

## Sistem Informasi

### Kelas D

#### 1. Soal

Selesaikan Kasus Taxi tersebut dengan menggunakan input langsung dari program (tidak perlu input dari user| tanpa objek scanner)

- Minimal input 3 taxi dengan masing-masing 2, 3, 4 penumpang. Taxi dalam array (main program)
- Tampilkan hasil akhir berupa Data penumpang dan Pendapatan Setiap Taxi, serta Hasil Pendapatan Total Perusahaan Taxi

#### 2. Narasi Jawaban

Mula-mula buat sebuah class yang bernama penumpang dan berisi data penumpang seperti nama, asal dan tujuan penumpang. Kemudian buat class Taxi yang berisi beberapa variabel yang digunakan untuk perhitungan besarnya tarif yang akan dibayarkan oleh penumpang. Lalu terakhir buat class taxiBeraksi yang berisi perintah untuk mencetak data penumpang, tarif, jumlah pendapatan pertaxi, jumlah pendaatn perusahaan

Di dalam class taxi pada method buatID terdapat sebuah statemen seleksi dimana digunakan untuk menentukan nomor id supir. Pada method pembayaran juga terdapat proses seleksi yang mana jika penumpang hanya membuka pintu dan membatalkan maka akan dikenakan tarif Rp.5000 begitu juga jika penumpang naik dengan jarak 2 km, dan jika naik lebih dari 2 km maka akan dikenakan tarif Rp.1500 per meternya. Pada method cetakDataPenumpang ada proses perulangan yang akan mengulang jumlah penumpang yang naik taxi tersebut.

#### 3. Diagram Class

Taxi
-asal : String -tujuan : String -nama : String - idTaxi : String -platNomor : String -namaSopir : String -biayaAwal : double final static -biayaPerKM : double final static -costTotal : double static -jumPenumpang : int -jumPendapatan: double -taxiCount : int -KM : int
+getNama (); +getAsal (); +getTujuan (); +buatID (int count); +toString (); +getJumlahTaxi(); +perjalanan(Penumpang Cust, int KM); +pembayaran(int KM);

<pre>+cetakDataPenumpang (); +dataCust(Penumpang cust) +getjumlahPendapatan ();</pre>
---

#### 4. Source Code

- Class Taxi

	<pre>//Judul      : Menghitung Penghasilan Taxi // Nama      : Wahyu Mega K. // Nim       : 145150407111042 // Kelas     : D // Tanggal   : 30 Maret 2015 // No Tugas  : 3  package taxi;  public class Taxi {      private String idTaxi;    /// deklarasi variabel     private String platNomor;     private String namaSupir;     private final static double biayaAwal = 5000;     private final static double biayaPerKM = 1500;     private static int taxiCount = 0;     private static double costTotal = 0;     private Penumpang[] penumpang = new Penumpang[10];     private double[] km = new double[10];     private int jumPenumpang;     private double jumPendapatan;      public Taxi(String plat, String Supir) { /// konstruktor         this.platNomor = plat;    /// inisialisasi nilai         this.namaSupir = Supir;         taxiCount++;    /// increment jumlah taxi         this.idTaxi = buatID(taxiCount);    /// inisialisasi ID taxi dengan memanggil method buatID         jumPenumpang = 0;    /// inisialisasi         jumPendapatan = 0;     }      private String buatID(int count) { /// method         String ID;    /// deklarasi variabel         if (count &lt; 10) {    /// jika jumlah taxi &lt;10             ID = "0" + String.valueOf(count);    /// inisialisasi ID dengan menambahkan nomor pendaftaran         } else {    /// jika &gt;10             ID = String.valueOf(count);    /// inisialisasi ID dengan menggunakan nomor pendaftaran         }         return ("2015" + ID);    /// pengembalian ID taxi dengan menambahkan tahun     } }</pre>
--	--

	<pre> public String toString() {    /// method     return String.format("Taxi id = %s   Plat = %s   Supir = %s", idTaxi, platNomor, namaSupir);    /// mengembalikan string yang telah diformat }  public static int getJumlahTaxi() {    // method     return taxiCount;    /// pengembalian nilai taxiCount }  public void perjalanan(Penumpang Cust, int KM) {    // method     jumPenumpang++;    /// increment jumlahPenumpang     penumpang[jumPenumpang - 1] = Cust;    // pengisian cust pada array penumpang     km[jumPenumpang - 1] = KM;     System.out.printf("Taxi %s   Penumpang ke %d : %s : %d-km : Bayar = %5.2f \n", idTaxi, jumPenumpang, Cust.getNama(), KM, pembayaran(KM));    /// mencetak dengan menggunakan formta }  public double pembayaran(int KM) {    // method     double biaya = 0;    // inisialisasi     if (KM &lt;= 2) {    /// jika KM &lt;=2         biaya = biayaAwal;    /// pemberian nilai pada biaya     } else {    // jika tidak maka         biaya = ((KM - 2) * biayaPerKM + biayaAwal);    /// biaya KM-2 * biayaPerKM + biayaAwal     }     jumPendapatan += biaya;    /// hasil dari jumlah pendapata + biaya di inisialisasikan ke jumPendapatan     return biaya;    /// pengembalian nilai }  public void cetakDataPenumpang () {    // method     for (int i=0 ; i&lt;jumPenumpang; i++) {    ///    melakukan perulangan         dataCust (penumpang [i]);    /// memanggil method dataCust dengan parameter penumpang [i]     } }  public void dataCust(Penumpang cust) {    /// method     System.out.printf("Nama : %s   Asal : %s   Tujuan : %s ", cust.getNama(), cust.getAsal(), cust.getTujuan());    /// mencetak data penumpang sesuai dengan yang format yang ada     System.out.println(); }  public double getjumlahPendapatan () {    /// method     return jumPendapatan ;    /// pengembailan nilai jumPendapatan }  } </pre>
--	---

- Class Penumpang

	<pre>//Judul      : Menghitung Penghasilan Taxi // Nama      : Wahyu Mega K. // Nim       : 145150407111042 // Kelas     : D // Tanggal   : 30 Maret 2015 // No Tugas  : 3  package taxi;  class Penumpang {      private String Nama; /// deklarasi variabel     private String Asal;     private String Tujuan;      public Penumpang(String N, String A, String T) { // konstruktor         this.Nama = N; // inisialisasi nilai         this.Asal = A;         this.Tujuan = T;     }      public String getNama() { // method         return this.Nama; // pengembalian nilai     }      public String getAsal () {         return this.Asal;     }      public String getTujuan () {         return this.Tujuan ;     } }</pre>
--	---

- Class TaxiBeraksi

	<pre>//Judul      : Menghitung Penghasilan Taxi // Nama      : Wahyu Mega K. // Nim       : 145150407111042 // Kelas     : D // Tanggal   : 30 Maret 2015 // No Tugas  : 3  package taxi;  public class TaxiBeraksi {      public static void main(String args[]) {         Taxi Taxi1 = new Taxi("N-2150", "Supardi"); // instansiasi         taxi         System.out.println(Taxi1); // mencetak data taxi         Taxi Taxi2 = new Taxi("N-1214", "Ridwan");         System.out.println(Taxi2);          Taxi Taxi3 = new Taxi("N-1245", "Ahmad");         System.out.println(Taxi3);          System.out.println("Jumlah Taxi = " + Taxi.getJumlahTaxi());         // mencetak dan memanggil method getJumlahTaxi pada kelas taxi         System.out.println();          Penumpang Cust1 = new Penumpang("Budi", "Mergan", "Dinoyo");         // instansiasi penumpang         Taxi1.perjalanan(Cust1, 5); // memanggil method perjalanan         pada taxi          Cust1 = new Penumpang("Samsul", "Landungsari", "ALun-alun");         Taxi1.perjalanan(Cust1, 7);          System.out.println();          Penumpang Cust2 = new Penumpang("Doni", "Bandulan", "Dinoyo");         Taxi2.perjalanan(Cust2, 3);          Cust2 = new Penumpang("Dino", "Dinoyo", "ALun-alun");         Taxi2.perjalanan(Cust2, 9);          Cust2 = new Penumpang("Reni", "Merjosari", "ALun-alun");         Taxi2.perjalanan(Cust2, 2);          System.out.println();          Penumpang Cust3 = new Penumpang("Ani", "Merjosari", "Dinoyo");         Taxi3.perjalanan(Cust3, 5);          Cust3 = new Penumpang("Rino", "Bandulan", "ALun-alun");         Taxi3.perjalanan(Cust3, 1);          Cust3 = new Penumpang("Deni", "Landungsari", "Dinoyo");         Taxi3.perjalanan(Cust3, 0);</pre>
--	--

	<pre>Cust3 = new Penumpang("Boni", "Arjosari", "ALun-alun"); Taxi3.perjalanan(Cust3, 13);  System.out.println();  System.out.println("Data Penumpang Taxi 1 : "); // mencetak kata data penumpang taxi Taxi1.cetakDataPenumpang(); /// memanggil method cetakDataPenumpang System.out.println("Data Penumpang Taxi 2 : "); Taxi2.cetakDataPenumpang(); System.out.println("Data Penumpang Taxi 3 : "); Taxi3.cetakDataPenumpang();  System.out.println();  System.out.printf("Pendapatan Taxi 1 : %.2f \n", Taxi1.getjumlahPendapatan()); /// mencetak dengan memanggil method getJumlahPendapatan System.out.printf("Pendapatan Taxi 2 : %.2f \n", Taxi2.getjumlahPendapatan()); System.out.printf("Pendapatan Taxi 3 : %.2f \n", Taxi3.getjumlahPendapatan()); System.out.println();  System.out.printf("Total Pendapatan Perusahaan : %.2f \n", (Taxi1.getjumlahPendapatan() + Taxi2.getjumlahPendapatan() + Taxi3.getjumlahPendapatan())); /// mencetak hasil pendapatan perusahaan dari taxi1, taxi 2, taxi 3      } }</pre>
--	---

## 5. Screen Shot Tampilan

## Sistem Informasi Kelas D

```
run:
Taxi id = 201501 | Plat = N-2150 | Supir = Supardi
Taxi id = 201502 | Plat = N-1214 | Supir = Ridwan
Taxi id = 201503 | Plat = N-1245 | Supir = Ahmad
Jumlah Taxi = 3

Taxi 201501 | Penumpang ke 1 : Budi : 5-km : Bayar = 9500.00
Taxi 201501 | Penumpang ke 2 : Samsul : 7-km : Bayar = 12500.00

Taxi 201502 | Penumpang ke 1 : Doni : 3-km : Bayar = 6500.00
Taxi 201502 | Penumpang ke 2 : Dino : 9-km : Bayar = 15500.00
Taxi 201502 | Penumpang ke 3 : Reni : 2-km : Bayar = 5000.00

Taxi 201503 | Penumpang ke 1 : Ani : 5-km : Bayar = 9500.00
Taxi 201503 | Penumpang ke 2 : Rino : 1-km : Bayar = 5000.00
Taxi 201503 | Penumpang ke 3 : Deni : 0-km : Bayar = 5000.00
Taxi 201503 | Penumpang ke 4 : Boni : 13-km : Bayar = 21500.00

Data Penumpang Taxi 1 :
Nama : Budi | Asal : Mergan | Tujuan : Dinoyo
Nama : Samsul | Asal : Landungsari | Tujuan : ALun-alun
Data Penumpang Taxi 2 :
Nama : Doni | Asal : Bandulan | Tujuan : Dinoyo
Nama : Dino | Asal : Dinoyo | Tujuan : ALun-alun
Nama : Reni | Asal : Merjosari | Tujuan : ALun-alun
Data Penumpang Taxi 3 :
Nama : Ani | Asal : Merjosari | Tujuan : Dinoyo
Nama : Rino | Asal : Bandulan | Tujuan : ALun-alun
Nama : Deni | Asal : Landungsari | Tujuan : Dinoyo
Nama : Boni | Asal : Arjosari | Tujuan : ALun-alun

Pendapatan Taxi 1 : 22000.00
Pendapatan Taxi 2 : 27000.00
Pendapatan Taxi 3 : 41000.00

Total Pendapatan Perusahaan : 90000.00
BUILD SUCCESSFUL (total time: 3 seconds)
```