*/\**

*Nama          : Amir Salim , Andre Nathaniel Adipraja , Prames Ray lapian , Ibrahim Dafi Iskandar*

*NPM           : 140810210015 , 140810200042 , 140810210059 , 140810210039*

*Kelas         : A*

*Tanggal       : 19 Oktober 2022*

*Nama Program  : Soal2\_ArrOOP.py*

*Deskripsi     : program untuk mencari pegawai dan gajinya*

*\*/*

*from* os *import* system

class Waktu:

*#Attribute*

    \_\_jam=0

    \_\_menit=0

    \_\_detik=0

*#Constructor*

    def \_\_init\_\_(*self*, \**args*):

*if* (len(*args*) == 3):

*self*.\_\_jam = int(*args*[0])

*self*.\_\_menit = int(*args*[1])

*self*.\_\_detik = int(*args*[2])

*elif*(len(*args*)==0):

*self*.\_\_jam = int(0)

*self*.\_\_menit = int(0)

*self*.\_\_detik = int(0)

*else*:

            print("False number of argument in constructor")

*#Input Method*

    def setJam(*self*,*jam*):

*self*.\_\_jam = int(*jam*)

    def setMenit(*self*,*menit*):

*self*.\_\_menit = int(*menit*)

    def setDetik(*self*,*detik*):

*self*.\_\_detik = int(*detik*)

    def inputWaktu(*self*):

*self*.\_\_jam = int(input("Masukkan jam : "))

*self*.\_\_menit = int(input("Masukkan menit : "))

*self*.\_\_detik = int(input("Masukkan detik : "))

*#Output Method*

    def getJam(*self*):

*return* *self*.\_\_jam

    def getMenit(*self*):

*return* *self*.\_\_menit

    def getDetik(*self*):

*return* *self*.\_\_detik

    def getWaktu(*self*):

        nolJam =""

        nolMenit=""

        nolDetik=""

*if*(*self*.\_\_jam<10):

            nolJam="0"

*if*(*self*.\_\_menit<10):

            nolMenit="0"

*if*(*self*.\_\_detik<10):

            nolDetik="0"

*return* nolJam + str(*self*.\_\_jam) + ":" + str(nolMenit)+ str(*self*.\_\_menit) + ":" +nolDetik+ str(*self*.\_\_detik)

*#Proses*

    def convertToSecond(*self*):

        hasil = *self*.\_\_detik + (int(60) \* *self*.\_\_menit) + (int(3600) \* *self*.\_\_jam)

*return* hasil

    def secondToClock(*self*,*second*:int):

*self*.\_\_menit = int(*second*/60)

*self*.\_\_detik = int(*second*%60)

*self*.\_\_jam = int(*self*.\_\_menit/60)

*self*.\_\_menit = int(*self*.\_\_menit%60)

    def cariDurasi(*self*,*akhir*):

        temp = Waktu()

        detikAwal = *self*.convertToSecond()

        detikAkhir = *akhir*.convertToSecond()

*if*(detikAkhir<detikAwal):

            detikAkhir+=86400

        detikHasil = detikAkhir - detikAwal

        temp.secondToClock(detikHasil)

*return* temp

*#-----------------------------------------------------------------------------------------------------------------------------------*

class Pegawai:

*#Attributes*

    \_\_nip = " "

    \_\_nama = " "

    \_\_gol = 0

    \_\_datang = Waktu()

    \_\_pulang = Waktu()

*#Constructor*

    def \_\_init\_\_(*self*):

*self*.\_\_nip = " "

*self*.\_\_nama = " "

*self*.\_\_gol = 0

*self*.\_\_datang = Waktu(0,0,0)

*self*.\_\_pulang = Waktu(0,0,0)

*#Input*

    def setNama(*self*,*nama*):

*self*.\_\_nama = *nama*

    def setNIP(*self*,*nip*):

*self*.\_\_nip = *nip*

    def setGol(*self*,*gol*):

*self*.\_\_gol = int(*gol*)

    def setWaktuDatang(*self*,*datang*):

*self*.\_\_datang = *datang*

    def setWaktuPulang(*self*,*pulang*):

*self*.\_\_pulang = *pulang*

    def inputPegawai(*self*):

        print("--- Input Pegawai ---")

*self*.\_\_nip = input("NIP Pegawai : ")

*self*.\_\_nama = input("Nama Pegawai : ")

*self*.\_\_gol = int(input("Golongan gaji : "))

        print("\n--- Jam Mulai Kerja --- ")

*self*.\_\_datang.inputWaktu()

        print("\n--- Jam Selesai Kerja --- ")

*self*.\_\_pulang.inputWaktu()

*# Output*

    def getNama(*self*):

*return* *self*.\_\_nama

    def getNIP(*self*):

*return* *self*.\_\_nip

    def getGol(*self*):

*return* int(*self*.\_\_gol)

    def getWaktuDatang(*self*):

*return* *self*.\_\_datang

    def getWaktuPulang(*self*):

*return* *self*.\_\_pulang

    def outputPegawai(*self*):

        print("NIP : ", *self*.\_\_nip)

        print("Nama : ", *self*.\_\_nama)

        print("Golongan Gaji : ", *self*.\_\_gol)

        print("Waktu Datang : ", *self*.\_\_datang.getWaktu())

        print("Waktu Pulang : ", *self*.\_\_pulang.getWaktu())

*#Proses*

    def getLamaKerja(*self*):

*return* *self*.\_\_datang.cariDurasi(*self*.\_\_pulang)

    def getWaktuLembur(*self*):

        delJam = Waktu(8,0,0)

        hasil = Waktu(0,0,0)

*if*(*self*.getLamaKerja().getJam() >= 8):

            hasil=delJam.cariDurasi(*self*.getLamaKerja())

*return* hasil

    def getTambahanLembur(*self*):

        tambahan = int(0)

*if*(*self*.\_\_gol==1):

            tambahan = int (50000\**self*.getWaktuLembur().getJam())

*elif*(*self*.\_\_gol==2):

            tambahan = int (70000\**self*.getWaktuLembur().getJam())

*elif*(*self*.\_\_gol==3):

            tambahan = int (150000\**self*.getWaktuLembur().getJam())

*elif*(*self*.\_\_gol==4):

            tambahan = int (200000\**self*.getWaktuLembur().getJam())

*return* tambahan

    def getGajiPokok(*self*):

        gapok = int(0)

*if*(*self*.\_\_gol == 1):

            gapok = int(150000)

*elif*(*self*.\_\_gol == 2):

            gapok = int(200000)

*elif*(*self*.\_\_gol == 3):

            gapok = int(400000)

*elif*(*self*.\_\_gol == 4):

            gapok = int(500000)

*return* gapok

    def getGajiHarian(*self*):

        tambahan = *self*.getTambahanLembur()

        gapok = *self*.getGajiPokok()

        gajiHarian = tambahan + gapok

*return* gajiHarian

    def getPeringatan(*self*):

        hasil = ""

*if*(*self*.getLamaKerja().getJam()<8):

            hasil = "SP1"

*return* hasil

*#-----------------------------------------------------------------------------------------------------------------------------------*

class LarikPegawai:

    \_\_ukuran=int(0)

    \_\_larikp = []

*#Constructor*

    def \_\_init\_\_(*self*,*ukuran*):

*self*.\_\_ukuran = *ukuran*

*#Input*

    def setUkuran(*self*,*ukuran*):

*self*.\_\_ukuran = *ukuran*

    def inputLarik(*self*):

        i = int(0)

*while*(i<*self*.\_\_ukuran):

            print("Pegawai ke - " , (i+1),"\n")

            obj = Pegawai()

            obj.inputPegawai()

*self*.\_\_larikp.append(obj)

            system("cls")

            i = i+1

    def getTotalGaji(*self*):

        i = int(0)

        total = int(0)

*while*(i<*self*.\_\_ukuran):

            total = total + *self*.\_\_larikp[i].getGajiHarian()

            i = i+1

*return* total

    def printLarikTabel(*self*):

        print("\t\t\t\t==== Daftar Gaji Harian PT Informatika ====")

        no = int(1)

*if*(*self*.\_\_larikp[0] == " "):

            print("Data Kosong !!!")

*else*:

            i = int(0)

            print("================================================================================================================================")

            print("No\tNIP\tNama\tGolongan\tDatang\t\tPulang\t\tLama\t\tJamLembur\tGajiHarian\tStatus")

            print("================================================================================================================================")

*while*(i<*self*.\_\_ukuran):

*if*(*self*.\_\_larikp[0].getNIP() == " "):

*break*

*else*:

                    print(

                        no , "\t" ,

*self*.\_\_larikp[i].getNIP() , "\t" ,

*self*.\_\_larikp[i].getNama() , "\t" ,

*self*.\_\_larikp[i].getGol() , "\t\t" ,

*self*.\_\_larikp[i].getWaktuDatang().getWaktu() , "\t" ,

*self*.\_\_larikp[i].getWaktuPulang().getWaktu() , "\t" ,

*self*.\_\_larikp[i].getLamaKerja().getWaktu() , "\t " ,

*self*.\_\_larikp[i].getWaktuLembur().getWaktu() , "\t" ,

*self*.\_\_larikp[i].getGajiHarian() , "\t " ,

*self*.\_\_larikp[i].getPeringatan() , "\t"

                    )

                    no = no +1

                i = i+1

            print("================================================================================================================================")

*#-----------------------------------------------------------------------------------------------------------------------------------*

uk = int(input("Banyak Pegawai : "))

lp = LarikPegawai(uk)

system("cls")

lp.inputLarik()

system("cls")

lp.printLarikTabel()

print("Total gaji = " , lp.getTotalGaji())

* 1. Screenshot:

