1. **DEFINISI MASALAH**

1. 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.

2. 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. Karena produksi jaket tersebut terbilang masih baru pihak perusahaan melakukan strategi dengan memberi diskon harga bila pembeli membeli banyak jenis jaket, diskon di berikan apabila :

a. Jaket A di beli sebanyak lebih dari 100 buah maka harga menjadi 95.000 per biji

b. Jaket B di beli sebanyak lebih dari 100 buah maka harga menjadi 120.000 per biji

c. 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.

**SOURCE CODE**

**1.**

**dtbs.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | public class dtbs {  public static void Penjumlahan(int a, int b) {  int nilai = a + b;  System.out.println("nilai penjumlahan adalah : " + nilai);  }  public static void Pengurangan(int a, int b) {  int nilai = a - b;  System.out.println("nilai pengurangan adalah : " + nilai);  }  public void Perkalian(int a, int b) {  int nilai = a \* b;  System.out.println("nilai perkalian adalah : " + nilai);  }  public void Pembagian(double a, double b) {  double nilai = a / b;  System.out.println("nilai pembagian adalah : " + nilai /\*+ getSederhana()\*/);  }  public void Sederhana(int a, int b) {  int nilai = a / b;  System.out.println("Hasil dari penyederhanaan adalah\t: " + nilai);  }  } |

**Operasi.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  2324  25  26  27  28  29  30  31  32  33  3435  36  37  38  39  40  41  42  43  44  45  46  47  48  49 | import java.util.Scanner;  public class Operasi {  public static void main(String[] args) {  Scanner a = new Scanner(System.in);  Scanner b = new Scanner(System.in);  dtbs d = new dtbs();  int ulang = -1;  do {  System.out.println("\t1. Pertambahan\n\t2. Pengurangan\n\t3. Perkalian\n\t4. Pembagian\n");  System.out.print("Pilih\t: ");  int pilih = a.nextInt();  System.out.print("Masukkan angka pertama\t: ");  int ang1 = a.nextInt();  System.out.print("Masukkan angka kedua\t: ");  int ang2 = b.nextInt();  System.out.println();  switch (pilih) {  case 1:  dtbs.Penjumlahan(ang1, ang2);  System.out.println();  break;  case 2:  dtbs.Pengurangan(ang1, ang2);  System.out.println();  break;  case 3:  d.Perkalian(ang1, ang2);  System.out.println();  break;  case 4:  d.Pembagian(ang1, ang2);  System.out.println();  d.Sederhana(ang1, ang2);  System.out.println();  break;  default:  System.out.println("Ulangi Program ?\n\t1. Ya\n\t2. Tidak\n");  System.out.print("Pilih\t: ");  ulang = a.nextInt();  System.out.println();  break;  }  System.out.println("Ulangi Program ?\n\t1. Ya\n\t2. Tidak\n");  System.out.print("Pilih\t: ");  ulang = a.nextInt();  } while (ulang == 1);  System.out.println("\tTerima Kasih\n");  }  } |

**2.**

**Database.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 | public class Database {  private int Beli;  private int A, B, C;  private final int Max = 100;  public int getBeli() {  return this.Beli;  }  public void setBeli(int Beli) {  this.Beli = Beli;  }  public int getMax() {  return this.Max;  }  public boolean Bonus(int take) {  int dt = 0;  dt = this.Beli + take;  if (dt > Max) {  this.Beli = this.Beli + take;  return true;  } else {  return false;  }  }  public void jenisA() {  if (Beli > 100) {  System.out.println("Mendapat Bonus 5000 @ baju");  int harga = Beli \* 95000;  System.out.println("Total Harga\t: Rp " + harga + ",-");  System.out.println();  } else {  System.out.println("Harga Tetap");  int harga = Beli \* 100000;  System.out.println("Total Harga\t: Rp " + harga + ",-");  System.out.println();  }  }  public void jenisB() {  if (Beli > 100) {  System.out.println("Mendapat Bonus 5000 @ baju");  int harga = Beli \* 95000;  System.out.println("Total Harga\t: Rp " + harga + ",-");  System.out.println();  } else {  System.out.println("Harga Tetap");  int harga = Beli \* 100000;  System.out.println("Total Harga\t: Rp " + harga + ",-");  System.out.println();  }  }  public void jenisC() {  if (Beli > 100) {  System.out.println("Mendapat Bonus 5000 @ baju");  int harga = Beli \* 95000;  System.out.println("Total Harga\t: Rp " + harga + ",-");  System.out.println();  } else {  System.out.println("Harga Tetap");  int harga = Beli \* 100000;  System.out.println("Total Harga\t: Rp " + harga + ",-");  System.out.println();  }  }  } |

**Jual\_Baju.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 | import java.util.Scanner;  public class Jual\_Baju {  public static void main(String[] args) {  Scanner a = new Scanner(System.in);//for int  Scanner b = new Scanner(System.in);//for String  Database bj = new Database();  System.out.println("\tBakhul Gombal Ora Larang\n");  System.out.println("Kami Menyediakan Gombal dengan jenis kain \n"  + "\t1. A = Kain Nagata\n\t2. B = Kain Mori\n\t3. C = Kain Jala\n");  System.out.print("Pilih\t: ");  int plh = a.nextInt();  switch (plh) {  case 1:  System.out.print("Gombal jenis kain Nagata\t: ");  bj.setBeli(a.nextInt());  //bj.Bonus(100);  bj.jenisA();  System.out.println();  break;  case 2:  System.out.print("Gombal jenis kain Mori\t: ");  bj.setBeli(a.nextInt());  //bj.Bonus(100);  bj.jenisB();  System.out.println();  break;  case 3:  System.out.print("Gombal jenis kain Jala\t: ");  bj.setBeli(a.nextInt());  //bj.Bonus(100);  bj.jenisC();  System.out.println();  break;  default:  System.out.println("Tidak Ada Yang Anda Pilih, Berarti Anda Tidak Membeli");  System.out.println("\tTERIMA KASIH");  }  }  } |

1. **PEMBAHASAN**
2. **dtbs.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | Kelas dtbs  Method penjumlahan  Nilai input  Output  kurung  method pengurangan  nilai input  Output  kurung  method perkalian  nilai input  Output  kurung  method pembagian  nilai input  Output  kurung  method penyederhanaan dari pembagian  nilai input  Output  kurung  kurung |

**Operasi.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  2324  25  26  27  28  29  30  31  32  33  3435  36  37  38  39  40  41  42  43  44  45  46  47  48  49 | Import scanner  Kelas operasi  Method static void main  Scanner input a  Scanner input b  Pemanggil dari class dtbs  Integer ulang  do  output pilihan  output pilih  input pilih  perintah memasukkan angka pertama  input angka pertama  perintah memasukkan angka kedua  input angka kedua  output enter  switch pilih  case 1:  pemanggil static method penjumlah  output enter  break;  case 2:  pemanggil static method pengurang  output enter  break;  case 3:  pemanggil static method perkalian  output enter  break;  case 4:  pemanggil static method pembagian  output enter  pemanggil static method penyederhanaan  output enter  break;  default:  pilihan ulangi program  perintah pilih  input ulang  output enter  break;  kurung  pilihan ulangi program  perintah pilih  input ulang  } while ulang berinput 1 maka kembali ke do  Output terima kasih  kurung  kurung |

**2. Database.java**

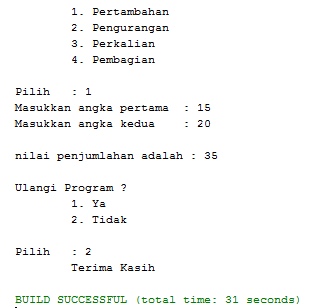
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63 | Keals Database  private integer Beli  private integer A dan B dan C  private final integer Max = 100  getter Beli  kembali ke Beli  kurung  Setter Beli  Set Beli  Kurung  Getter Max  Kembali ke Max;  kurung  boolean Bonus (konstruktor take)  integer dt = 0  perhitungan dt = Beli + take;  jika (dt lebih dari Max) {  perhitungan Beli = Beli + take;  return true  } else  return false  kurang  kurang  public void jenisA  jika (Beli lebih dari 100) {  Output mendapat bonus  Perhitungan bonus  Output total  Output enter  } jika tidak dapat bonus maka  Output harga tetap  Perhitungan harga tetaop  Output total  Output enter  kurung  kurung  public void jenisA  jika (Beli lebih dari 100) {  Output mendapat bonus  Perhitungan bonus  Output total  Output enter  } jika tidak dapat bonus maka  Output harga tetap  Perhitungan harga tetaop  Output total  Output enter  kurung  kurung  public void jenisA  jika (Beli lebih dari 100) {  Output mendapat bonus  Perhitungan bonus  Output total  Output enter  } jika tidak dapat bonus maka  Output harga tetap  Perhitungan harga tetaop  Output total  Output enter  kurung  kurung  kurung |

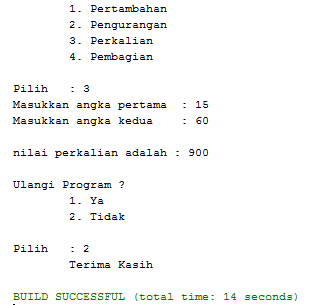
**Jual\_Baju.java**

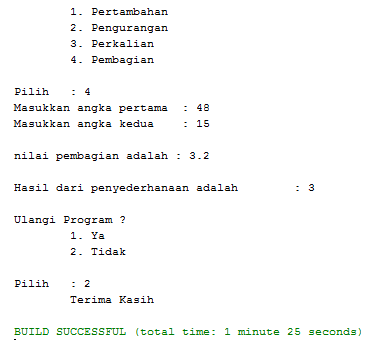
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38 | Import Scanner  Kelas Jual Baju  public static void main  Scanner a  Scanner b  Pemanggil kelas Database  Output header  Output pilihan Beli  Output perintah pilih  Input plh  Switch plh  case 1:  output pilihan A  input jumlah Beli  //bj.Bonus(100);  memanggil dari jenisA  output enter  break;  case 2:  output pilihan B  input jumlah Beli  //bj.Bonus(100);  memanggil dari jenisB  output enter  break;  case 3:  output pilihan C  input jumlah Beli  //bj.Bonus(100);  memanggil dari jenisC  output enter  break;  default:  output peringatan  output terima kasih  kurung  kurung  kurung |

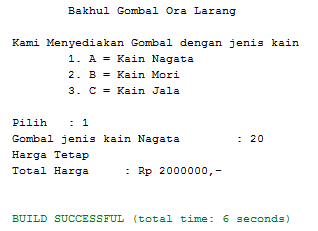
1. **SCREENSHOT PROGRAM**

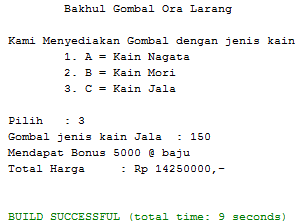
**1.**

****

****

****

****

****

1. **PRAKTIKUM**

**A.**

1. Apakah yang disebut dengan static variabel? Dan apa fungsi dari static variabel serta kapan kita dapat menggunakan static variabel?

Jawab : Method yang tidak tergantung terhadap objek manapun dari suatu kelas.

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

Jawab : Jika tidak diberi static, maka akan menjadi method biasa.

3. Lakukan percobaan diatas dan benahi jika menemukan kesalahan!

Jawab : Terdapat perbedaan nama class pada Vehicle

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

Jawab : Akan keluar hasil dari perkalian nilai a dan b

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

Jawab : Akan keluar hasil dari penjumlahan nilai a dan b

6. 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!

Jawab :

|  |  |
| --- | --- |
|  | public void hittungPembagian(int a, int b) {  int nilai = a / b;  System.out.println("nilai pembagian adalah : " + nilai);  } |

**B.**

1. Benahi kode Vehicle1 dan TestVehicle1 dan perbaiki jika menemui kesalahan!

Jawab : Terjadi error karena sudah ada yang menggunakan final diatasnya, atau malah akan terdapat 2 konstanta final

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

Jawab : Terjadi error karena tidak terdapat suatu nilai yang menjadi batasan pada konstanta final

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

Jawab : Terdapat perbedaan nama class pada Vehicle

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

Jawab : Tidak teerjadi perubahan meski hak access diberi static

1. **KESIMPULAN**
2. Apa yang dimaksud static method ?

Method yang tidak tergantung terhadap objek manapun dari suatu kelas.

1. Kapan kita harus menggunakan static method ?

Ketika kita ingin mendeklarasikan suatu method yang tidak tergantumg atau independen terhadap objek manapun dari suatu kelas.

1. Apa perbedaan dari sintax dari yang menggunakan dan tidak ?

Jika menggunakan method static, kita tidak harus membuat objeknya, namun jika kita tidak menggunakan static method, kita harus membuat objeknya terlebih dahuklu

1. Apa yang dimaksud konstanta final ?

Konstanta final adalah tipe data yang tidak dapat diubah ubah lagi.

1. Kapan kita harus menggunakannya ?

Ketika kita menginginkan tipe data yang bersifat tetap atau tudak dapat di override lagi.