1. **DEFINISI MASALAH**
2. 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.
3. **SOURCE CODE**

**Kode Program Siam**

**Kode Utama Siam**

|  |  |
| --- | --- |
| 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 | package OOPpraktikumBAB1;  import java.util.Scanner;  public class siam1 {  public static void main(String[] args) {  Scanner in = new Scanner (System.in);    //orang pertama  siam2 sam = new siam2();    System.out.println("=<SISTEM INFORMASI AKADEMIK MAHASISWA>=");  System.out.print("Nama Mahasiswa : ");  sam.nama = in.nextLine();  System.out.print("NIM : ");  sam.nim = in.nextInt();  System.out.print("Jurusan : ");  sam.jur = in.nextLine();  sam.aturData(sam.nama, sam.nim, sam.jur);    System.out.println("===============================");  System.out.print("Jumlah Mata Kuliah : ");  sam.jum = in.nextInt();    for (int i=0 ; i<sam.jum ; i++){  System.out.print("Mata Kuliah : ");  sam.mk = in.next();  System.out.print("Kode : ");  sam.code = in.nextInt();  }  sam.aturMataKuliah(sam.mk, sam.code);    System.out.print("IP : ");  sam.ip = in.nextFloat();  System.out.print("SKS : ");  sam.sks = in.nextInt();  sam.aturIP(sam.ip,sam.sks);    //orang kedua  siam2 sam2 = new siam2();    System.out.println("=<SISTEM INFORMASI AKADEMIK MAHASISWA");  System.out.print("Nama Mahasiswa : ");  sam2.nama = in.nextLine();  System.out.print("NIM : ");  sam2.nim = in.nextInt();  System.out.print("Jurusan : ");  sam2.jur = in.next();  sam2.aturData(sam2.nama, sam2.nim, sam2.jur);    System.out.println("===============================");  System.out.print("Jumlah Mata Kuliah : ");  sam2.jum = in.nextInt();    for (int i=0 ; i<sam2.jum ; i++){  System.out.print("Mata Kuliah : ");  sam2.mk = in.next();  System.out.print("Kode : ");  sam2.code = in.nextInt();  }  sam2.aturMataKuliah(sam2.mk, sam2.code);    System.out.print("IP : ");  sam2.ip = in.nextFloat();  System.out.print("SKS : ");  sam2.sks = in.nextInt();  sam2.aturIP(sam2.ip,sam2.sks);    //orang ketiga  siam2 sam3 = new siam2();    System.out.println("=<SISTEM INFORMASI AKADEMIK MAHASISWA");  System.out.print("Nama Mahasiswa : ");  sam3.nama = in.nextLine();  System.out.print("NIM : ");  sam3.nim = in.nextInt();  System.out.print("Jurusan : ");  sam3.jur = in.next();  sam3.aturData(sam3.nama, sam3.nim, sam3.jur);    System.out.println("===============================");  System.out.print("Jumlah Mata Kuliah : ");  sam3.jum = in.nextInt();    for (int i=0 ; i<sam3.jum ; i++){  System.out.print("Mata Kuliah : ");  sam3.mk = in.next();  System.out.print("Kode : ");  sam3.code = in.nextInt();  }  sam.aturMataKuliah(sam3.mk, sam3.code);    System.out.print("IP : ");  sam3.ip = in.nextFloat();  System.out.print("SKS : ");  sam3.sks = in.nextInt();  sam3.aturIP(sam3.ip,sam3.sks);  }  } |

**Kode Kelas Siam**

|  |  |
| --- | --- |
| 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 | package OOPpraktikumBAB1;  public class siam2 {  String nama, jur, mk;  int nim, jum, code, sks;  float ip;    public void aturData(String name, int NIM, String jurusan){  nama = name;  nim = NIM;  jur = jurusan;  }    public void aturMataKuliah(String MatKul, int kode){  mk = MatKul;  code = kode;  }    public void aturIP(float IP, int SKS){  ip = IP;  sks = SKS;  if ( IP >= 3.0 && SKS >= 22 ){  System.out.println("data anda benar");  }  else if (IP >= 2.5 && IP < 3.0 && SKS < 22){  System.out.println("data anda benar");  }  else if (IP >= 2.0 && IP < 2.5 && SKS < 19){  System.out.println("data anda benar");  }  else if (IP >= 1.5 && IP < 2.0 && SKS < 16){  System.out.println("data anda benar");  }  else if (IP < 1.5 && SKS < 12){  System.out.println("data anda benar");  }  else  System.out.println("data anda salah");  }  } |

1. **PEMBAHASAN**

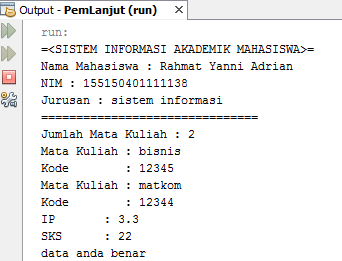
**Kode Utama Siam**

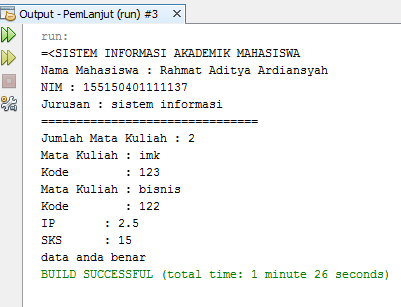
|  |  |  |
| --- | --- | --- |
| 1  2  3  4  5  7  8  10  12  13  14  15  16  17  18  20  21  22  24  25  26  27  28  30  32  33  34  35  36 | Deklarasi Package dengan nama OOPpraktikumBAB1  Melakukan import dari class Scanner  Deklarasi kelas dengan nama Siam1  Deklarasi Main Method  Instansiasi in sebagai kelas Scanner  Keterangan orang pertama  Instansiasi objek bernama sam sebagai kelas Siam2  Mencetak kalimat "=<SISTEM INFORMASI AKADEMIK MAHASISWA>="  Mencetak “Nama Mahasiswa : “  Menyimpan inputan sebagai nama dengan menggunakan objek sam  Mencetak “NIM : ”  Menyimpan inputan sebagai nim dengan menggunakan objek sam  Mencetak “Jurusan : “  Menyimpan inputan sebagai jur dengan menggunakan objek sam  Method aturData dari class siam2 yang diisi dengan inputan nama, nim, jur  Mencetak "==============================="  Mencetak "Jumlah Mata Kuliah : "  Menyimpan inputan sebagai jum dengan menggunakan objek sam  Membuat perulangan for sebanyak inputan yang dimasukkan  Mencetak "Mata Kuliah : "  Memasukkan inputan ke dalam mk dengan menggunakan objek sam  Mencetak "Kode : "  Memasukkan inputan ke dalam code dengan menggunakan objek sam  Method aturMataKuliah dari class siam2 yang diisi dengan inputan mk dan code  Mencetak “IP :”  Memasukkan inputan ke dalam ip dengan menggunakan objek sam  Menceak “SKS : “  Memasukkan inputan ke dalam sks dengan menggunakan objek sam  Method aturIP dari class siam2 yang diisi dengan inputan ip dan sks |  |

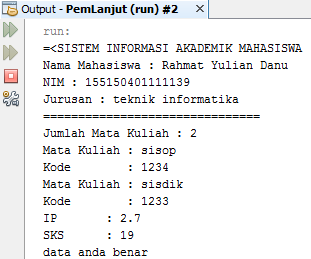
**Kode Kelas Siam**

|  |  |
| --- | --- |
| 1  3  4  5  6  8  10  11  12  15  16  17  20  21  22  23  24  26  29  32  35  38 | Deklarasi Package dengan nama OOPpraktikumBAB1  Deklarasi kelas dengan nama Siam2  Deklarasi nama, jur, mk dengan String  Deklarasi nim, jum, code, sks dengan int  Deklarasi ip dengan float  Method aturData yang sebelumnya sudah dimasukkan inputan pada class siam1  Menyamakan deklarasi nama dari class siam2 dengan inputan yang ada di method aturData  Menyamakan deklarasi nim dari class siam2 dengan inputan yang ada di method aturData  Menyamakan deklarasi jur dari class siam2 dengan inputan yang ada di method aturData  Method aturMataKuliah yang sebelumnya sudah dimasukkan inputan pada class siam1  Menyamakan deklarasi mk dari class siam2 dengan inputan yang ada di method aturMataKuliah  Menyamakan deklarasi code dari class siam2 dengan inputan yang ada di method aturMataKuliah  Method aturIP yang sebelumnya sudah dimasukkan inputan pada class siam1  Menyamakan deklarasi ip dari class siam2 dengan inputan yang ada di method aturIP  Menyamakan deklarasi sks dari class siam2 dengan inputan yang ada di method aturIP  Membuat seleksi kodisi dimana ip >= 3.0 maka dapat memilih beban sks lebih dari sama dengan 22  Mencetak “data anda benar"  Membuat seleksi kodisi dimana 2.5 >= ip < 3.0 maka dapat memilih beban sks kurang dari 22  Membuat seleksi kodisi dimana 2.0>= ip < 2.5 maka dapat memilih beban sks kurang dari 19  Membuat seleksi kodisi dimana 1.5 >= ip < 2.0 maka dapat memilih beban sks kurang dari 16  Membuat seleksi kodisi dimana ip < 1.5 maka dapat memilih beban sks kurang dari 12  Jika data tidak sesuai maka akan mencetak ”data anda salah” |

1. **SCREENSHOT PROGRAM**

**

**

**

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

**Jawab :**

Variabel instance adalah variabel yang dideklarasikan di dalam kelas namun berada di luar method, konstruktor, maupun blok statement. Variabel lokal adalah variable yang di deklarasikan di dalam method, konstruktor ataupun blok statement.

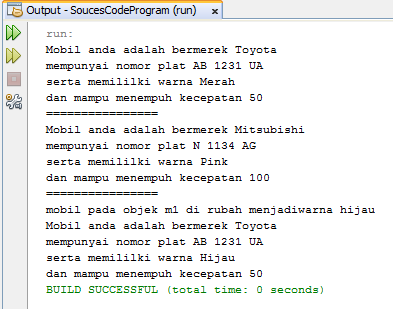
Perbedaan antara variabel instance dan lokal vriabel :

* Letak variabel isntance berada di luar method, konstruktor, maupun blok statement sedangkan variabel lokal method, konstruktor, maupun blok statement.
* Instance variabel dapat menggunakan access modifier seperti private, public, protected, dan default sedangkan lokal variabel tidak.
* Instance sudah memiliki nilai defaultnya sedangkan local variabel tidak mempunyai default value, jadi kita harus mendeklarasikan dan memberi nilai sebelum variabel itu digunakan .
* Instance variabel dibuat ketika objek dibuat yaitu menggunkan keyword “new”. Instace variabel dihancurkan ketika sebuah objek dihancurkan sedangkan variabel lokal tidak.

1. Lakukan percobaan diatas dan benahi jika menemukan kesalahan!

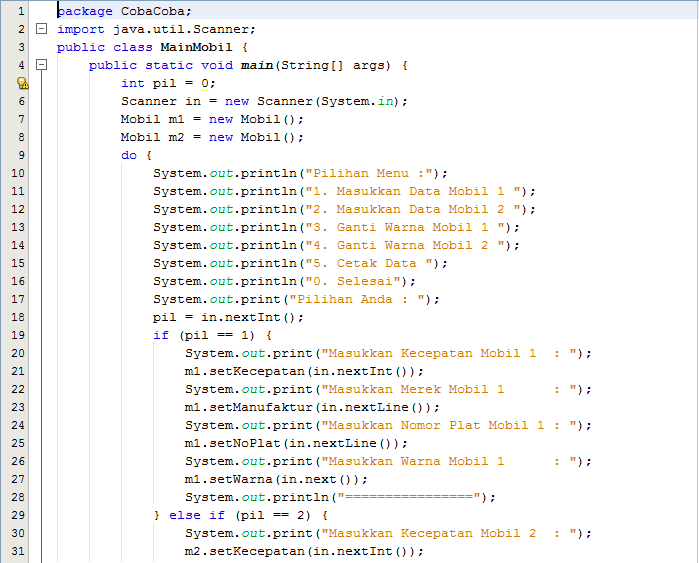
**Jawab :**

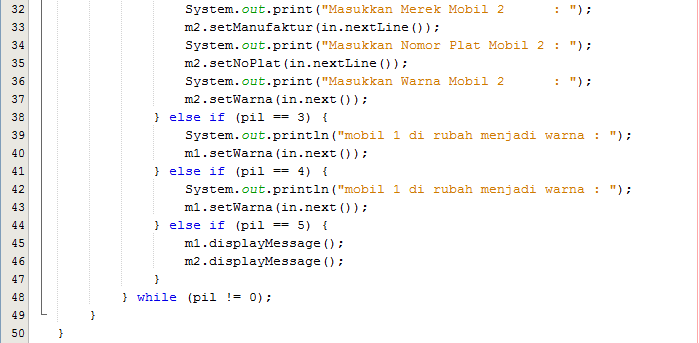
Tida ditemukan adanya kesalahan, program dapat berjalan dengan baik dan menghasilkan outputan seperti dibawah ini :



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

**Jawab :**





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 :**



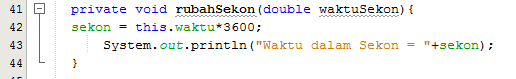
1. Tambahkan method bernama rubahSekon mempunyai parameter bertipe double dan hanya dapat dipanggil pada class mobil. Method ini memiliki fungsi untuk merubah masukan user yaitu jam menjadi sekon. Method tersebut di panggil pada method setWaktu dengan nilai parameter adalah nilai dari variabel parameter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

**Jawab :**



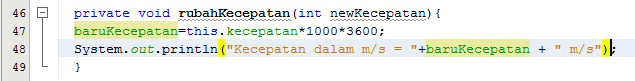
1. Tambahkan method pada class mobil dan hanya dapat dipanggil pada class mobil bernama rubahKecepatan yang mempunyai fungsi untuk merubah format kecepatan yang awalnya km/h menjadi m/s. Dipanggil di method setKecepatan!

**Jawab :**



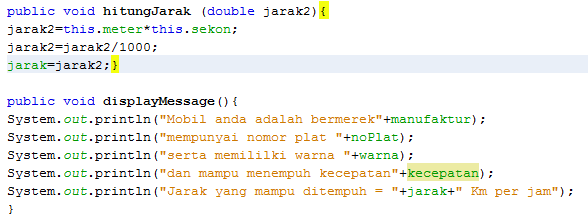
1. Tambahkan method pada class mobil bernama hitungJarak yang mempunyai aksi untuk menghitung jarak yang dapat di tempuh oleh mobil dengan rumus jarak = kecepatan \* waktu!

**Jawab :**



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

**Jawab :**



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 :**

Kelasnya adalah Buku dan objeknya adalah mahasiswa yang menulis buku tersebut. Dan dia memiliki karakteristik menulis ½ halaman perhari di dalm buku yang berisi 50 lembar. Maka dapat di buat programnya sebagai berikut.

Source Code di Kelas Buku :

public class Buku {

public int lmbr=50;

public int jmlhHal;

public int getJmlhHal(){

return jmlhHal=2\*lmbr;

}

}

Source Code di Kelas Mahasiswa :

public class Mahasiswa {

public double jmlhHari;

public double halamanAkanHabis = 0.5;

public double hitungwaktu(int hlman){

jmlhHari = hlman \* halamanAkanHabis \* 2;

return jmlhHari;

}

}

Source Code di Kelas TestBuku:

public class TestBuku {

public static void main(String[] args) {

Buku bobo=new Buku();

Mahasiswa mahasiwa1=new Mahasiswa();

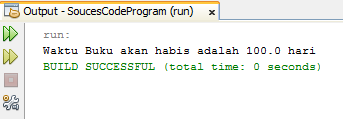
System.out.print("Waktu Buku akan habis adalah ");

System.out.println(mahasiwa1.hitungwaktu(bobo.getJmlhHal())+" hari");

}

}

Maka jika kelas TestBuku dijalankan akan didapati hasil sebagai berikut :



1. **KESIMPULAN**
2. Jelaskan apa yang dimaksud dengan OOP !

**Jawab :**

OOP atau Object Orinted Programming atau jika diterjemahkan dalam bahasa Indonesia berarti pemrograman berorientasi program. OOP merupakan suatau cara berfikir untuk mendefenisikan suatu objek dalam pemrogroman. Dengan adanya OOP kita dapat lebih mudah dalam mengembangkan program dengan cara mengiikuti model yang sudah ada dalam kehidupan sehari-hari. Bagian dari suatu permasalahan merupakan sebuah objek dan objek bisa dibagi lagi menjadi objek yang lebih kecil lagi. Contohnya, Pesawat, Pesawat adalah sebuah objek. Pesawat itu sendiri terbentuk dari beberapa objek yang lebih kecil lagi seperti mesin, roda, baling-baling, kursi, dll. Pesawat sebagai objek yang terbentuk dari objek-objek yang lebih kecil saling berhubungan, berinteraksi, berkomunikasi dan saling mengirim pesan kepada objek-objek yang lainnya. Begitu juga dengan program, sebuah objek yang besar dibentuk dari beberapa objek yang lebih kecil, objek-objek itu saling berkomunikasi, dan saling berkirim pesan kepada objek yang lain.

1. Bagaimana cara menginstansiasi ocject ?

**Jawab :**

Pada saat kita mendefinisikan suatu kelas, kita hanya membuat tipe data baru, bukan membuat objek baru.Untuk membuat objek baru dari tipe kelas yang telah didefinisikan, secara eksplisit kita perlu melakukan dua tahap.

* Pertama, kita perlu mendeklarasikan variabel yang digunakan sebagai referensi ke objek dari kelas bersangkutan.
* Kedua, kita perlu menginstansiasi kelas dengan menggunakan operator new dan memasukkan instance-nya ke dalam variabel referensi yang baru saja dideklarasikan.

Berikut ini ilustrasi dua tahap proses tersebut :

// Mendeklarasikan variabel k bertipe Kotak

Kotak k;  
// Melakukan instansiasi dan memasukkan referensi ke variabel k  
k = new Kotak();

Baris pertama (Tahap I) berguna untuk mendeklarasikan variabel k yang berperan sebagai referensi ke objek dari tipe Kotak. Pada tahap ini, variabel k akan bernilai null, yang berarti referensi tersebut belum menunjuk ke objek aktual manapun. Segala bentuk usaha untuk menggunakan variabel k disaat seperti ini akan menyebabkan kesalahan pada saat proses kompilasi program. Alasannya adalah karene variabel k masih bernilai null. Selanjutnya baris kedua (Tahap II) digunakan untuk mengalokasikan sebuah objek baru dan memasukkan referensinya ke dalam variabel k. Pada prakteknya, dua tahap di atas biasanya ditulis dalam satu baris, seperti berikut :

// Mendeklarasikan variabel k bertipe Kotak dan  
// Melakukan instansiasi dan memasukkan referensi ke variabel k  
Kotak k = new Kotak();

1. Apa yang dimaksusd dengan Class, Object, attribute, dan behavior/aksi ?

**Jawab :**

* 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. Contoh : Class Person, Vehicle, Tree, Fruit dan lain-lain.
* Object adalah instance dari class. Jika class secara umum mepresentasikan (template) sebuah object, sebuah instance adalah representasi nyata dari class itu sendiri. Contoh : Dari class Fruit kita dapat membuat object Mangga, Pisang, Apel, dan lain-lain.
* Attribute merupakan nilai data yang terdapat pada suatu object yang berasal dari class. Attributes merepresentasikan karakteristik dari suatu object.
* Behavior/aksi merupakan suatu operasi berupa fungsi-fungsi yang dapat dikerjakan oleh suatu object. Method didefinisikan pada class akan tetapi dipanggil melalui object. Contoh, pada object mangga : terdapat method ambilRasa , kupasKulit dan lainlain