1. **DEFINISI MASALAH**

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 | NamaLengkapperseorangan |
| Alamat | AlamatLengkap |
| NomorTelepon | Nomortelepon personal |
| Alamat E-Mail | Alamat E-Mail personal |

Buat implementasi dari method sebagai berikut :

1. Menyediakan accessor dan mutator method terhadap seluruh atribut

2. Constructor

1. **SOURCE CODE**

|  |  |
| --- | --- |
| 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 | **Class encapsulation**  package pbo;  public class encapsulation {  private String nomer;  private String Nama;  private String Alamat;  private String noTelp;  private String Email;  private String NoKTP;  public encapsulation(String nomer, String Nama, String Alamat, String noTelp, String Email, String NoKTP) {  this.nomer = nomer;  this.Nama = Nama;  this.Alamat = Alamat;  this.noTelp = noTelp;  this.Email = Email;  this.NoKTP = NoKTP;  }  public void setNomer(String nomer) {  this.nomer = nomer;  }  public void setNama(String Nama) {  this.Nama = Nama;  }  public void setAlamat(String Alamat) {  this.Alamat = Alamat;  }  public void setEmail(String Email) {  this.Email = Email;  }  public void setNoTelp(String noTelp) {  this.noTelp = noTelp;  }  public String getNomer() {  return nomer;  }  public String getNama() {  return Nama;  }  public String getEmail() {  return Email;  }  public String getAlamat() {  return Alamat;  }  public String getNoKTP() {  return NoKTP;  }  public void setNoKTP(String noKTP) {  this.NoKTP = NoKTP;  }  public String getNoTelp() {  return noTelp;  }  } |

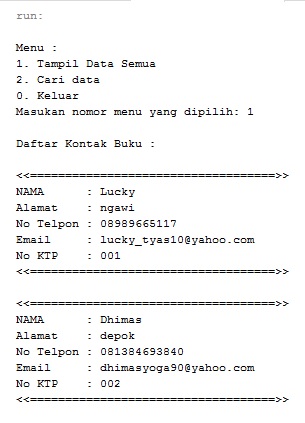
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  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  125  126  127  128  129  130  131  132  133  134  135  136  137138  139  140  141  142 | **Class main**  package pbo;  import java.util.ArrayList;  import java.util.Scanner;  public class main {  private static int navigasi;  public static void main(String[] args) {  ArrayList<encapsulation> book = new ArrayList<>();  book.add(0, new encapsulation("A001","Lucky", "ngawi", "08989665117", "lucky\_tyas10@yahoo.com", "001"));  book.add(1, new encapsulation("A002","Dhimas", "depok", "081384693840", "dhimasyoga90@yahoo.com", "002"));  book.add(2, new encapsulation("A003","Rangga", "bogor", "081318713647", "ranggakurniasandi@yahoo.com", "003"));  menu(book);  }  private static void menu(ArrayList<encapsulation> book) {  Scanner input = new Scanner(System.in);  do {  System.out.println("\nMenu : \n1. Tampil Data Semua\n2. Cari data\n0. Keluar");  System.out.print("Masukan nomor menu yang dipilih: ");  navigasi = input.nextInt();  System.out.println("");  switch (navigasi) {  case 1:  showAllKontak(book);  break;  case 2:  searchData(book);  break;  case 0:  System.out.println("Program terminated ! ");  break;  }  } while (navigasi != 3);  }  private static void showAllKontak(ArrayList<encapsulation> book) {  System.out.println("Daftar Kontak Buku :\n");  for (int i = 0; i < book.size(); i++) {  System.out.println("<<===================================>>");  System.out.println("NAMA : " +book.get(i).getNama());  System.out.println("Alamat : " + book.get(i).getAlamat());  System.out.println("No Telpon : " + book.get(i).getNoTelp());  System.out.println("Email : " + book.get(i).getEmail());  System.out.println("No KTP : " + book.get(i).getNoKTP());  System.out.println("<<===================================>>");  System.out.println("");  }  }  private static void searchData(ArrayList<encapsulation> book) {  Scanner input = new Scanner(System.in);  String Email;  String Nama;  String noKTP;  boolean checkEmail = false;  boolean checkNama = false;  boolean checknoKTP = false;  int kodeNama = -1;  int kodeEmail = -1;  int kodeKTP = -1;  int kodeObjek = -1;  int i = 0;  do {  do {  System.out.print("\nMasukan Nama : ");  Nama = input.next();  for (int j = 0; j < book.size(); j++) {  if (Nama.equalsIgnoreCase(book.get(j).getNama()) == true) {  checkNama = true;  kodeNama = j;  break;  }  }  if (checkNama == false) {  System.out.println("Alamat email tidak ada yang cocok , Ulangi lagi !");  }  } while (checkNama != true);  do {  System.out.print("Masukan Alamat email : ");  Email = input.next();  for (int j = 0; j < book.size(); j++) {  if (Email.equalsIgnoreCase(book.get(j).getEmail()) == true) {  checkEmail = true;  kodeEmail = j;  break;  }  }  if (checkEmail == false) {  System.out.println("Alamat email tidak ada yang cocok , Ulangi lagi !");  }  } while (checkEmail != true);  do {  System.out.print("Masukan No KTP : ");  noKTP = input.next();  for (int j = 0; j < book.size(); j++) {  if (noKTP.equalsIgnoreCase(book.get(j).getNoKTP()) == true) {  checknoKTP = true;  kodeKTP = j;  break;  }  }  if (checknoKTP == false) {  System.out.println("No KTP tidak ada yang cocok , Ulangi lagi !");  }  } while (checknoKTP != true);    do {  if (book.get(i).getNama() == book.get(kodeNama).getNama() && book.get(i).getEmail() == book.get(kodeEmail).getEmail() && book.get(i).getNoKTP() == book.get(kodeKTP).getNoKTP()) {  kodeObjek = i;  break;  }  i++;  } while (i < book.size());  if (kodeObjek != -1) {  getKontak(book, kodeObjek);  break;  } else {  i = 0;  System.out.println("Maaf kombinasi antara Nama , email dan no KTP tidak sesuai , ulangi lagi !");  }  } while (kodeObjek == -1);  }  private static void getKontak(ArrayList<encapsulation> book, int i) {  System.out.println("\nPencarian................");  System.out.println("\nInfo Kontak yang anda cari : ");  System.out.println("No.\t: " + book.get(i).getNomer());  System.out.println("NAMA\t: " + book.get(i).getNama());  System.out.println("Alamat\t: " + book.get(i).getAlamat());  System.out.println("No. Telpon: " + book.get(i).getNoTelp());  System.out.println("Email\t: " + book.get(i).getEmail());  System.out.println("No KTP\t: " + book.get(i).getNoKTP());  }  } |

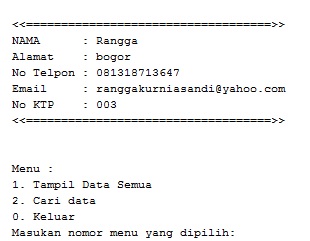
1. **PEMBAHASAN**

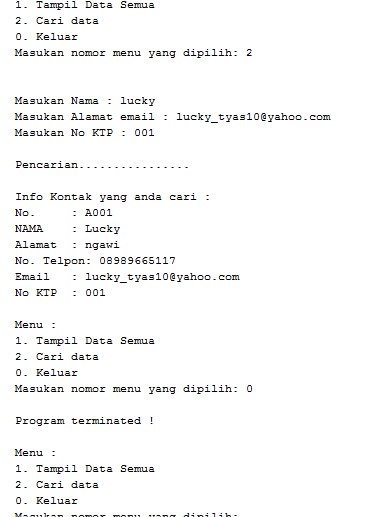
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66 | **Class encapsulation**  Deklarasi package Deklarasi kelas  Deklarasi variabel private bernama nomer dengan tipe String  Deklarasi variabel private bernama Nama dengan tipe String  Deklarasi variabel private bernama Alamat dengan tipe String  Deklarasi variabel private bernama noTelp dengan tipe String  Deklarasi variabel private bernama Email dengan tipe String  Deklarasi variabel private bernama NoKTP dengan tipe String  Blank space  Konstruktor YellowPage  Deklarasi variabel nomer dengan kata kunci this  Deklarasi variabel Nama dengan kata kunci this  Deklarasi variabel Alamat dengan kata kunci this  Deklarasi variabel noTelp dengan kata kunci this  Deklarasi variabel Email dengan kata kunci this  Deklarasi variabel NoKTP dengan kata kunci this  End of method  Blank space  Method void bernama setNomer berparameter nomer tipe String  Inisialisasi variabel nomer dengan kata kunci this  End of method  Blank space  Method void bernama setNama berparameter Nama tipe String  Inisialisasi variabel Nama dengan kata kunci this  End of method  Blank space  Method void bernama setAlamat berparameter Alamat tipe String  Inisialisasi variabel Alamat dengan kata kunci this  End of method  Blank space  Method void bernama setEmail berparameter Email tipe String  Inisialisasi variabel Email dengan kata kunci this  End of method  Blank space  Method void bernama setNoTelp berparameter NoTelp tipe String  Inisialisasi variabel NoTelp dengan kata kunci this  End of method  Blank space  Method void bernama getNomer  Pengembalian nilai nomer  End of method  Blank space  Method void bernama getNama  Pengembalian nilai Nama  End of method  Blank space  Method void bernama getEmail  Pengembalian nilai Email  End of method  Blank space  Method void bernama getAlamat  Pengembalian nilai Alamat  End of method  Blank space  Method void bernama getNoKTP  Pengembalian nilai NoKTP  End of method  Blank space  Method void bernama setNoKTP berparameter NoKTP tipe String  Inisialisasi variabel NoKTP dengan kata kunci this  End of method  Blank space  Method void bernama getNoTelp  Pengembalian nilai NoTelp  End of method  End of class |

|  |  |
| --- | --- |
| 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  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143 | **Class main**  Deklarasi package Melakukan impurt dari ArrayList  Melakukan import dari package Scanner  Deklarasi kelas  Deklarasi variabel private static int bernama navigasi  Blank space  Main method  Blank space  Instasiasi ArrayList YellowPage  Blank space  Membuat array dari konstruktor YellowPage ("A001","Richa", "Pekalongan", "1234", "RichaAmalia@gmail.com", "1147-001")  Membuat array dari konstruktor YellowPage ("A002","Gogot", "Sidoarjo", "4321", "Gogot\_Keren@gmail.com", "1147-002")  Membuat array dari konstruktor YellowPage ("A003","Dika", "Indonesia", "6669", "Dika\_Keren@gmail.com", "1147-003")  Blank space  Memanggil method menu  End of method  Blank space  Method static void bernama menu dengan parameter ArrayList pada YellowPage  Memberikan perintah pada java agar user dapat mengisi nilai variable yang dideklarasikan  Perintah do  Mencetak “Menu:   1. Tampil Data Semua 2. Cari Data 3. Keluar”   Mencetak “Masukkan nomor menu yang dipilih:”  Memasukkan navigasi yang diinputkan oleh user  Mencetak enter  Switch-case untuk variabel navigasi  Case 1  Memanggil method showAllKontak  Perintah selesai  Case 2  Memanggil searchData  Perintah selesai  Case 3  Mencetak “Program terminated !”  Perintah selesai  End of switch-case  End of do dan perintah while apabila navigasi = 3  End of method  Blank space  Method showAllKontak berparameter ArrayList pada YellowPage  Mencetak “Daftar Kontak Buku: “  Looping  Mencetak “===============================”  Mencetak “NAMA:” dan memanggil method getNama  Mencetak “Alamat:” dan memanggil method getAlamat  Mencetak “No Telpon:” dan memanggil method getNoTelp  Mencetak “Email:” dan memanggil method getEmail  Mencetak “No KTP:” dan memanggil method getNoKTP  Blank space  Mencetak “===============================”  Mencetak enter  End of looping  End of method  Blank space  Method searchData berparameter ArrayList pada YellowPage  Memberikan perintah pada java agar user dapat mengisi nilai variable yang dideklarasikan  Deklarasi variabel Email tipe String  Deklarasi variabel Nama tipe String  Deklarasi variabel NoKTP tipe String  Deklarasi variabel checkEmail = false tipe boolean  Deklarasi variabel checkNama = false tipe boolean  Deklarasi variabel checknoKTP = false tipe boolean  Deklarasi variabel kodeNama = -1 tipe int  Deklarasi variabel kodeEmail = -1 tipe int  Deklarasi variabel kodeKTP = -1 tipe int  Deklarasi variabel kodeObjek = -1 tipe int  Deklarasi variabel i = 0 tipe int  Blank space  Perintah do  Perintah do  Mencetak “Masukan Nama: “  Memasukkan nama yang diinputkan oleh user  Looping  Kondisi jika Nama sama dengan getNama = true  Inisialisasi checkNama=true  Inisialisasi kodeNama = j  Perintah selesai  End of if  End of looping  Kondisi jika checkNama = false  Mencetak “Nama tidak ada yang cocok, Ulangi lagi!”  End of if  End of do dan perintah while jika checkNama= true  Blank space  Peintah do  Mencetak “Masukan Alamat email:”  Memasukkan email yang diinputkan oleh user  Looping  Kondisi jika Email sama dengan getEmail = true  Inisialisasi checkEmail = true  Inisialisasi kodeEmail = j  Perintah selesai  End of looping  End of if  Kondisi jika checkEmail = false  Mencetak “ Alamat email tidak ada yang cocok, Ulangi lagi!”  End of if  End of do dan perintah while jika checkNama= true  Blank space  Peintah do  Mencetak “Masukan No KTP:”  Memasukkan noKTP yang diinputkan oleh user  Looping  Kondisi jika noKTP sama dengan getNoKTP = true  Inisialisasi checknoKTP = true  Inisialisasi kodeKTP = j  Perintah selesai  End of looping  End of if  Kondisi jika checknoKTP = false  Mencetak “No KTP tidak ada yang cocok, Ulangi lagi!”  End of if  End of do dan perintah while jika checkNama= true  Blank space  Perintah do  Kondisi jika get (i) getNama = get(kodeNama)getNama &(i) getEmail = get(kodeEmail)getEmail & get(i) getNoKTP = get(kodeKTP)getNoKTP  Inisialisasi kodeObjek = i  Perintah selesai  End of if  Increment i  End of do dan perintah while jika i kurang dari book.size  Blank space  Kondisi jika kodeObjek tidak samadengan -1  Memanggil getKontak  Perintah selesai  Kondisi lain  Inisialisasi I = 0  Mencetak “Maaf kombinasi antara Nama, email dan no KTP tidak sesuai, ulangi lagi!”  End of if  End of do dan perintah while jika kodeObjek tidak sama dengan -1  End of method  Blank space  Method getKontak berparameter ArrayList pada YellowPage dan int i  Mencetak “Pencarian……………..”  Mencetak “Info Kontak yang anda cari: “  Mencetak “No. : “ dan memanggil get(i) pada book dan getNomer  Mencetak “NAMA : “ dan memanggil get(i) pada book dan getNAMA  Mencetak “Alamat: “ dan memanggil get(i) pada book dan getAlamat  Mencetak “No. Telpon“ dan memanggil get(i) pada book dan getNoTelp  Mencetak “Email: “ dan memanggil get(i) pada book dan getEmail  Mencetak “No. KTP: “ dan memanggil get(i) pada book dan getNoKTP  End of method  End of class |

**SCREENSHOT PROGRAM**



****

****

1. **PRAKTIKUM**
2. Encapsulation 1
3. Lakukan percobaan di atas dan benahi jika menemukan kesalahan!

Jawab:

|  |  |
| --- | --- |
|  | public static void main(String[] args) {  Student s1 = new Student();  s1.setName("Enkapsulasi");  s1.setMark(90);//harusnya tanda petik dihilangkan  System.out.println("s1Name is " + s1.getName());//menggunakan getname bukan setname  System.out.println("s1Mark is " + s1.getMark());//menggunakan getname bukan setname  System.out.println("name dan mark " + s1.getName()+ " " + s1.getMark());//pemangilanharusdisertaiobjek |

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:

* Pada baris 5 program tidak akan error jika dihilangkan tanda petiknya. Karena tipe data yang digunakan adalah integer.Jika menggunakan tanda petik program error karena tanda petik digunakan hanya untuk tipe data String.
* Jikapadabaris 7 diubah, makaakanmenampilkannilai variable berupaangka yang diinputkanpadabarisnomer 5.

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

Jawab:

Terjadi error. Karena sebelumnya Name sudah menggunakan access modifier private, sehingga untuk mengubah nilainya seharusnya menggunakan access modifier public.

1. Jika kedua kelas diatas terdapat dalam package yang sama apakah konsep enkapsulasitetap berfungsi? jelaskan!

Jawab: Dapat tetap berfungsi. Karena encapsulasi bisa terjadi pada package yang sama maupun paket berbeda, sehinggga penggunaan method mutator dan accessor yang memiliki modifier public yang artinya bisa diakses di semua class dalam package yang sama

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

Jawab:

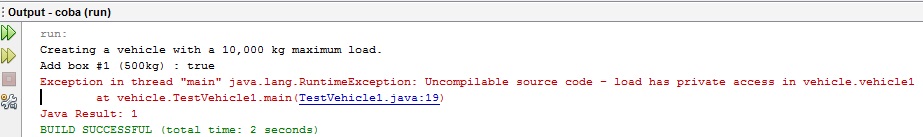
|  |  |
| --- | --- |
|  | public double getLoad(){  return this.load;  } |

1. Tambahkan source code 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:

Terjadi error



1. Ubahlah tipe data pada atribut load dan maxload pada class Vehicle1 menjadi public.Jalankan program, apakah output dari program tersebut?

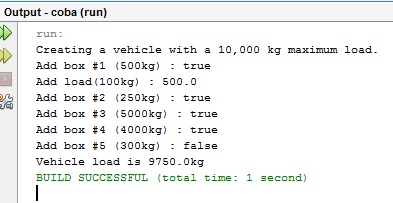
a. Tambahkan source kode berikut dibawah baris ke 6 pada class TestVehicle1.

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

Jalankan program, apakah output dari program tersebut?

Kembalikan program seperti semula.

Jawab:

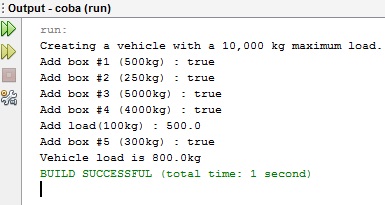


b. Tambahkan source kode berikut dibawah baris ke 12 pada class TestVehicle1.

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

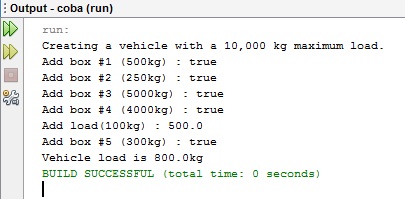
Jalankan program, apakah output dari program tersebutKembalikan program seperti semula.

Jawab:



1. Ulangi instruksi pada nomer 4 dengan mengubah tipe data pada atribut load dan maxloadpada class Vehicle1 menjadi protected.

Jawab:



1. Ulangi instruksi pada nomer 4 dengan mengubah tipe data pada atribut load dan maxload pada class Vehicle1 menjadi **default.**

Jawab:

Default modifier adalah modifier yang tidak tertulis sehingga nampak seperti tidak terdapat modifier. Saat program diberikan modifier default program akan tetap berjalan karena modifier default hanya menspesifikasikan kelas-kelas di paket yang sama yang bisa mengakses variable dan method suatu kelas.

1. **KESIMPULAN**

* Enkapsulasi adalah suatucarauntukmenyembunyikaninformasi detail darisuatu class.
* 4 tipe access modifier :
  + Default: Tipe akses ini hanya dalam satu class itu sendiri yang dapat mengaksesnya tergantung dari blok kodenya
  + Public: Tipe ini mengijinkan seluruh class dari luar atau dari dalam package bisa mengaksesnya
  + Protected: Tipe ini di gunakan untuk pewarisan dalam java, jadi kalau suatu super class mempunyai sebuah member, ini dapat diakses oleh subclassnya.
  + Private: Tipe ini hanya dapat diakses dimana class tersebut di buat.
* Package: Sarana/cara pengelompokkan dan pengorganisasian kelas-kelas dan interface yang sekelompok menjadi suatu unit tunggal dalam library.

Subclass: Kelas induk yang diturunkan disebut dengan superclass. Sedangkan kelas baru hasil turunan disebut subclass.