1. **DEFINISI MASALAH**

Sebelum mengerjakan soal di bawah ini tentukan dahulu objek, attribut, behaviour dan class.

1. Buatlah sebuah sistem sederhana yang menyerupai Sistem Informasi Akademik Mahasiswa (SIAM), dengan ketentuan user menginputkan Nama, Nim, IP serta jurusan. Selain itu mahasiswa juga dapat memasukkan kode Mata kuliah, Nama Mata kuliah dan jumlah sks mata kuliah tersebut. Jumlah sks yang di ambil harus sesuai dengan IP yang di dapat pada semester lalu. Buat skenario dengan banyak mahasiswa minimal 3 orang.
2. **SOURCE CODE**

|  |  |
| --- | --- |
|  | Mahasiswa.java |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124 | package pemrogramanlanjut;  import java.util.Scanner;  public class Mahasiswa {  private String Nama, Nim, Jurusan;  private double Ip;  private int MklMax, MklMin, TotalSKS = 0, Sks, a = 0, pilihan = 1;  private String[][] matkul = new String[10][3];  public Mahasiswa(String Nama, String Nim, String Jurusan, double Ip) {  this.Nama = Nama;  this.Nim = Nim;  this.Jurusan = Jurusan;  this.Ip = Ip;  }  public String getNama() {  return Nama;  }  public String getNim() {  return Nim;  }  public String getJurusan() {  return Jurusan;  }  public double getIp() {  return Ip;  }  public void setTotalSKS() {  if (Ip >= 3.00 && Ip <= 4.00) {  MklMax = 24;  MklMin = 22;  } else if (Ip >= 2.50) {  MklMax = 21;  MklMin = 19;  } else if (Ip >= 2.00) {  MklMax = 18;  MklMin = 16;  } else if (Ip >= 1.50) {  MklMax = 15;  MklMin = 12;  } else if (Ip < 1.50) {  MklMax = 11;  MklMin = 0;  }  }  public void setmatkul() {  Scanner in = new Scanner(System.in);  System.out.println("Total Sks yang bisa diambil adalah " + getMkMin() + "-" + getMkMax());  while (TotalSKS <= MklMax || TotalSKS >= MklMin) {  System.out.print("Masukkan kode mata kuliah : ");  matkul[a][0] = in.nextLine();  System.out.print("Masukkan Nama mata kuliah : ");  matkul[a][1] = in.nextLine();  System.out.print("Masukkan Sks mata kuliah : ");  matkul[a][2] = in.nextLine();  Sks = Integer.valueOf(matkul[a][2]);  TotalSKS += Sks;  if (TotalSKS > MklMax) {  System.out.println("Anda memmiliki kelebihan Sks, silahkan " + "ulangi program.");  System.out.print("Masukkan kode mata kuliah : ");  matkul[a][0] = in.nextLine();  System.out.print("Masukkan Nama mata kuliah : ");  matkul[a][1] = in.nextLine();  System.out.print("Masukkan Sks mata kuliah : ");  matkul[a][2] = in.nextLine();  Sks = Integer.valueOf(matkul[a][2]);  break;  }  if (TotalSKS == MklMax) {  break;  }  a++;  if (TotalSKS >= MklMin) {  System.out.println("Mata kuliah yang anda miliki telah memenuhi " + "mata kuliah minimal yang bisa diambil,tekan 0 untuk berhenti");  pilihan = in.nextInt();  if (pilihan == 0) {  break;  }  }  System.out.println("-----------------------------------------");  }  }  public int getMkMax() {  return MklMax;  }  public int getMkMin() {  return MklMin;  }  public void showData() {  System.out.printf("Nama\t : %s\nNim\t : %s\nJurusan\t : %s\nIP\t : " + "%.2f\n", getNama(), getNim(), getJurusan(), getIp());  System.out.println("Mata kuliah pilihan anda : ");  for (int b = 0; b <= a; b++) {  System.out.printf("%s\t%s\t\t%s\t\n", matkul[b][0], matkul[b][1], matkul[b][2]);  }  }  } |

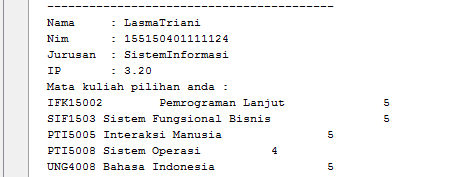
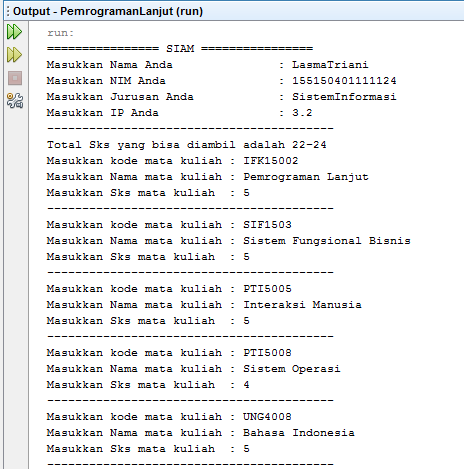
|  |  |
| --- | --- |
|  | MainMahasiswa.java |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38 | package pemrogramanlanjut;  import java.util.Scanner;  public class MainMahasiswa {  public static void main(String[] args) {  Scanner in = new Scanner(System.in);  System.out.println("================ SIAM ================");  Mahasiswa MHS[] = new Mahasiswa[3];  for (int i = 1; i <= MHS.length; i++) {  System.out.print("Masukkan Nama Anda\t\t : ");  String nama = in.next();  System.out.print("Masukkan NIM Anda\t\t : ");  String nim = in.next();  System.out.print("Masukkan Jurusan Anda\t\t : ");  String jurusan = in.next();  System.out.print("Masukkan IP Anda\t\t : ");  double ip = in.nextDouble();  MHS[i] = new Mahasiswa(nama, nim, jurusan, ip);  System.out.println("-----------------------------------------");  MHS[i].setTotalSKS();  MHS[i].setmatkul();  System.out.println("-----------------------------------------");  MHS[i].showData();  System.out.println("=========================================");  }  }  } |

1. **PEMBAHASAN**

|  |  |
| --- | --- |
|  | Mahasiswa.java |
| 1  3  5  7  8  9  11  13  15  16  17  18  21  22  25  26  29  30  33  34  37  38  41  44  47  50  56  57  58  60  61  62  63  64  65  66  67  69  70  72  74  76  77  79  80  82  83  84  87  88  90  91  93  96  97  101  106  107  110  111  114  115  118  120 | package pemrogramanlanjut digunakan untuk menghindari terjadinya kesamaan nama kelas yang sudah ada dalam java.  import java.util.Scanner digunakan untuk meload library Scanner yang berfungsi untuk menerima inputan dari user.  Deklarasi class dengan nama Mahasiswa.  Deklarasi instance variabel Nama, Nim, Jurusan dengan tipe String.  Deklarasi instance variabel Ip dengan tipe data double.  Deklarasi instance variabel MklMax, MklMin, TotalSKS, Sks dengan tipe data int.  Deklarasi instance variabel array dengan nama matkul dengan tipe data String.  Deklarasi constructor Mahasiswa dengan parameter Nama, Nim, Jurusan, Ip.  Memasukkan nilai parameter Nama kedalam instance variabel Nama.  Memasukkan nilai parameter Nim kedalam instance variabel Nim.  Memasukkan nilai parameter Jurusan kedalam instance variabel Jurusan.  Memasukkan nilai parameter Ip kedalam instance variabel Ip.  Deklarasi method getNama dengan tipe String.  Method getNama mengembaikan nilai ke variabel Nama.  Deklarasi method getNim dengan tipe String.  Method getNim mengembaikan nilai ke variabel Nim.  Deklarasi method getJurusan dengan tipe String.  Method getJurusan mengembaikan nilai ke variabel Jurusan.  Deklarasi method getIp dengan tipe String  Method getIp mengembaikan nilai ke variabel nilaiIp.  Deklarasi method void dengan nama setTotalSKS.  Seleksi kondisi apabila ip diatas 3 dan dibawah 4 maka bisa mengambil 22-24 sks.  Seleksi kondisi apabila ip diatas 2,5 dan dibawah 3 maka bisa mengambil 21-19 sks.  Seleksi kondisi apabila ip diatas 2 dan dibawah 2.5 maka bisa mengambil 18-16 sks.  Seleksi kondisi apabila ip diatas 1.5 dan dibawah 2 maka bisa mengambil 12-15 sks.  Seleksi kondisi apabila ip dibawah 1.5 maka bisa mengambil dibawah 11 sks.  Deklarasi method void setmatkul.  Deklarasi scanner dengan nama in.  Mencetak “Total Sks yang bisa diambil adalah” **+** nilai dari variabel getMkMin “**-**“ + nilai dari variabel getMkMax  Looping while dengan syarat sks yang diambil tidak lebih dari batas yang telah ditentukan.  Mencetak “Masukkan kode mata kuliah :”  Menyimpan inputan kode mata kuliah pada array matkul.  Mencetak “Masukkan Nama mata kuliah :”  Menyimpan inputan nama mata kuliah pada array matkul.  Mencetak “Masukkan Sks mata kuliah :”  Menyimpan inputan sks mata kuliah pada array matkul.  Mengkonversi data string pada array yang menyimpan sks mata kuliah ke tipe data integer.  Menambahkan nilai variabel Sks ke dalam variabel TotalSKS.  Seleksi kondisi apabila TotalSKS lebih dari batas maksimum.  Mencetak “Anda memiliki kelebihan Sks, silahkan ulangan program” apabila kelebihan Sks.  Mencetak “Masukkan kode mata kuliah :”  Menyimpan inputan kode mata kuliah pada array matkul.  Mencetak “Masukkan Nama mata kuliah :”  Menyimpan inputan nama mata kuliah pada array matkul.  Mencetak “Masukkan Sks mata kuliah :”  Menyimpan inputan sks mata kuliah pada array matkul.  Mengkonversi data string pada array yang menyimpan sks mata kuliah ke tipe data integer.  Break.  Seleksi kondisi apabila TotalSKS sama dengan batas maksimum.  Break.  Increment a.  Seleksi kondisi apabila TotalSKS memenuhi batas minimum.  Mencetak “Mata kuliah yang anda miliki telah memenuhi mata kuliah minimal yang bisa diambil, tekan 0 untuk berhenti”.  Menyimpan inputan ke variabel pilihan.  Seleksi kondisi apabila pilihan bernilai 0 maka program akan di break.  Mencetak “-------------------------------------------------------------------“.  Deklarasi method getMkMax dengan tipe data integer.  Method getMkMax mengembaikan nilai ke variabel MklMax.  Deklarasi method getMkMin dengan tipe data integer.  Method getMkMin mengembaikan nilai ke variabel MklMin.  Deklarasi metod showData untuk menampilkan data mahasiswa tersebut.  Mencetak data mahasiswa.  Mencetak “Mata kuliah pilihan anda :”  Mencetak data mata kuliah pilihan mahasiswa. |

|  |  |
| --- | --- |
|  | MainMahasiswa.java |
| 1  3  5  7  8  9  11  12  14  15  16  17  18  19  20  21  23  24  27  28  32  34 | package pemrogramanlanjut digunakan untuk menghindari terjadinya kesamaan nama kelas yang sudah ada dalam java.  import java.util.Scanner digunakan untuk meload library Scanner yang berfungsi untuk menerima inputan dari user.  Deklarasi class dengan nama MainMahasiswa.  Deklarasi main method.  Deklarasi scanner dengan variabel in.  Perintah mencetak “=============== SIAM ================”.  Deklarasi array MHS menggunakan class Mahasiswa dengan jumlah 3.  Deklarasi looping untuk mengisi data mahasiswa.  Perintah mencetak “Masukkan Nama Anda: “.  Menyimpan inputan user kedalam variabel nama.  Perintah mencetak “Masukkan NIM Anda: “.  Menyimpan inputan user kedalam variabel nim.  Perintah mencetak “Masukkan Jurusan Anda: “.  Menyimpan inputan user kedalam variabel jurusan.  Perintah mencetak “Masukkan IP Anda: “.  Menyimpan inputan user kedalam variabel ip.  Inisialisasi nilai MHS[i] dengan menggunakan variabel nama, nim, jurusan, ip sebagai parameter.  Perintah mencetak “--------------------------------------“.  Memanggil metod setTotalSKS untuk mengetahui jumlah SKS yang bsia diambil.  Memanggil method setMatkul untuk mengisi variabel Matkul.  Memanggil method showData untuk menampilkan data keseluruhan tentang mahasiswa tersebut.  Perintah mencetak “========================================”. |

1. **SCREENSHOT PROGRAM**

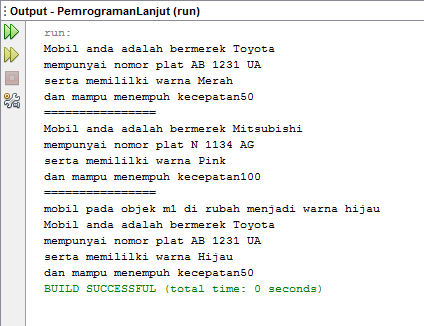


1. **PRAKTIKUM**
2. Class
3. Apakah yang disebut dengan variabel instance dan lokal variabel? Jelaskan perbedaanya!

**Jawab:** Instance Variable: Variable yang dideklarasikan di dalam class. Lokal Variable: Variable yang dideklarasikan di dalam method.

1. Lakukan percobaan diatas dan benahi jika menemukan kesalahan!

**Jawab:** Telah dilakukan dan dibenahi.



1. Rubah kode pada mainMobil diatas menjadi proses meminta masukan dari user dan buat menjadi interaktif!

**Jawab:**

public static void main(String[] args) {

Scanner input = new Scanner (System.in);

System.out.print("Jumlah Mobil : ");

int jum = input.nextInt();

for (int i=1; i<=jum; i++) {

System.out.print("Masukkan Kecepatan : ");

int kec=input.nextInt();

Mobil m = new Mobil();

m.setKecepatan(kec);

System.out.print("Manufaktur : ");

String man= input.next();

m.setManufaktur(man);

System.out.print("Masukkan Warna : ");

String warna = input.next();

m.setWarna(warna);

System.out.print("Masukkan No Plat : ");

String plat= input.next();

m.setNoPlat(plat);

System.out.println("Waktu Tempuh : ");

double wktu=input.nextDouble();

m.setWaktu(wktu, kec);

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

m.displayMessage();

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

}

}

1. Tambahkan method pada class mobil bernama setWaktu yang berparameter double, yang kemudian disimpan pada variabel waktu! (Ketetuannya adalah user harus menginputkan dalam satuan jam)

**Jawab:**

private double waktu;

public void setWaktu (double i, double j){

this.waktu=i;

this.rubahSekon(i);

this.hitungJarak(i, j );

}

1. Tambahkan method bernama rubahSekon mempunyai parameter bertipe double dan hanya dapat dipanggil pada class mobil. Method ini memiliki fungsi untuk merubah masukan user yaitu jam menjadi sekon. Method tersebut di panggil pada method setWaktu dengan nilai parameter adalah nilai dari variabel parameter method setWaktu!

**Jawab:**

public void rubahSekon(double i){

this.sekon=i \* 3600;

}

1. Tambahkan method pada class mobil dan hanya dapat dipanggil pada class mobil bernama rubahKecepatan yang mempunyai fungsi untuk merubah format kecepatan yang awalnya km/h menjadi m/s. Dipanggil di method setKecepatan!

**Jawab:**

public void rubahKecepatan(double i){

this.kec= (i\*1000)/3600;

}

1. Tambahkan method pada class mobil bernama hitungJarak yang mempunyai aksi untuk menghitung jarak yang dapat di tempuh oleh mobil dengan rumus jarak = kecepatan \* waktu!

**Jawab:**

public void hitungJarak(double i, double a){

this.jarak=i \* a;

}

1. Tambahkan informasi jarak yang dapat ditempuh pada method displayMessage kemudian rubah satuannya yang awalnya m (meter) menjadi km (kilometer)!

**Jawab:**

public void displayMessage() {

System.out.println("Mobil anda adalah bermerek " + manufaktur);

System.out.println("mempunyai nomor plat " + noPlat);

System.out.println("serta memiliki warna " + warna);

System.out.println("dan mampu menempuh kecepatan " + kecepatan+" km/h");

System.out.println("dan mampu menempuh kecepatan " + kec+" m/s");

System.out.println("mampu menempuh jarak "+kec\*sekon+" meter");

System.out.println("mampu menempuh jarak "+jarak+" Kilometer");

}

1. Mahasiswa A ingin menulis pada sebuah buku tulis yang ingin dia miliki, isi lembar buku tersebut adalah 50 lembar. Setiap harinya ia menulis sebanyak 100 kata perhari yang cukup untuk 1/2 halaman buku. Buatlah rumus untuk menghitung berapa lama ia menghabiskan 1 buku tersebut serta identifikasilah objek, dan karakteristiknya kemudian implementasikan dalam bentuk class.

**Jawab:**

**Class Buku**

public class Buku {

private int LembarKertas;

public void setLembarKertas(int lmbr){

LembarKertas=lmbr;

}

public int jumlahHari(){

int hari = (int)((LembarKertas\*2)/0.5);

return hari;

}

public void showHari(){

System.out.println("Lama hari yang diperlukan untuk menulis 1 buku : "+jumlahHari());

}

}

**Class BukuTest**

public class BukuTest {

public static void main(String[] args) {

Buku buku1 = new Buku();

buku1.setLembarKertas(50);

buku1.showHari();

}

}



1. **KESIMPULAN**
2. OOP (Object Oriented Programming) adalah suatu metode pemrograman yang berorientasi kepada objek. Tujuan dari OOP diciptakan adalah untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari.
3. Cara menginstansiasi objek adalah dengan cara menggunakan operator new dan memasukkan instance-nya ke dalam variabel referensi yang baru saja dideklarasikan.

**Contoh:** Mahasiswa Mhs1 = new Mahasiswa();

1. Pengertian dari
2. Class

Class merupakan suatu blueprint atau cetakan untuk menciptakan suatu instant dari object. class juga merupakan grup suatu object dengan kemiripan attributes/properties, behaviour dan relasi ke object lain.

1. Object

Object adalah instance dari class. Jika class secara umum merepresentasikan (template) sebuah object, sebuah instance adalah representasi nyata dari class itu sendiri.

1. Atribut

Atribut adalah karateristik yang dimiliki oleh objek. Dibagian itu kita dapat mendeklarasikan atribut-atribut yang dibutuhkan kelas untuk membentuk objek.

1. Behavior

Behavior adalah tingkah laku yang dapat dilakukan oleh objek, behavior adalah kelakuan atau sifat dari Object seperti Mahasiswa tadi bisa memilih mata kuliah dan sebagainya.