MAKALAH, PERCOBAAN, LATIHAN, DAN TUGAS

MODUL PRAKTIKUM 4

Disusun sebagai salah satu tugas

mata kuliah PBO I



Patricia Joanne

140810160065

Dikumpulkan tanggal

19 September 2017

PROGRAM STUDI S-1 TEKNIK INFORMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS PADJADJARAN

2017

**Tugas Pendahuluan**

1. Apakah yang dimaksud dengan kelas, method, atribut dan obyek?

2. Buatlah contoh suatu kelas dan definisikan atribut dan methodnya!

3. Apakah yang dimaksud Overloading Constructor?

4. Buatlah kelas yang berisi main method yang membuat obyek dari kelas yang telah dibuat di soal no. 3. Selanjutnya obyek tersebut mengakses atribut dan methodnya.

5. Apa fungsi package?

6. Apa kegunaan kata kunci *Import* dan *This*!

1. Pengertian Class

Class merupakan suatu blueprint atau cetakan untuk menciptakan suatu instant dari object. Class juga merupakan grup suatu object dengan kemiripan attributes/properties, behaviour dan relasi ke object lain.

Pengertian Method

Method merupakan suatu operasi berupa fungsi-fungsi yang dapat dikerjakan oleh suatu object. Method didefinisikan pada class akan tetapi dipanggil melalui object. Metode menentukan perilaku objek, yakni apa yang terjadi ketika objek itu dibuat serta berbagai operasi yang dapat dilakukan objek sepanjang hidupnya.

Pengertian Atribut

Atribut merupakan nilai data yang terdapat pada suatu object yang berasal dari class. Attributes merepresentasikan karakteristik dari suatu object.

Pengertian Object

Object adalah instance dari class. Jika class secara umum merepresentasikan (template) sebuah object, sebuah instance adalah representasi nyata dari class itu sendiri.

2. Contoh class

public class Mahasiswa {

private String npm;

private String nama;

public void cetak(){

System.out.println("Mahasiswa Teknik Informatika");

}

public static void main(String args[]){

Mahasiswa mhs =new Mahasiswa();

mhs.cetak();

}

3. Overloading konstruktor merupakan suatu mekanisme pembuatan konstruktor yang memiliki bentuk lebih dari satu. Dalam hal ini pembeda antara satu konstruktor dengan konstruktor yang lain berupa jumlah parameter atau tipe parameter.

4. Package adalah sebuah sarana untuk mengelompokkan atau mengorganisasikan kelas dan interface yang sama atau sekelompok menjadi satu unit tunggal dalam library. Package mempengaruhi mekanisme hak akses ke kelas di dalamnya. Hal terpenting yang diperhatikan pada saat mendeklarasikan package, bahwa class tersebut harus disimpan pada suatu directory yang sama dengan nama packagenya. Alasan menggunakan package pada java ialah untuk menghindari tabrakan nama kelas yang akan dibuat dengan nama kelas yang sudah ada. Selain itu, salah satu yang menjadi keuntungan menggunakan package adalah untuk mudahnya developer dalam hal mencari dan me-manage akses yang diberikan.

5.Import

Meng-import paket (package) . Secara fisik, package dapat berupa folder yang berisi file/class, interface/enum lain yang kegunaannnya hampir sama sehingga patut untuk dikelompokkan. Package mempengaruhi hak akses ke class lainnya.

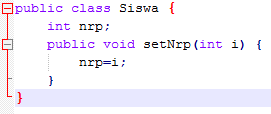
This

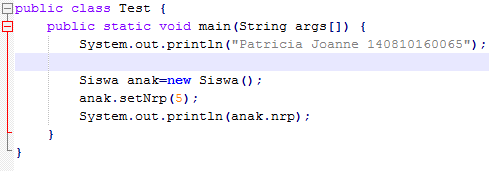
Digunakan untuk mewakili sebuah instance dari kelas dimana ia muncul. Kata kunci this dipergunakan pada pembuatan kelas dan digunakan untuk menyatakan objek sekarang. Untuk menghindari variabel yang sama antara variabel class dengan variabel properti.

**Bab 4**

**Pengenalan Pemrograman Berbasis Objek**

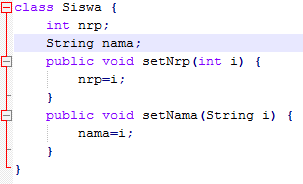
Percobaan 1: Mengakses anggota suatu class

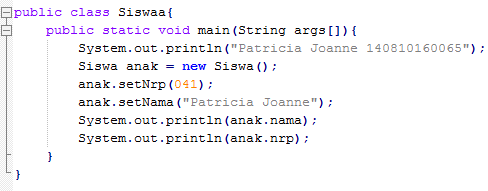






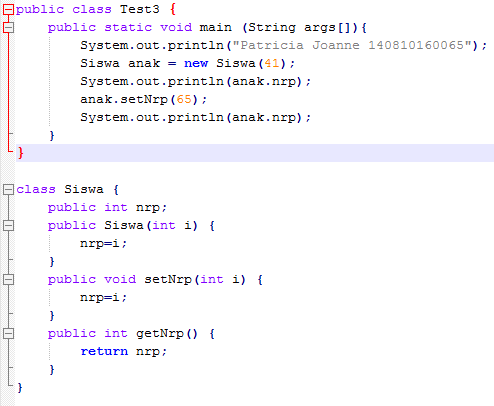
Percobaan 2: Mengakses anggota suatu class

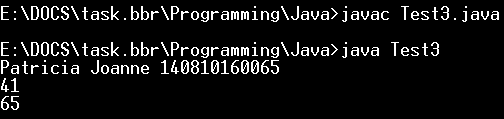




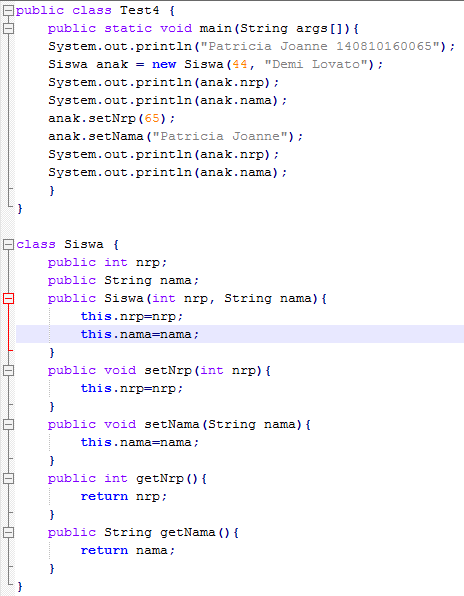


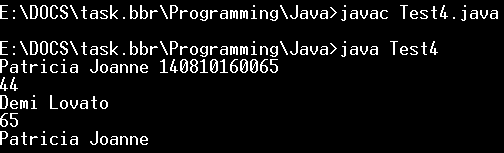
Percobaan 3: Mengimplementasikan UML class diagram dalam program



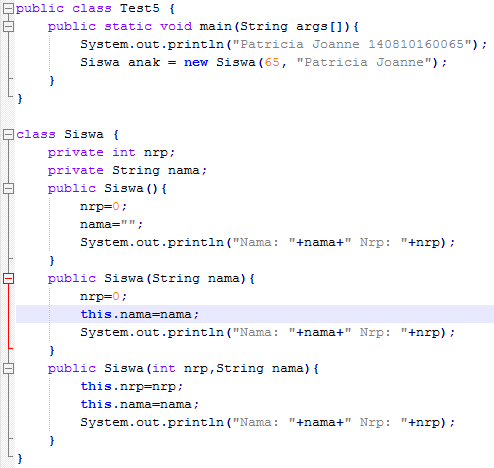


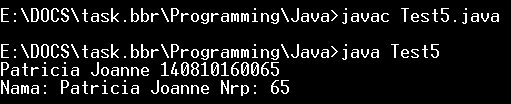
Percobaan 4: Melakukan enkapsulasi pada suatu class



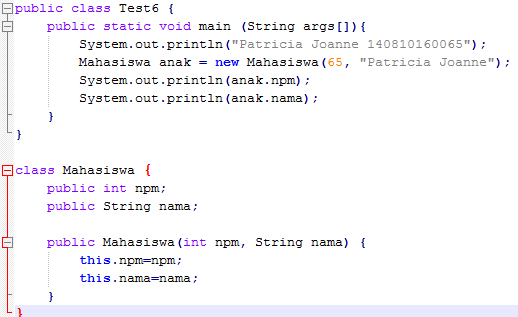


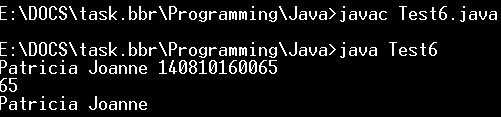
Percobaan 5: Melakukan overloading constructor



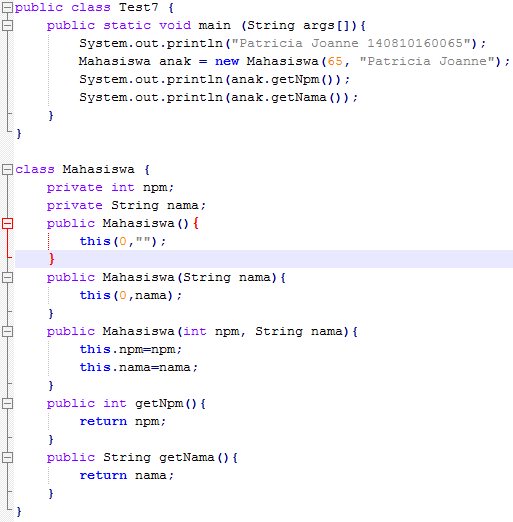


Percobaan 6: Menggunakan kata kunci this



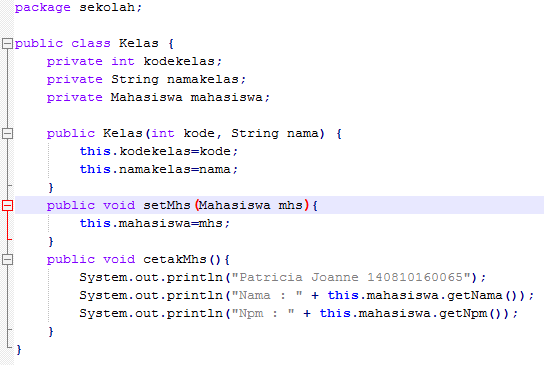


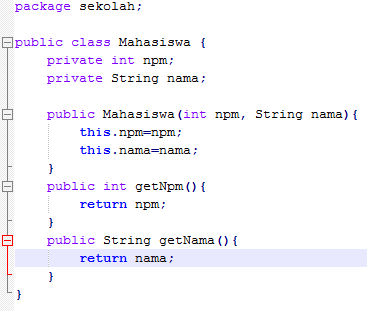
Percobaan 7: Memakai kata kunci this pada overloading constructor

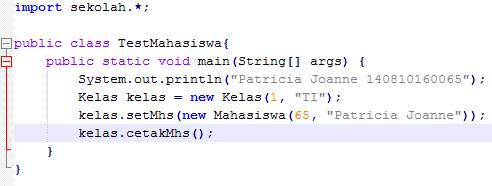


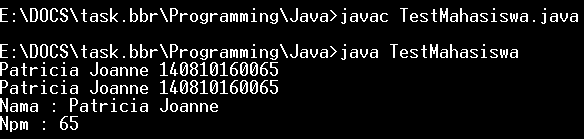


Percobaan 8: Menggunakan package dan import

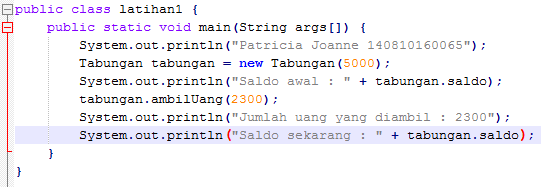


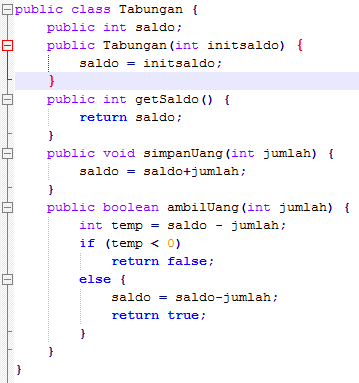


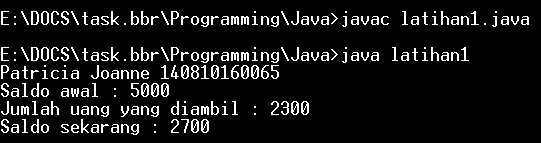




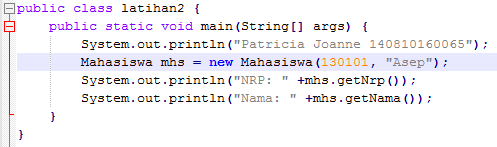
Latihan 1: Mengimplementasikan UML class diagram dalam program untuk class Tabungan

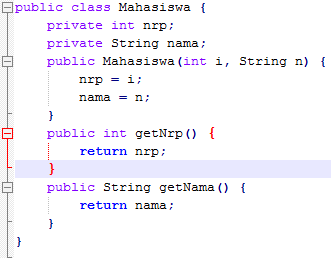






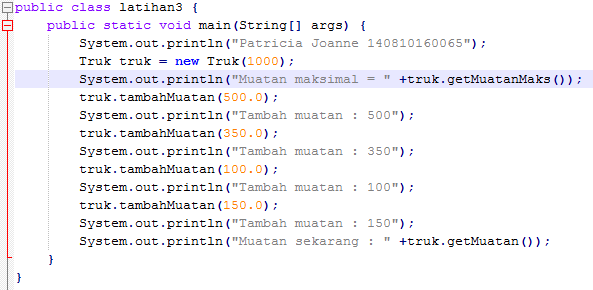
Latihan 2: Mengimplementasikan UML class diagram dalam program untuk class Mahasiswa

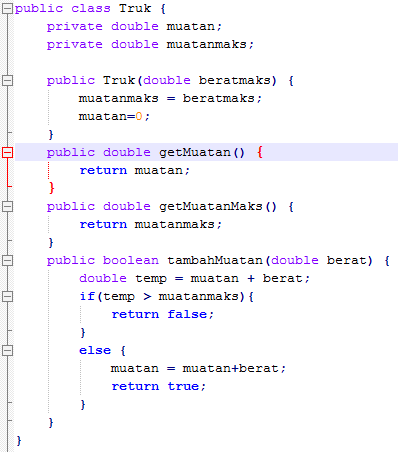


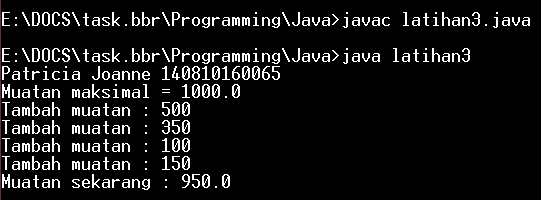




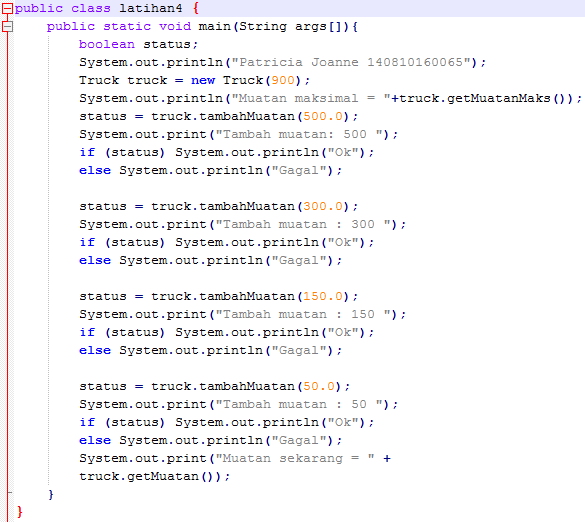
Latihan 3: Mengimplementasikan UML class diagram dalam program untuk class Truk

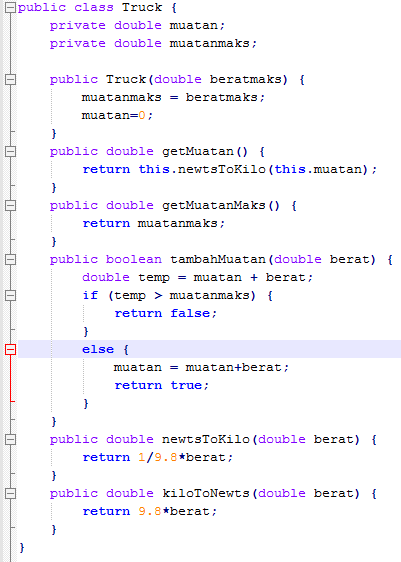


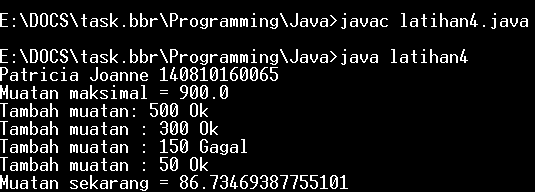




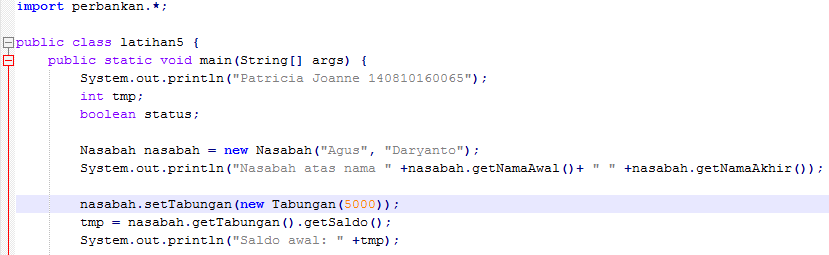
Latihan 4: Mengimplementasikan UML class diagram dalam program untuk class Truk

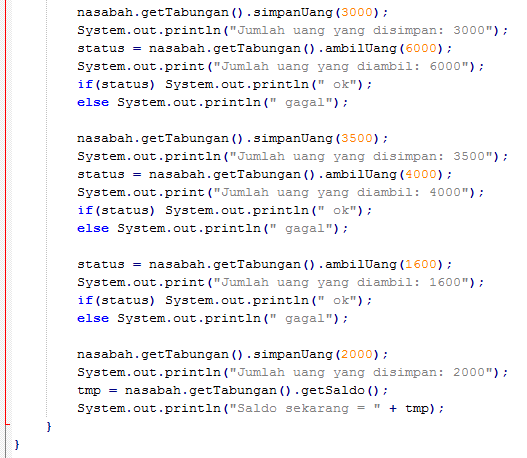


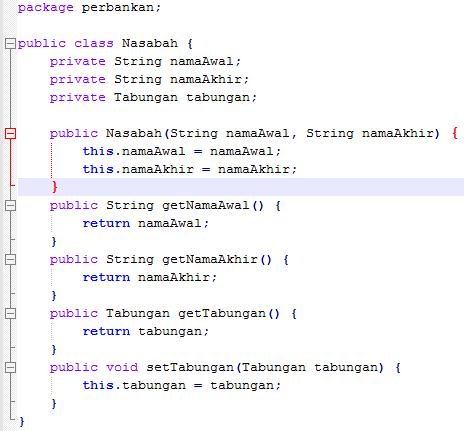


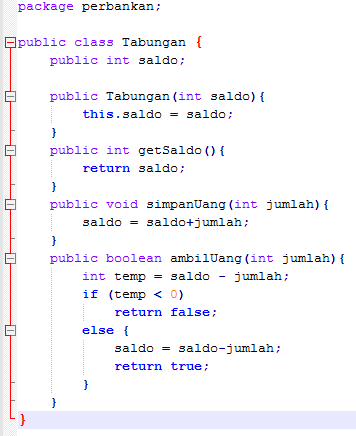


Latihan 5: Mengimplementasikan UML class diagram dalam program untuk package perbankan



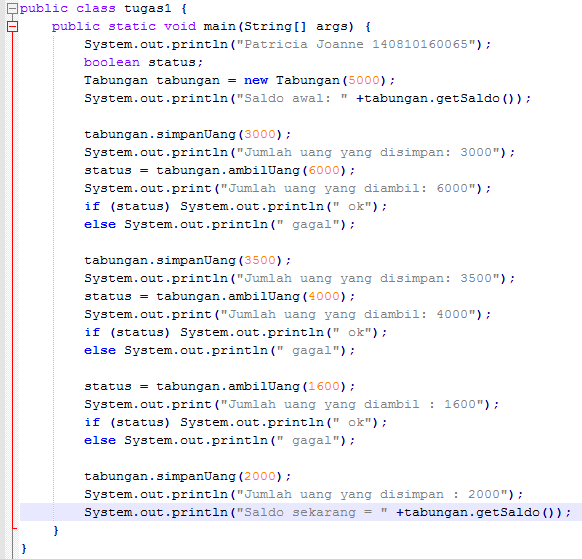


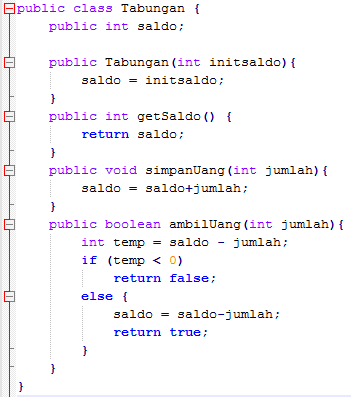


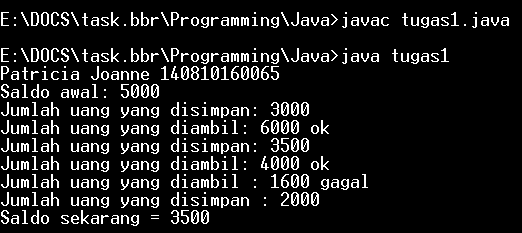




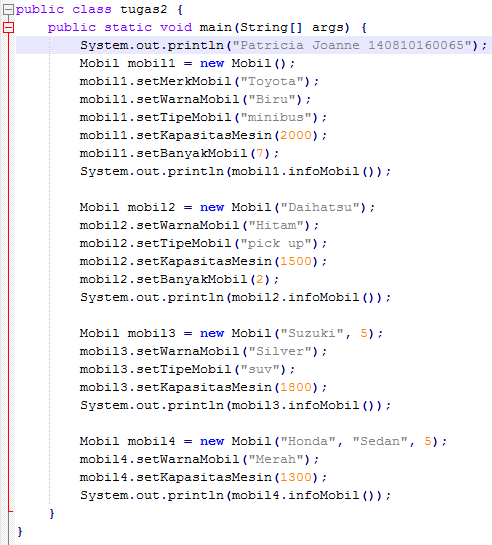
Tugas 1: Mengimplementasikan UML class diagram dalam program untuk class Tabungan

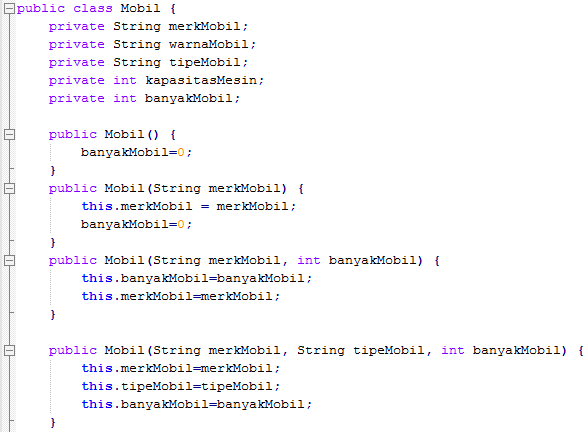


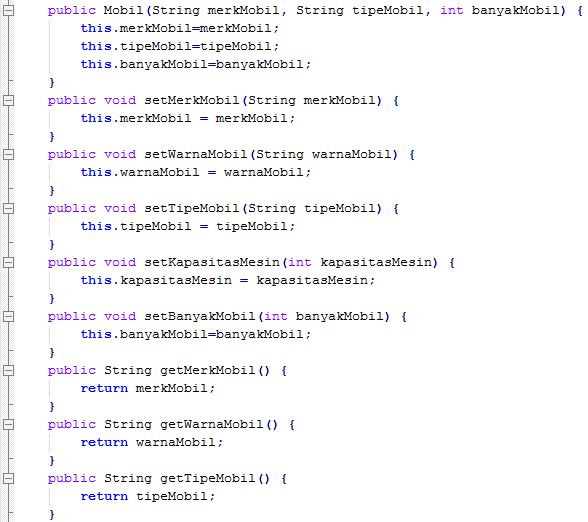


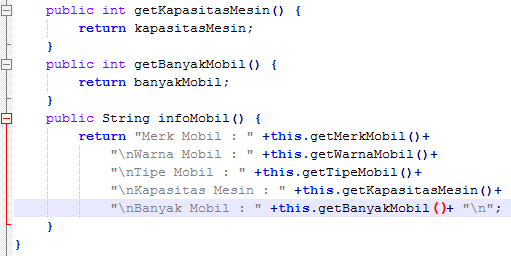


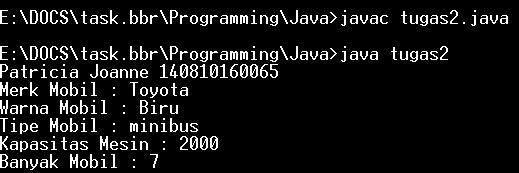
Tugas 2: Menganalisa, membuat UML class diagram dan implementasi program

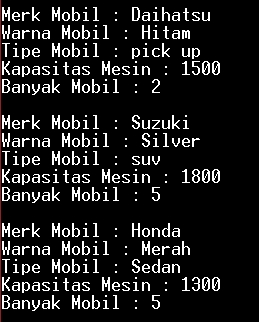




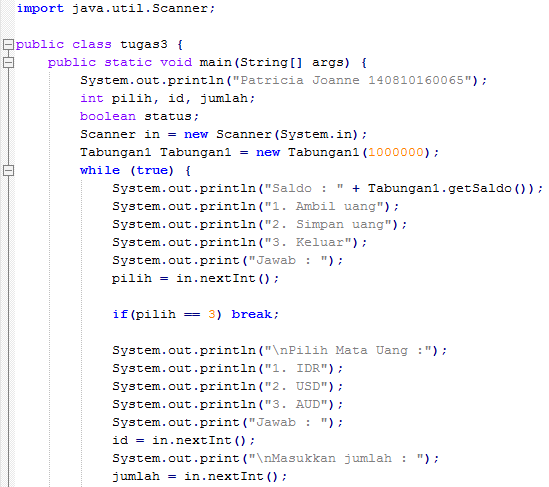


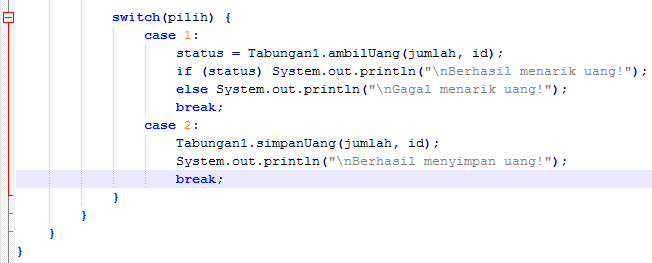


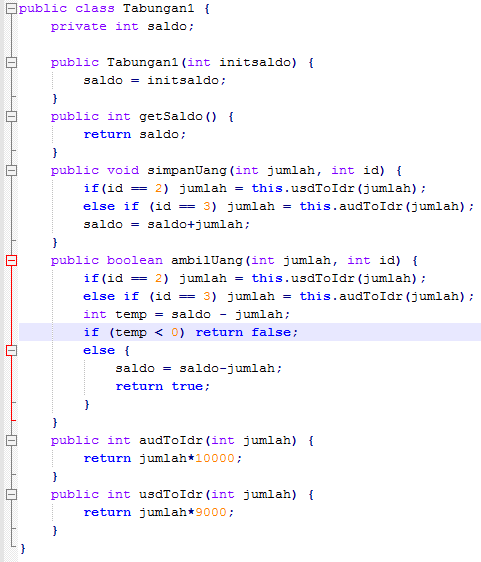


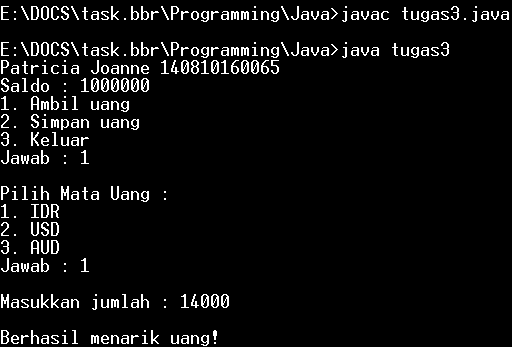


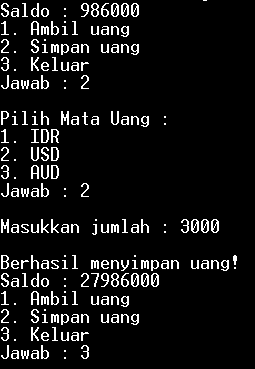
Tugas 3: Menerapkan konsep enkapsulasi pada kelas Tabungan yang terdapat di Tugas 1











Tugas 4: Mengembangkan package perbankan dengan tambahan class Bank

