

Tugas Pertemuan 6

Aplikasi Simulasi Transaksi Keuangan Sederhana Nasabah Perbankan

(Debugging dan Melengkapi Program)

Terdapat sebuah aplikasi sederhana yang mensimulasikan transaksi keuangan dari beberapa nasabah perbankan. Aplikasi ini mampu membuat lebih dari satu objek perorangan yang memiliki identitas (nama, umur) dan akun perbankan. Setiap objek perorangan dapat membuka 0 atau lebih akun bank. Setiap objek perorangan juga dapat melihat daftar rekening bank, beserta saldo untuk masing-masing akun bank yang dimilikinya. Melalui akun bank tersebut nantinya setiap objek bisa melakukan transaksi keuangan baik terhadap akun bank miliknya sendiri seperti menyetor uang, menarik uang, ataupun mentransfer uang kepada objek perorangan lainnya (penerima) yang telah memiliki akun bank.

Aplikasi ini masih belum sempurna, dikarenakan programmer yang membuatnya sudah tidak sanggup melanjutkannya karena memiliki pekerjaan lain yang lebih mendesak. Program ditulis dengan Bahasa java dan terdiri 2 file java saja yaitu file BankAccount.java dan sebuah file yang digunakan sebagai antarmuka simulasi yaitu BankAccountTest.java. Pada file BankAccount.java terdiri dari beberapa class di dalamnya ini dikarenakan si programmer sedikit malas memisahkannya menjadi file java tersendiri untuk setiap classnya.

Anda bisa mulai mendownload sourcecodenya pada google Classroom dan mengekstrak kedua file tersebut di computer anda (*disediakan juga project NetBeans -Ant-Java Application dalam format zip*). Selanjutnya jika Anda Menggunakan IDE selain NetBeans versi 12, maka untuk mengcompilennya bisa dengan Menggunakan perintah **javac** seperti berikut (*Pastikan currentpath anda berada dalam folder dimana file java tersebut berada*) :

```
nanoy@nanoyulian MINGW64 ~/OneDrive/Documents/TUTORIAL/JAVA-OOP/scripts/pset/pset3/BankAccount_incomplete_onlyfiles
$ javac -d . BankAccountTest.java BankAccount.java
Note: BankAccountTest.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
```

Nantinya akan terbentuk sebuah **folder bankaccount** akibat deklarasi package pada sourcecode. Untuk menjalankannya seperti biasa dengan menggunakan perintah **java** :

```
nanoy@nanoyulian MINGW64 ~/OneDrive/Documents/TUTORIAL/JAVA-OOP/scripts/pset/pset3/BankAccount_incomplete_onlyfiles
$ java bankaccount/BankAccountTest
==== Pilih Menu ====
1.Buat Objek Person!
2.Buat Akun Bank pada Objek Person!
3.Lihat daftar rekening dari objek Person!
4.Setor Uang ke Rekening dari objek Person
5.Tarik Uang dari Rekening objek Person
6.Transfer Uang!
7.Tampilkan daftar Objek Person!
=====
Pilih Menu no :
█
```

Aplikasi terlihat berjalan dengan baik pada awalnya, tidak terjadi masalah saat memilih menu 1. Buat Objek Person! Dan menu 2. Buat Akun Bank pada Objek Person!., tetapi saat memilih menu no. 3 – 7 akan memunculkan output yang tidak diharapkan.

Tugas Anda adalah melengkapi kekurangan pada sourcecode tersebut agar aplikasi simulasi ini bisa berjalan dengan benar dan mengupload sourcecode yang telah dilengkapi ke google

classroom. Sertakan juga hasil *screenshot* simulasi pada menu 3 – 7 yang telah dilengkapi. Sebagai simulasi buatlah minimal 2 objek yang masing-masing memiliki 2 akun bank. Terkait pengisian kode bank dibebaskan selama memiliki Panjang 3 digit. Dan Pastikan untuk no.rekening WAJIB 9 digit (sudah termasuk kode bank)!!. Untuk nama nasabah, umur, nama bank, dan jumlah uang saat pembukaan akun bank juga dibebaskan tetapi tolong saat menampilkan jumlah saldo formatnya dibatasi sampai 2 angka dibelakang koma! *Misalnya Rp. 30000,00*, karena saat ini keluaran pada aplikasi masih seperti ini (menu 7) :

```
Pilih Menu no :
7
Total Saldo Semua Rekening nano : Rp.0.0
1. [nama: nano] [umur: 56] [total saldo semua bank : Rp. 0.000000]
Total Saldo Semua Rekening doni : Rp.0.0
2. [nama: doni] [umur: 22] [total saldo semua bank : Rp. 0.000000]
```

(contoh output tampilan format uang yang perlu diperbaiki, seharusnya **Rp. 0,00**)

Perlu diketahui bahwa si programmer pernah memberikan pesan bahwa untuk melengkapi sourcecode ini cukup menambahkan **kode-kode pada body method** yang telah disediakan dan **Menggunakan method yang ada** dalam file BankAccount.java saja seperti **.getAkunBank(noRek)**. Programmer sebelumnya juga sudah memastikan bahwa pada **file BankAccountTest.java** dianggap telah selesai dan sementara tidak perlu dilakukan perubahan. Si programmer juga telah memberikan beberapa komentar pada sourcecodenya untuk lebih memudahkan Anda membaca alur program simulasi ini. Selain itu Dia berpesan bahwa akan sangat memudahkan jika sebelumnya Anda memiliki basic yang baik terkait *objek, class, interface, association, random, String, ArrayList yang memuat objek beserta methodnya, java.util.scanner, looping (for), return method*.

=====

Bonus Tambahan (tidak wajib, jika mampu)

Walaupun file BankAccountTest.java bukan menjadi target utama yang harus diperbaiki dan sudah berjalan dengan benar, tetapi kode di dalamnya terlalu panjang dan mungkin bisa diperpendek sehingga lebih mudah dibaca dan dikembangkan. Selain itu Anda juga dipersilahkan menambahkan validasi pengecekan jika saldo pada salah satu akun bank ternyata tidak mencukupi untuk melakukan transaksi penarikan, ataupun transfer. (*misalnya saldo sebesar Rp. 10000 tetapi melakukan transfer atau penarikan sebesar Rp. 20000 maka seharusnya transaksi akan dibatalkan*).

=====

...Have fun coding...