1. **DEFINISI MASALAH**
2. Buatlah program dengan menggunakan class untuk menghitung penjumalahan, pengurangan, perkalian dan pembagian. Method penjumlahan dan pengurangan menggunakan static method sedangkan sisanya menggunakan method non static. Tambahkan method bertipe non static bernama Sederhana untuk menyederhanakan sebuah pecahan.
3. CV. Labkomdas adalah perusahaan yang bergerak di bidang produksi jaket, terdapat 3 tipe jaket yang memiliki harga yang tetap (final) yaitu jaket dengan bahan A, B dan C yang masing-masing memiliki harga persatuannya adalah Rp 100.000, Rp 125.000, Rp 175.000. Pabrik memberi diskon bagi pembeli yang membeli banyak jenis jaket, diskon di berikan apabila :

* Jaket A di beli sebanyak lebih dari 100 buah maka harga menjadi 95.000 per biji.
* Jaket B di beli sebanyak lebih dari 100 buah maka harga menjadi 120.000 per biji.
* Jaket C di beli sebanyak lebih dari 100 buah maka harga menjadi 160.000 per biji.

Buat program dari kasus diatas dengan materi yang sudah di ajarkan oleh asisten

NB : nilai final adalah harga Jaket A, B dan C.

1. **SOURCE CODE**

**Masalah 1**

|  |  |
| --- | --- |
| **Perhitungan.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 | import java.util.Scanner;  public class Perhitungan {  private double bilangan1;  private double bilangan2;  private static double hasil;  public static Perhitungan p = new Perhitungan();  public void setBilangan1(double bilangan1) {  this.bilangan1 = bilangan1;  }  public void setBilangan2(double bilangan2) {  this.bilangan2 = bilangan2;  }  public double getBilangan1() {  return bilangan1;  }  public double getBilangan2() {  return bilangan2;  }  public double getHasil() {  return hasil;  }  public static double penjumlahan(double b1,double b2) {  hasil=b1+b2;  Perhitungan.hasil=hasil;  return hasil;  }  public static double pengurangan(double b1,double b2) {  hasil=b1-b2;  Perhitungan.hasil=hasil;  return hasil;  }  public double perkalian(double b1,double b2) {  hasil=b1\*b2;  Perhitungan.hasil=hasil;  return hasil;  }  public double pembagian(double b1,double b2) {  hasil=b1/b2;  Perhitungan.hasil=hasil;  return hasil;  }  public void cetakHasil(double hasil,String tanda) {  System.out.println(this.bilangan1 + " " + tanda + " " +  this.bilangan2 + " = " + hasil);  }  public double sederhana(double hasil) {  hasil = Math.round(hasil \* 100.0) / 100.0;  return hasil;  }  static int BilPecahan(int pembilang, int penyebut) {  int s = 2, t = 2;  int x = pembilang, y = penyebut;  while (x != y) {  if (x > y) {  while ((pembilang % s) != 0) {s++;}  x = pembilang / s;  s++;  } else {  while ((penyebut % t) != 0) {t++;}  y = penyebut / t;  t++;  }  }  return (x);  }  public void sederhana() {  Scanner scan = new Scanner(System.in);  System.out.print("Pembilang = ");  int pembilang1 = scan.nextInt();  System.out.print("Penyebut = ");  int penyebut1 = scan.nextInt();  int pembilang = BilPecahan(pembilang1, penyebut1);  int pembilang2 = pembilang1 / pembilang;  int penyebut2 = penyebut1 / pembilang;  System.out.println(pembilang1+"/"+penyebut1+" =  "+pembilang2+"/"+penyebut2+"\n");  }  } |
| **Main.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 | import java.util.\*;  public class Main {  public static Scanner scan = new Scanner(System.in);  public static Perhitungan pht = new Perhitungan();  public static void main(String[] args) {  String loop = "y";  do {  cetakMenu();  switch (scan.nextInt()) {  case 0: loop="n";break;  case 1:  input();  Perhitungan.penjumlahan(pht.getBilangan1(),  pht.getBilangan2());  pht.cetakHasil(pht.getHasil(),"+");  sederhanakan();  if (scan.next().equals("y")) {  pht.cetakHasil(pht.sederhana(pht.getHasil()),"+");  }  System.out.println("");  break;  case 2:  input();  Perhitungan.pengurangan(pht.getBilangan1(),  pht.getBilangan2());  pht.cetakHasil(pht.getHasil(),"-");  sederhanakan();  if (scan.next().equals("y")) {  pht.cetakHasil(pht.sederhana(pht.getHasil()), "-");  }  System.out.println("");  break;  case 3:  input();  pht.perkalian(pht.getBilangan1(),  pht.getBilangan2());  pht.cetakHasil(pht.getHasil(),"x");  sederhanakan();  if (scan.next().equals("y")) {  pht.cetakHasil(pht.sederhana(pht.getHasil()), "x");  }  System.out.println("");  break;  case 4:  input();  pht.pembagian(pht.getBilangan1(),  pht.getBilangan2());  pht.cetakHasil(pht.getHasil(),":");  sederhanakan();  if (scan.next().equals("y")) {  pht.cetakHasil(pht.sederhana(pht.getHasil()), ":");  }  System.out.println("");  break;  case 5:  pht.sederhana();  break;  default:  System.out.println("Pilihan tidak ada");  break;  }  } while (loop.equals("y"));  }  public static void cetakMenu() {  System.out.println("PERHITUNGAN");  System.out.println("1. Penjumlahan");  System.out.println("2. Pengurangan");  System.out.println("3. Perkalian");  System.out.println("4. Pembagian");  System.out.println("5. Penyederhanaan Pecahan");  System.out.println("0. Keluar");  System.out.print("Pilih:");  }  public static void input() {  System.out.print("Masukkan bilangan 1 = ");  pht.setBilangan1(scan.nextDouble());  System.out.print("Masukkan bilangan 2 = ");  pht.setBilangan2(scan.nextDouble());  }  public static void sederhanakan() {  System.out.print("sederhanakan(y/n)?");  }  } |

**Masalah 2**

|  |  |
| --- | --- |
| **PabrikJaket.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 | public class PabrikJaket {  private final int jaketA = 100000;  private final int jaketB = 125000;  private final int jaketC = 175000;  private final static PabrikJaket pj = new PabrikJaket();  private String[][] keranjang = new String[100][4];  public int getJaketA() {  return jaketA;  }  public int getJaketB() {  return jaketB;  }  public int getJaketC() {  return jaketC;  }  public void pilihan(String pilihan) {  System.out.print("\nHarga satuan Rp.");  if (pilihan.equals("a")) {  System.out.print(pj.getJaketA());  } else if (pilihan.equals("b")) {  System.out.print(pj.jaketB);  } else {  System.out.print(pj.getJaketC());  }  System.out.print("\nApabila membeli lebih dari 100 biji");  System.out.print("\nHarga satuan menjadi ");  if (pilihan.equals("a")) {  System.out.print(pj.getJaketA() - 5000);  } else if (pilihan.equals("b")) {  System.out.print(pj.jaketB - 5000);  } else {  System.out.print(pj.getJaketC() - 15000);  }  }  public int pembelian(String pilihan, int jumlah) {  int total = 0;  if (jumlah > 0 && jumlah <= 100) {  if (pilihan.equals("a")) {  total = pj.getJaketA() \* jumlah;  } else if (pilihan.equals("b")) {  total = pj.getJaketB() \* jumlah;  } else {  total = pj.getJaketC() \* jumlah;  }  } else if (jumlah > 100) {  if (pilihan.equals("a")) {  total = (pj.getJaketA() - 5000) \* jumlah;  } else if (pilihan.equals("b")) {  total = (pj.getJaketB() - 5000) \* jumlah;  } else {  total = (pj.getJaketC() - 15000) \* jumlah;  }  }  return total;  }  public void keranjang(String pilihan, int total, int jumlah,  int x, int bayar) {  keranjang[x][0] = pilihan;  keranjang[x][1] = Integer.toString(jumlah);  keranjang[x][2] = Integer.toString(total);  keranjang[x][3] = Integer.toString(bayar);  }  public String[][] getKeranjang(){  return keranjang;  }  public static void cetakStruk(String[][] keranjang) {  int y = 0;  System.out.println("Jenis\tQty Harga");  System.out.println("------------------");  for (int x = 0; x < keranjang.length; x++) {  if (keranjang[x][0] == null) {  break;  }  System.out.print("Jaket " + keranjang[x][0] + "\t");  System.out.print(" " + keranjang[x][1]);  System.out.println(" Rp." + keranjang[x][2]);  y = x;  }  System.out.println("------------------");  System.out.println("Total = Rp." + keranjang[y][3]);  }  } |

|  |  |
| --- | --- |
| **Main.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 | import java.util.\*;  public class Main {  private static PabrikJaket pabrikjaket = new PabrikJaket();  private static Scanner scan = new Scanner(System.in);  public static void main(String[] args) {  String loop="y";  int jumlah=0,total=0,x=0,bayar=0;  do{  cetakMenu();  switch(scan.nextByte()){  case 0:loop="n";break;  case 1:  pabrikjaket.pilihan("a");  cetakJumlah("A");  jumlah=scan.nextInt();  System.out.println("");  total=pabrikjaket.pembelian("a",jumlah);  bayar = bayar + total;  pabrikjaket.keranjang("A", total,jumlah,x,bayar);  x++;  break;  case 2:  pabrikjaket.pilihan("b");  cetakJumlah("B");  jumlah=scan.nextInt();  System.out.println("");  total=pabrikjaket.pembelian("b",jumlah);  bayar = bayar + total;  pabrikjaket.keranjang("B", total,jumlah,x,bayar);  x++;  break;  case 3:  pabrikjaket.pilihan("c");  cetakJumlah("C");  jumlah=scan.nextInt();  System.out.println("");  total=pabrikjaket.pembelian("c",jumlah);  bayar = bayar + total;  pabrikjaket.keranjang("C", total,jumlah,x,bayar);  x++;  break;  case 4:  System.out.println("\nKERANJANG BELI");  PabrikJaket.cetakStruk(pabrikjaket.getKeranjang());  System.out.print("Beli(y/n):");  if(scan.next().equals("y")){  System.out.println("Pembelian Berhasil");loop="n";  System.out.println("\n\nSTRUK PEMBELIAN");  PabrikJaket.cetakStruk(pabrikjaket.getKeranjang());  System.out.println("TERIMA KASIH\n\n");  }  System.out.println("");  break;  default:  System.out.println("Pilihan tidak ada\n");  break;  }  } while(loop=="y");  }  public static void cetakMenu(){  System.out.println("MENU");  System.out.println("Beli Jaket");  System.out.println("1. Bahan A");  System.out.println("2. Bahan B");  System.out.println("3. Bahan C");  System.out.println("4. Lihat Keranjang");  System.out.println("0. Keluar");  System.out.print("Pilih: ");  }  public static void cetakJumlah(String pilihan){  System.out.print("\nJumlah jaket "+pilihan+" yang anda  beli: ");  }  } |

1. **PEMBAHASAN**

**Masalah 1**

|  |  |
| --- | --- |
| **Perhitungan.java** | |
| 1  2  3  4  5  6  7  8  10  11  13  14  16  17  19  20  22  23  24  25  27  28  29  30  32  33  34  35  37  38  39  40  42  43  46  47  48  50  51  52  53  54  55  56  57  58  59  60  61  64  66  67  68  69  70  71  72  73  74  75 | Memanggilan fungsi scanner dalam java.  Deklarasi class dengan nama Perhitungan.  Inisialisasi atribut bilangan1 bertipe data double dengan access modifier private.  Inisialisasi atribut bilangan2 bertipe data double dengan access modifier private.  Inisialisasi atribut hasil bertipe data double dengan access modifier private dan bertipe static.  Inisialisasi objek p dari class Perhitungan dengan tipe static dan access modifier public.  Deklarasi method setter setBilangan1 dari atribut bilangan1 dengan parameter variable bilangan1 bertipe data double.  Merubah nilai atribut bilangan1 dengan nilai dari parameter setBilangan1.  Deklarasi method setter setBilangan2 dari atribut bilangan2 dengan parameter variable bilangan2 bertipe data double.  Merubah nilai atribut bilangan2 dengan nilai dari parameter setBilangan2.  Deklarasi method getter getBilangan1 dari atribut bilangan1 bertipe data double.  Mengembalikan nilai attribut bilangan1 ke dalam method getter getBilangan2.  Deklarasi method getter getBilangan2 dari atribut bilangan2 bertipe data double.  Mengembalikan nilai attribut bilangan2 ke dalam method getter getBilangan2.  Deklarasi method getter getHasil dari atribut hasil bertipe data double.  Mengembalikan nilai attribut hasil ke dalam method getter getHasil.  Deklarasi method penjumlahan dengan pengembalian nilai bertipe data double dan bertipe static dengan access modifier public berparameter variable b1 dan b2 bertipe data double.  Pengisian atribut hasil dengan nilai b1 ditambah b2.  Pemanggilan attribut hasil dari class Perhitungan untuk diisi dengan variable hasil.  Pengembalian nilai kedalam method penjumlahan dari nilai variable hasil.  Deklarasi method pengurangan dengan pengembalian nilai bertipe data double dan bertipe static dengan access modifier public berparameter variable b1 dan b2 bertipe data double.  Pengisian atribut hasil dengan nilai b1 dikurangi b2.  Pemanggilan attribut hasil dari class Perhitungan untuk diisi dengan variable hasil.  Pengembalian nilai kedalam method pegurangan dari nilai variable hasil.  Deklarasi method perkalian dengan pengembalian nilai bertipe data double dan dengan access modifier public berparameter variable b1 dan b2 bertipe data double.  Pengisian atribut hasil dengan nilai b1 dikali b2.  Pemanggilan attribut hasil dari class Perhitungan untuk diisi dengan variable hasil.  Pengembalian nilai kedalam method perkalian dari nilai variable hasil.  Deklarasi method pembagian dengan pengembalian nilai bertipe data double dan dengan access modifier public berparameter variable b1 dan b2 bertipe data double.  Pengisian atribut hasil dengan nilai b1 dibagi b2.  Pemanggilan attribut hasil dari class Perhitungan untuk diisi dengan variable hasil.  Pengembalian nilai kedalam method pembagian dari nilai variable hasil.  Deklarasi method cetakHasil tanpa pengembalian dengan access modifier public berparameter variable hasil bertipe data double dan variable tanda bertipe data String.  Proses mencetak nilai dari bilangan1 diikuti nilai dari variable tanda kemudian nilai dari bilangan2 diikuti nilai dari variable hasil.  Deklarasi method sederhana dengan pengembalian nilai bertipe data double dengan access modifier public berparameter variable hasil bertipe data double.  Pengisian nilai varabile hasil dengan penghitungan dari fungsi Math.round untuk menyederhanakan pecahan.  Pengembalian nilai variable hasil kedalam method sederhana.  Dekalrasi method BilPecahan bertipe static dengan pengembalian data bertipe integer berparameter pembilang, penyebut bertipe data integer.  Deklarasi variable s,t bernilai 2 bertipe data integer.  Deklarasi variable x bernilai sama dengan pembilang dan y bernilai sama dengan penyebut bertipe data integer.  Inisialisasi perulangan while-do yang berulang ketika x tidak sama dengan y.  Pengecekan apakah niali dari x lebih dari y.  Inisialisasi perulangan while-do yang berulang apabila pembilang modulus s tidak berinilai 0 maka akan menambahkan niali s ditambah 1 (increment).  Perubahan nilai x menjadi pembilang dibagai s.  Penambahan nilai s dengan 1 (increment).  Fungsi yang dijalankan ketika fungsi if diatas tidak terpenuhi.  Inisialisasi perulangan while-do yang berulang apabila pembilang modulus t tidak berinilai 0 maka akan menambahkan niali t ditambah 1 (increment).  Perubahan nilai x menjadi penyebut dibagai t.  Penambahan nilai t dengan 1 (increment).  Proses pengembalian nilai x kedalam method BilPecahan.  Deklarasi method sederhana tanpa pengembalian nilai (void) dengan access modifier public.  Inisialisasi objek scan untuk inputan dari class Scanner.  Proses mencetak “Pembilang = “.  Inisialisasi variable pembilang1 bertipe data integer bernilai inputan dari user.  Proses mencetak “Penyebut = “.  Inisialisasi variable penyebut1 bertipe data integer bernilai inputan dari user.  Inisialisasi variable pembilang bernilai method BilPecahan berparameter nilai pembilang1 dan penyebut1.  Inisialisasi variable pembilang2 bernilai pembilang1 dibagi pembilang.  Inisialisasi variable penyebut2 berniali penyebut1 dibagi pembilang.  Proses mencetak nilai pembilang1/penyebut1 = pembilang2/penyebut2. |

|  |  |
| --- | --- |
| **Main.java** | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  15  16  17  18  20  21  22  23  24  26  27  28  29  31  32  33  34  35  37  38  39  40  42  43  44  45  46  48  49  50  51  53  54  55  56  57  58  59  60  62  64  65  66  67  68  69  70  71  72  74  75  76  77  78  80  81 | Proses memanggil semua class dari package java.util.  Deklarasi class dengan nama Main.  Deklarasi objek scan dari class Scanner untuk memanggil fungsi inputan dengan tipe static dengan access modifier public.  Deklarasi objek pht dari class Perhitungan bertipe static dan dengan access modifier public.  Main method java.  Deklarasi variable loop bertipe data String dengan nilai “y”.  Inisialisasi perulangan do-while.  Proses pemanggilan method cetakMenu.  Proses pengecekan inputan user bertipe data integer dengan fungsi switch.  Fungsi yang dijalankan apabila nilai inputan user bernilai 0. Apabila nilai switch bernilai 0 maka nilai dari variable loop akan dirubah menjadi “n”. Case 0 diakhiri.  Fungsi yang dijalankan apabila nilai dari inputan user bernilai 1.  Proses memanggil method input.  Pemanggilan method static penjumlahan dari class Perhitungan dengan parameter method getBilangan1 dan getBilangan2 dari objek pht.  Proses memanggil method cetakHasil dari objek pht dengan parameter nilai method getHasil dari objek pht dan “+”.  Proses memanggil method sederhanakan.  Pengecekan kondisi apakah nilai dari inputan user bernilai “y”.  Proses memanggil method cetakHasil dari objek pht dengan parameter method sederhana dari objek pht berparameter method getHasil dari objek pht dan “+”.  Proses mencetak “ ” (enter).  Fungsi untuk mengakhiri case1.  Fungsi yang dijalankan apabila nilai dari inputan user bernilai 2.  Proses memanggil method input.  Pemanggilan method static pengurangan dari class Perhitungan dengan parameter method getBilangan1 dan getBilangan2 dari objek pht.  Proses memanggil method cetakHasil dari objek pht dengan parameter nilai method getHasil dari objek pht dan “-”.  Proses memanggil method sederhanakan.  Pengecekan kondisi apakah nilai dari inputan user bernilai “y”.  Proses memanggil method cetakHasil dari objek pht dengan parameter method sederhana dari objek pht berparameter method getHasil dari objek pht dan “-”.  Proses mencetak “ ” (enter).  Fungsi untuk mengakhiri case2.  Fungsi yang dijalankan apabila nilai dari inputan user bernilai 3.  Proses memanggil method input.  Pemanggilan method static perkalian dari class Perhitungan dengan parameter method getBilangan1 dan getBilangan2 dari objek pht.  Proses memanggil method cetakHasil dari objek pht dengan parameter nilai method getHasil dari objek pht dan “x”.  Proses memanggil method sederhanakan.  Pengecekan kondisi apakah nilai dari inputan user bernilai “y”.  Proses memanggil method cetakHasil dari objek pht dengan parameter method sederhana dari objek pht berparameter method getHasil dari objek pht dan “x”.  Proses mencetak “ ” (enter).  Fungsi untuk mengakhiri case3.  Fungsi yang dijalankan apabila nilai dari inputan user bernilai 4.  Proses memanggil method input.  Pemanggilan method static pembagian dari class Perhitungan dengan parameter method getBilangan1 dan getBilangan2 dari objek pht.  Proses memanggil method cetakHasil dari objek pht dengan parameter nilai method getHasil dari objek pht dan “:”.  Proses memanggil method sederhanakan.  Pengecekan kondisi apakah nilai dari inputan user bernilai “y”.  Proses memanggil method cetakHasil dari objek pht dengan parameter method sederhana dari objek pht berparameter method getHasil dari objek pht dan “:”.  Proses mencetak “ ” (enter).  Fungsi untuk mengakhiri case4.  Fungsi yang dijalankan ketika inputan user bernilai 5.  Proses memanggil method sederhana dari objek pht.  Fungsi untuk mengakhiri case 5.  Fungsi yang dijalankan ketika inputan user tidak sesuai dengan case diatas.  Proses mencetak “Pilihan tidak ada”.  Fungsi untuk mengakhiri default.  Ekspresi loop do-while yang berulangan ketika nilai dari variable loop bernilai “y”.  Deklarasi method cetakMenu tanpa pengembalian (void) bertipe static dengan access modifier public.  Proses mencetak “Perhitungan”.  Proses mencetak “1. Penjumlahan”.  Proses mencetak “2. Pengurangan”.  Proses mencetak “3. Perkalian”.  Proses mencetak “4. Pembagian”.  Proses mencetak “5. Penyederhanaan Pecahan”.  Proses mencetak “0. Keluar”.  Proses mencetak “Pilih:”.  Dekalrasi method input tanpa pengembalian (void) bertipe static dengan access modifier public.  Proses mencetak “Masukkan bilangan 1 = “.  Proses memanggil method setBilangan1 dari objek pht dengan parameter inputan user bertipe data double.  Proses mencetak “Masukkan bilangan 2 = “.  Proses memanggil method setBilangan2 dari objek pht dengan parameter inputan user bertipe data double.  Dekalrasi method sederhanakan tanpa pengembalian (void) bertipe static dengan access modifier public.  Proses mencetak “Sederhanakan(y/n)?”. |

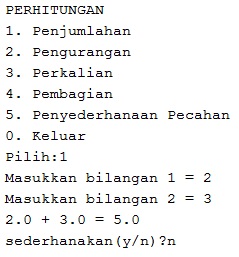
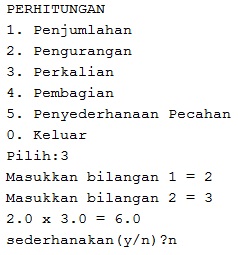
**Masalah 2**

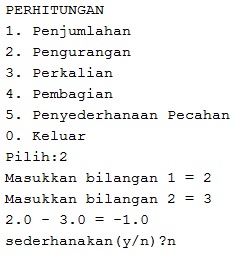
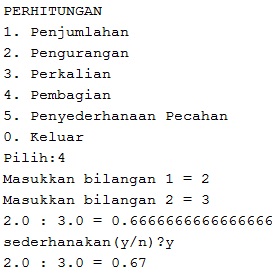
|  |  |
| --- | --- |
| **Perhitungan.java** | |
| 1  2  3  4  5  6  7  8  10  11  13  14  16  17  18  19  20  21  22  23  25  26  27  28  29  30  31  32  35  36  37  38  39  40  41  42  43  45  46  47  48  49  50  51  54  56  58  59  60  61  63  64  66  67  68  69  70  71  72  74  75  76  77  79  80 | Deklarasi class dengan nama PabrikJaket.  Inisialisasi atribut konstanta jaketA bertipe data integer dengan access modifier private bernilai 100000.  Inisialisasi atribut konstanta jaketB bertipe data integer dengan access modifier private bernilai 125000.  Inisialisasi atribut konstanta jaketC bertipe data integer dengan access modifier private bernilai 175000.  Inisialisasi konstanta objek pj dari class PabrikJaket bertipe static dengan access modifier private.  Inisialisasi array dua dimensi keranjang dengan panjang 100 dan 4 dengan access modifier private.  Dekalrasi method getter getJaketA dari atribut jaketA bertipe data integer.  Proses pengembalian data dari atribut jaketA ke dalam method getter getJaketA.  Dekalrasi method getter getJaketB dari atribut jaketB bertipe data integer.  Proses pengembalian data dari atribut jaketB ke dalam method getter getJaketB.  Dekalrasi method getter getJaketC dari atribut jaketC bertipe data integer.  Proses pengembalian data dari atribut jaketC ke dalam method getter getJaketC.  Deklarasi method pilihan tanpa pengembalian data (void) dengan access modifier public berparameter variable pilihan bertipe data String.  Proses mencetak “Harga satua Rp.”.  Proses pengecekan apakah nilai dari variable pilihan bernilai “a”.  Proses mencetak nilai mehtod getJaketA dari objek pj.  Proses pengecekan apakah nilai dari variable pilihan bernilai “b”.  Proses mencetak nilai method getJaketB dari objek pj.  Fungsi yang dijalankan ketika kedua pengecekan diatas tidak terpenuhi.  Proses mencetak nilai method getJaketC dari objek pj.  Proses mencetak “Apabila membeli lebih dari 100 biji”.  Proses mencetak “Harga satuan menjadi”.  Proses pengecekan apakah nilai dari variable pilihan bernilai “a”.  Proses mencetak nilai mehtod getJaketA dikurangi 5000 dari objek pj.  Proses pengecekan apakah nilai dari variable pilihan bernilai “b”.  Proses mencetak nilai method getJaketB dikurangi 5000 dari objek pj.  Fungsi yang dijalankan ketika kedua pengecekan diatas tidak terpenuhi.  Proses mencetak nilai method getJaketC dikurangi 15000 dari objek pj.  Deklarasi method pembelian dengan pengembalian data bertipe data integer dengan access modifier public berparameter variable pilihan bertipe data String dan variable jumlah bertipe data integer.  Inisialisasi variable total bertipe data integer bernilai 0.  Pengecekan apakah nilai dari variable jumlah lebih dari 0 dan kurang dari samadengan 100.  Proses pengecekan apakah nilai dari variable pilihan bernilai “a”.  Pengisian variable total dengan nilai method getJaketA dari objek pj dikali jumlah.  Proses pengecekan apakah nilai dari variable pilihan bernilai “b”.  Pengisian variable total dengan nilai method getJaketB dari objek pj dikali jumlah.  Fungsi yang dijalankan ketika kedua pengecekan diatas tidak terpenuhi.  Pengisian variable total dengan nilai method getJaketC dari objek pj dikali jumlah.  Fungsi yang dijalankan ketika nilai jumlah lebih dari 100 dan kurang dari samadengan 0.  Proses pengecekan apakah nilai dari variable pilihan bernilai “a”.  Pengisian variable total dengan nilai method getJaketA dari objek pj dikurangi 5000 kemudian dikali jumlah.  Proses pengecekan apakah nilai dari variable pilihan bernilai “b”.  Pengisian variable total dengan nilai method getJaketB dari objek pj dikurangi 5000 kemudian dikali jumlah.  Fungsi yang dijalankan ketika kedua pengecekan diatas tidak terpenuhi.  Pengisian variable total dengan nilai method getJaketC dari objek pj dikurangi 15000 kemudian dikali jumlah.  Proses pengembalian nilai variable total kedalam method pembelian.  Deklarasi method keranjang tanpa pengembalian nilai (void) dengan access modifier public berparameter variable pilihan bertipe data String dan variable total, jumlah, x, dan bayar bertipe data integer.  Pengisian array keranjang indeks ke x,0 dengan nilai variable pilihan.  Pengisian array keranjang indeks ke x,1 dengan nilai dari variable jumlah dikonversi menjadi String dengan fungsi Integer.toString.  Pengisian array keranjang indeks ke x,2 dengan nilai dari variable total dikonversi menjadi String dengan fungsi Integer.toString.  Pengisian array keranjang indeks ke x,2 dengan nilai dari variable bayar dikonversi menjadi String dengan fungsi Integer.toString.  Deklarasi method getter getKeranjang atribut array keranjang bertipe data arrray String dua dimensi dengan access modifier public.  Proses pengembalian array keranjang ke dalam method getKeranjang.  Deklarasi method cetakStruk tanpa pengembalian data (void) bertipe static dengan access modifier public berparameter array dua dimensi keranjang bertipe data String.  Inisialisasi variable y bertipe data integer bernilai 0.  Proses mencetak “Jenis Qty Harga”.  Proses mencetak “----------------------“.  Inisialisasi perulangan for dengan variable x bertipe data integer bernilai 0. Perulangan akan terus berulang selama nilai x kurang dari panjang array keranjang. Nilai x akan bertambah 1 setiap perulangan (increment).  Pengecekan apakah nilai dari keranjang indeks ke x,0 bernilai null (kosong).  Fungsi untuk menghentikan perulangan for.  Proses mencetak “Jaket “ diikuti nilai dari array keranjang indeks ke x,0.  Proses mencetak nilai dari array keranjang indeks ke x,1.  Proses mencetak “Rp.” Diikuti nilai dari array keranjang indeks ke x,2.  Proses merubah nilai y menjadi sama dengan nilai x.  Proses mencetak “-------------------“.  Proses mencetak “Total = Rp.” diikuti nilai dari array keranjang indeks ke y,3. |

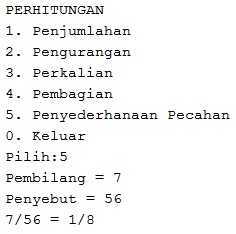
|  |  |
| --- | --- |
| **Main.java** | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  35  37  38  39  40  41  42  43  44  45  46  47  48  49  50  52  53  54  55  56  58  60  61  62  63  64  65  66  67  68  70  71 | Proses memanggil semua class dari package java.util.  Deklarasi class dengan nama Main.  Deklarasi objek pabrikjaket dari class PabrikJaket bertipe static dan dengan access modifier public.  Deklarasi objek scan dari class Scanner untuk memanggil fungsi inputan dengan tipe static dengan access modifier public.  Main method java.  Deklarasi variable loop bertipe data String dengan nilai “y”.  Deklarasi variable jumlah, total, x, bayar masing-masing bernilai 0 bertipe data integer,  Inisialisasi perulangan do-while.  Proses pemanggilan method cetakMenu.  Proses pengecekan inputan user bertipe data byte dengan fungsi switch.  Fungsi yang dijalankan apabila nilai inputan user bernilai 0. Apabila nilai switch bernilai 0 maka nilai dari variable loop akan dirubah menjadi “n”. Case 0 diakhiri.  Fungsi yang dijalankan apabila nilai dari inputan user bernilai 1.  Proses pemanggilan method pilihan dari objek pabrikjaket berparameter “a”.  Proses pemanggilan method cetakJumlah berparameter “A”.  Pengisian variable jumlah dengan inputan user bertipe data Integer.  Proses mencetak “ ” (enter).  Pengisian variable total dengan nilai method pembelian dari objek pabrikjaket berparameter “a”, dan nilai variable jumlah.  Pengisian variable bayar dengan nilai dari variable itu sendiri ditambah nilai dari variable total.  Proses pemanggilan method keranjang dari objek pabrikjaket berparameter “A”, nilai total, jumlah, x dan bayar.  Proses penambahan nilai variable x dengan 1 (increment).  Fungsi untuk menghentikan case 1.  Fungsi yang dijalankan apabila nilai dari inputan user bernilai 2.  Proses pemanggilan method pilihan dari objek pabrikjaket berparameter “b”.  Proses pemanggilan method cetakJumlah berparameter “B”.  Pengisian variable jumlah dengan inputan user bertipe data Integer.  Proses mencetak “ ” (enter).  Pengisian variable total dengan nilai method pembelian dari objek pabrikjaket berparameter “b”, dan nilai variable jumlah.  Pengisian variable bayar dengan nilai dari variable itu sendiri ditambah nilai dari variable total.  Proses pemanggilan method keranjang dari objek pabrikjaket berparameter “B”, nilai total, jumlah, x dan bayar.  Proses penambahan nilai variable x dengan 1 (increment).  Fungsi untuk menghentikan case 2.  Fungsi yang dijalankan apabila nilai dari inputan user bernilai 3.  Proses pemanggilan method pilihan dari objek pabrikjaket berparameter “c”.  Proses pemanggilan method cetakJumlah berparameter “C”.  Pengisian variable jumlah dengan inputan user bertipe data Integer.  Proses mencetak “ ” (enter).  Pengisian variable total dengan nilai method pembelian dari objek pabrikjaket berparameter “c”, dan nilai variable jumlah.  Pengisian variable bayar dengan nilai dari variable itu sendiri ditambah nilai dari variable total.  Proses pemanggilan method keranjang dari objek pabrikjaket berparameter “C”, nilai total, jumlah, x dan bayar.  Proses penambahan nilai variable x dengan 1 (increment).  Fungsi untuk menghentikan case 3.  Fungsi yang dijalankan apabila nilai dari inputan user bernilai 4.  Proses mencetak “KERANJANG BELI”.  Pemanggilan method cetakStruk bertipe static dari class PabrikJaket dengan parameter getKeranjang dari objek pabrikjaket.  Proses mencetak “Beli (y/n) : “.  Pengecekan apkah nilai dari inputan user bernilai “y”.  Proses mencetak “Pembelian berhasil” dan diikuti perubahan nilai variable loop menjadi “n”.  Proses mencetak “STRUK PEMBELIAN”.  Proses pemanggilan method cetakStruk bertipe static dari class PabrikJaket berparameter nilai getKeranjang dari objek pabrikjaket.  Proses mencetak “TERIMA KASIH”.  Proses mencetak “ ” (enter”.  Fungsi untuk menghentikan case 4.  Fungsi yang dijalankan apabila case diatas tidak terpenuhi.  Proses mencetak “Pilihan tidak ada”.  Fungsi untuk menghentikan default.  Ekspresi perulangan do-while yakni akan terus berulang ketika nilai variable loop bernilai “y”.  Deklarasi method cetakMenu tanpa pengembalian nilai (void) bertipe static dan mempunya access modifier public.  Proses mencetak “MENU”.  Proses mencetak “Beli Jaket”.  Proses mencetak “1. Bahan A”.  Proses mencetak “2. Bahan B”.  Proses mencetak “3. Bahan C”.  Proses mencetak “4 Lihat Keranjang”.  Proses mencetak “0. Keluar”.  Proses mencetak “Pilih: “.  Deklarasi method cetakJumlah tanpa pengembalian nilai (void) bertipe static dan mempunya access modifier public dan berparameter variable pilihan bertipe data String.  Proses mencetak “Jumlah Jaket” diikuti nilai variable pilihan dan “yang anda beli : “. |

1. **SCREENSHOT PROGRAM**

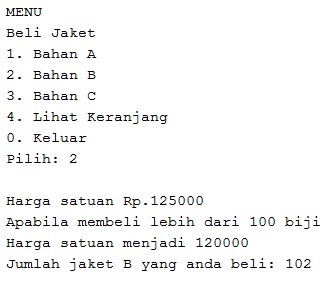
**Masalah 1**

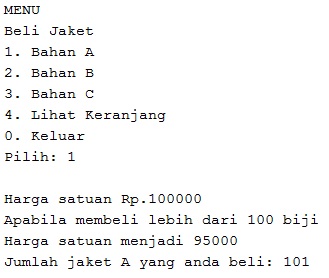
****

****

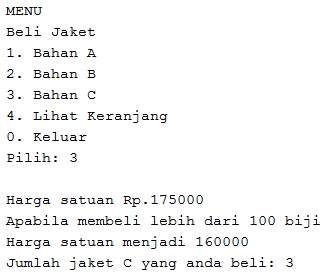
****

**Masalah 2**

****







****

1. **PRAKTIKUM**
2. **Static Method**
3. Apakah yang disebut dengan static variabel? Dan apa fungsi dari static variabel serta kapan kita dapat menggunakan static variabel?

|  |  |
| --- | --- |
| Jawaban : | Static variable ialah variable yang menggunakan keyword static yang berfungsi agar variable tersebut tidak untuk dirubah-rubah. Dalam pengaksesan dari class lain tidak diakses menggunakan objek namun langsung nama class tersebut diikuti nama variable atau method. |

1. Mengapa pada main method harus dituliskan static? Jelaskan jawaban anda beserta dengan alasan!

|  |  |
| --- | --- |
| Jawaban : | Karena method main() akan dipanggil oleh interpreter Java untuk pengeksekusiannya tanpa harus melakukan instansiasi terhadap suatu kelas terlebih dahulu. |

1. Lakukan percobaan diatas dan benahi jika menemukan kesalahan!

|  |  |
| --- | --- |
| Jawaban : | Tidak ada error yang terjadi pada kedua class Aritmatika.java dan MainAritmatika.java |

1. Jika pada tubuh method hitungPenjumlahan ditambahkan syntax hitungPerkalian(a,b) apa yang terjadi? Jelaskan?

|  |  |
| --- | --- |
| Jawaban : | Tidak terjadi error, karena method hitungPerkalian bertipe static dan bisa diakses dari method non-static dalam satu class tersebut. |

1. Jika pada tubuh method hitungPerkalian ditambahkan syntax hitungPenjumlahan(a,b) apa yang terjadi? Jelaskan?

|  |  |
| --- | --- |
| Jawaban : | Akan terjadi error, karena method hitungPenjumlahan tidak bertipe static yakni dalam pemanggilannya harus mendelarasikan objek darik class tersebut terlebih dahulu. |

1. Tambahkan method non static dengan nilai balikan double untuk menghitung pembagian dengan parameter String nil dan String nil2, dan panggil method tersebut pada method main

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Jawaban : | Pada Aritmatika.java ditambahkan method :   |  |  | | --- | --- | | **Aritmatika.java** | | | 1  2  3  4  5  6  7 | public double hitungPembagian(String nil1, String nil2){  double a = Double.parseDouble(nil1);  double b = Double.parseDouble(nil2);  double hasil = a/b;  return hasil;  } |   Pada Main.java ditambahkan kode :   |  |  | | --- | --- | | **Main.java** | | | 1  2  3  4  5  6  7 | System.out.print("masukkan nilai 1 : ");  String val1 = in.next();  System.out.print("masukkan nilai 2 : ");  String val2 = in.next();  Aritmetika b = new Aritmetika();  System.out.println("nilai pembagian adalah : "+b.hitungPembagian(val1, val2)); | |

1. **Konstanta Final**
2. Benahi kode Vehicle1 dan TestVehicle1 dan perbaiki jika menemui kesalahan!

|  |  |
| --- | --- |
| Jawaban : | Tidak ada kesalahan yang membuat error pada kedua class tersebut. |

1. Hapus separator “/” pada file Vehicle1.java pada baris 4-6 serta pada file TestVehicle1.java pada baris 6, apa yang terjadi dan jelaskan!

|  |  |
| --- | --- |
| Jawaban : | Akan terjadi error pada kedua class karena Main.java akan mencoba merubah nilai variable maxLoad yang bertipe final melalui konstruktor sedangkan variable yang beritipe final nilainya tidak dapat dirubah dalam bentuk apapun. |

1. Pada file Vehicle1.java variabel load ubah menjadi konstanta final, apa yang terjadi, jelaskan!

|  |  |
| --- | --- |
| Jawaban : | Akan terjadi error pada class tersebut pada baris 17 karena pada baris tersebut ada perubahan untuk variable load yang ditambah dengan weight yang hasilnya akan disimpan pada variable load kembali sedangkan variable load telah berubah tipe menjadi final. |

1. Tambahkan keyword “static” pada file Vehicle1.java variabel maxLoad, apa yang terjadi dan jelaskan!

|  |  |
| --- | --- |
| Jawaban : | Tidak terjadi error atau kesalahan pada program karena variable tersebut masih dapat dikses oleh method setelahnya yang menggunakan variable tersebut. |

1. **KESIMPULAN**

Static method adalah method yang dapat dipakai tanpa harus menginisialisasi suatu class dengan objek terlebih dahulu. Method static hanya dimiliki oleh class dan tidak dapat digunakan oleh instance (atau objek) dari suatu class. Method static dibedakan dari method yang dapat instance di dalam suatu class oleh kata kunci static. Method static digunakan ketika method tersebut berfungsi tidak untuk dimiliki suatu objek seperti nama, nim. Namun method static befungsi sebagai method yang dimiliki class tersebut seperti cetakData, dan sebagainya. Perbedaan dengan instance method ialah dalam pemanggilannya. Dalam instance method harus didefinisikan objek dari class method tersebut sebelum memanggil method. Contoh ketika main class memanggil method instance dari class mahasiswa :

|  |  |
| --- | --- |
| **Main.java** | |
| 1  2 | Mahasiswa mhs = new Mahasiswa();  mhs.setNama(“Arif”); |

Contoh diatas menunjukkan bahwa method instance setNama harus dipanggil melalui inisialisasi objek mhs terlebih dahulu. Sedangkan method static langsung dipanggil melalui nama class-nya. Contoh :

|  |  |
| --- | --- |
| **Main.java** | |
| 1 | Mahasiswa.cetakData(); |

Seperti contoh diatas, method static cetakData dalam pemanggilannya tidak melalui objek yang didefinisikan terlebih dahulu, namun langsung dengan nama class-nya.

Konstanta final adalah suatu fungsi untuk mendefinisikan variable menjadi konstanta atau final yakni tidak dapat dirubah nilainya dengan cara apapun namun tetap dapat diakses nilainya untuk dipakai di variable lain. Contoh :

|  |  |
| --- | --- |
| **Main.java** | |
| 1  2  3 | final double phi=3.14;  phi=10;  double luas = phi \* 7 \* 7; |

Contoh diatas apabila dieksekusi akan terjadi error pada baris ke-2 karena baris tersebut mencoba merubah nilai dari variable phi menjadi 10 sedangkan variable phi sudah bertipe konstanta. Namun, apabalia baris ke-2 dihilangkan, baris ke-3 akan tetap bisa berjalan untuk perhitungan yang menggunakan variable phi tersebut. Konstanta final digunakan ketiga mendefinisikan suatu variable yang bernilai tetap dan tidak mungkin untuk dirubah atau diganti nilainya dengan yang baru. Contoh : Medefinisikan phi untuk menghitung lingkaran tidak mungkin nilainya dirubah menjadi 10 karena nilai phi tersebut sudah bersifat tetap dan tidak boleh dirubah.