**DEFINISI MASALAH**

1. Tugas Anda adalah membuat sebuah class yang memuat data-data pada buku alamat.

Tabel berikut mendefinisikan informasi yang dimiliki oleh buku alamat.

|  |  |
| --- | --- |
| Attribut | Deskripsi |
| Nama | Nama Lengkap perseorangan |
| Alamat | Alamat Lengkap |
| Nomor Telepon | Nomor telepon personal |
| Alamat E-mail | Alamat E-mail personal |

Tabel 1: Atribut dan Deskripsi Atribut

Buat implementasi dari method sebagai berikut :

1. Menyediakan accessor dan mutator method terhadap seluruh atribut
2. Constructor

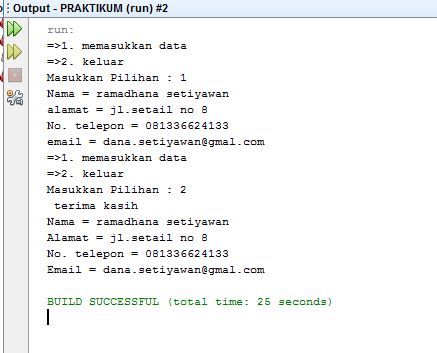
**SOURCE CODE**

|  |  |  |
| --- | --- | --- |
| buku | | |
| 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 | public class buku {  private String nama;  private String alamat;  private String telepon;  private String email;  public String getNama() {  return nama;  }  public void setNama(String nama) {  this.nama = nama;  }  public String getAlamat() {  return alamat;  }  public void setAlamat(String alamat) {  this.alamat = alamat;  }  public String getTelepon() {  return telepon;  }  public void setTelepon(String telepon) {  this.telepon = telepon;  }  public String getEmail() {  return email;  }  public void setEmail(String email) {  this.email = email;  }  public void display(){  System.out.println("Nama = "+getNama());  System.out.println("Alamat = "+ getAlamat());  System.out.println("No. telepon = "+ getTelepon());  System.out.println("Email = " + getEmail());  System.out.println(" ");  }  } | |
|  | | |
| Testbuku | | |
| 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 | | import java.util.Scanner;  public class Testbuku {  public static void main(String[] args) {  int pilihan;  Scanner in = new Scanner(System.in);  Scanner input = new Scanner(System.in);  Scanner inp = new Scanner(System.in);  Scanner masuk = new Scanner(System.in);  Scanner ina = new Scanner(System.in);  buku bu = new buku();  do {  System.out.println("=>1. memasukkan data");  System.out.println("=>2. keluar");  System.out.print("Masukkan Pilihan : ");  pilihan = inp.nextInt();  switch (pilihan) {  case 1:  System.out.print("Nama = ");  bu.setNama(in.nextLine());  System.out.print("alamat = ");  bu.setAlamat(input.nextLine());  System.out.print("No. telepon = ");  bu.setTelepon(ina.nextLine());  System.out.print("email = ");  bu.setEmail(masuk.nextLine());  break;  case 2:  System.out.println(" terima kasih ");  }  } while (pilihan != 2);  bu.display();  }  } |

**PEMBAHASAN**

|  |  |
| --- | --- |
| buku | |
| 1  2  3  4  5  6  7  9  10  12  13  15  16  18  19  21  22  24  25  27  28  30  31  32  33  34  35 | Deklarasi kelas dengan nama buku  Variable nama bertipe data String dengan access modifier private  Variable alamat bertipe data String dengan access modifier private  Variable telepon bertipe data String dengan access modifier private  Variable email bertipe data String dengan access modifier private  Deklarasi method getNama beracces modifier public  Pengembaian nilai ke method getNama dari variable Nama  Deklarasi method setNama beraccess modifier public  Pengisian variable nama  Deklarasi method getAlamat beracces modifier public  Pengembaian nilai ke method getAlamat dari variable Alamat  Deklarasi method setNama beraccess modifier public  Pengisian variable nama  Deklarasi method getTelepon beracces modifier public  Pengembaian nilai ke method getTelepon dari variable Telepon  Deklarasi method setTelepon beraccess modifier public  Pengisian variable Telepon  Deklarasi method getEmail beracces modifier public  Pengembaian nilai ke method getEmail dari variable Email  Deklarasi method setEmail beraccess modifier public  Pengisian variable Email  Deklarasi method display bertipe void beracces modifier public  Mencetak “nama = “ dan method getNama  Mencetak “Alamat = “ dan method getAlamat  Mencetak “No. Telepon =” dan method getTelepon  Mencetak “Email = “ dan method getEmail  Mencetak “ “ sebagai space |
|  | |
| Testbuku | |
| 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  30  31 | Memasukkan program Scanner ke java  Nama kelas Testbuku  Method yang digunakan bertipe void  Variable pilihan bertipe data interger  Program Scanner dengan inisialisasi in  Program Scanner dengan inisialisasi input  Program Scanner dengan inisialisasi inp  Program Scanner dengan inisialisasi masuk  Program Scanner dengan inisialisasi ina  Mendeklrasikan sebuah objek bernama bu dari kelas buku  Memulai perulangan do  Mencetak “=>1. Memasukkan data”  Mencetak “=>2. Keluar”  Mencetak “=>Masukkan Pilihan :”  Memasukkan variable pilihan dengan cara di Scanner  Operasi Swtich Case berparameter pilihan  Case 1  Mencetak “ Nama = “  Memanggil method setNama dan memasukkan data dengan cara di Scanner  Mencetak “ alamat = “  Memanggil method setAlamat dan memasukkan data dengan cara di Scanner  Mencetak “No. Telepon “  Memanggil method setTelepon dan memasukkan data dengan cara di Scanner  Mencetak “ email = “  Memanggil method setEmail dan memasukkan data dengan cara di Scanner  Mengakhiri case 1  Memulai case 2  Mencetak “terima kasih”  Program akan di ulang jika variabel pilihan tidak sama dengan 2  Memanggil method display dari class buku |

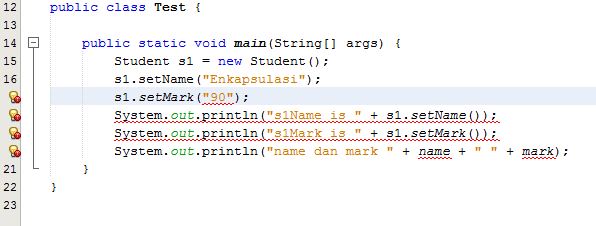
**SCREENSHOT PROGRAM**

****

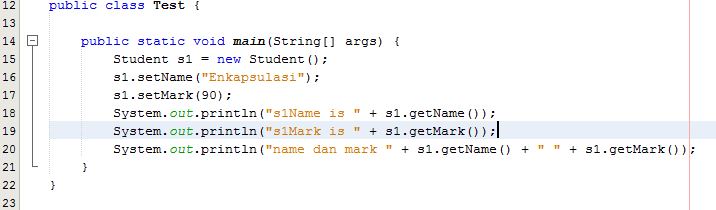
**PRAKTIKUM**

1. Encapsulation 1
2. Lakukan percobaan diatas dan benahi jika menemukan kesalahan!

Jawab :

Terdapat kesalahan yaitu pada :

Dan yang seharusnya adalah :



1. Jika pada baris 6 s1.setName diubah menjadi s1.getName apa yang terjadi?jelaskan!

Jawab :

Program akan berjalan dan tidak ada eror karena s1.getName akan menghasilkan outputnya.

1. Lakukan perubahan pada baris 5 dengan menghilangkan String di tanda dalam kurung lalu ubah baris 7 seperti no.3 apa yang terjadi? Jelaskan!

Jawab :

Program akan berjalan karena tanda String itu membuat program itu eror.

1. Setelah diperbaiki, ubahlah hak akses pada baris 4(pada class Student)menjadi private apa yang terjadi jika class Test dijalankan? Jelaskan!

Jawab :

jika pada pada baris ke 4(pada class Student) diganti menjadi private program akan eror dan class Test tidak dapat dijalankan karena private hanya bisa diakses oleh kelas itu sendiri

1. Jika kedua kelas diatas terdapat dalam package yang sama apakah konsep enkapsulasi tetap berfungsi? Jelaskan!

Jawab :

Konsep enkapsulasi akan terus berfungsi karena konsep itu bertujuan untuk mencegah suatu proses program yang berjalan secara sembarangan atau di intervensi oleh program lain. Untuk menjaga kebutuhan program agar dapat diakses sewaktu –waktu, sekaligus untuk menjaga program tersebut.

1. **Encapsulation 2**
2. Method apakah yang menjadi accessor (getter)?

Jawab :

Method yang menjadi accessor(getter) adalah method yang digunakan untuk membaca nilai variabel pada class, baik berupa instance maupun static.

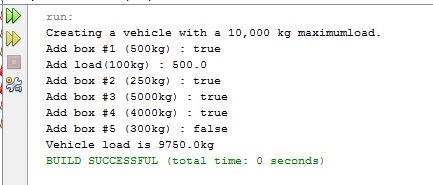
1. Tambahkan source berikut dibawah baris ke 6 pada class TestVehicle1.

System.out.println(“Add load(100kg); “ +(vehicle.load=500));

Jalankan program, apakah output dari proram tersebut?

Kembalikan program seperti semula.

Jawab :



1. Ubahlah tipe data pada atribut load dan maxload pada class Vehicle1 menjadi **public**.

Jalankan program, apakah output dari program tersebut?

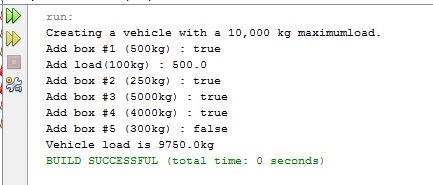
1. Tambahkan source kode berikut dibawah baris ke 6 pada class TestVehicle1.

System.out.println("Add load(100kg) : " + (vehicle.load=500));

Jalankan program, apakah output dari program tersebut?

Kembalikan program seperti semula.

Jawab :



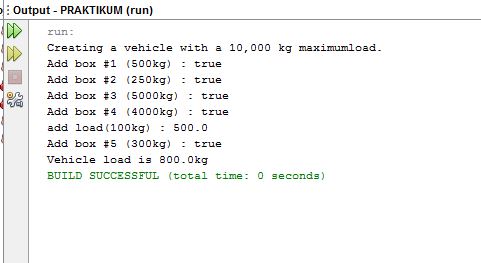
1. Tambahkan source kode berikut dibawah baris ke 12 pada class TestVehicle1.

System.out.println(“add load(100kg) : “ +(vehicle.load=500));

Jalankan program, apakah output dari program tersebut?

Kembalikan program seperti semula.

Jawab :



1. Ulangi instruksi pada nomer 4 dengan mengubah tipe data pada atribut load dan maxload pada class vehicle1 menjadi protected.

Jawab :

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | public class Vehicle1 {  protected double load, maxLoad;  public Vehicle1(double max) {  this.maxLoad = max;  }  public double getLoad() {  return this.load;  }  public double getMaxLoad() {  return this.maxLoad;  }  public boolean addBox(double weight) {  double temp = 0.0D;  temp = this.load + weight;  if (temp <= maxLoad) {  this.load = this.load + weight;  return true;  } else {  return false;  }  }  } |

1. Ulangi instruksi pada nomer 4 dengan mengubah tipe data pada atribut load dan maxload pada class vehicle1 menjadi default.

Jawab :

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | public class Vehicle1 {  default double load, maxLoad;  public Vehicle1(double max) {  this.maxLoad = max;  }  public double getLoad() {  return this.load;  }  public double getMaxLoad() {  return this.maxLoad;  }  public boolean addBox(double weight) {  double temp = 0.0D;  temp = this.load + weight;  if (temp <= maxLoad) {  this.load = this.load + weight;  return true;  } else {  return false;  }  }  } |

**KESIMPULAN**

1. Enkapsulasi adalah suatu konsep proses untuk menyembunyikan atau pembungkus untuk menjaga suatu proses agar tidak dapat diakses secara sembarangan atau di interval oleh program lain. Konsep ini sangat penting untuk menjaga kebutuhan program agar dapat diakses sewaktu – waktu, sekaligus menjaga program tersebut.
2. Saat program java ingin diciptakan, user menginginkan beberapa data dilarang untuk diakses, implementasi inilah sering disebut access modifiers. Access Modifiers berguna untuk memberikan hak akses kepada user. Tetapi tidak semua data yang berada di dalam suatu kelas, atau turunannya dapat diakses, karena memmpunyai batasan – batasan yang berlaku.
3. terdapat 4 macam Acces Modifiers, yaitu :
4. Akses Default adalah tipe class dalam package yang sama yang memiliki hak akses terhadap variabel dan methods dalam class
5. Akses Public adalah class member untuk diakses baik dari dalam dan luar.
6. Akses Private pengaksesan class hanya dapat diakses oleh class dimana tipe ini dibuat.
7. Akse Protected adalah mengijinkan class member untuk diakses oleh method dalam class tersebut dan elemen-elemen subclass.