Kuis Pemrograman 1

Dasar-Dasar Pemrograman 1
CSGE601020
Semester Genap 2017/2018

Batas waktu pengumpulan:
Sabtu, 12 Mei 2018, 12:00 Waktu Scele

Tujuan dari Lab ini adalah melatih Anda agar menguasai bahan kuliah yang diajarkan di kelas. Mahasiswa diperbolehkan untuk berdiskusi, tetapi Anda tetap harus **menuliskan sendiri** solusi/kode program dari soal yang diberikan tanpa bantuan orang lain. Belajarlah menjadi mahasiswa yang mematuhi integritas akademik. **Sikap jujur merupakan sebuah sikap yang dimiliki mahasiswa UI.**

Peringatan:
Hindari mengumpulkan pekerjaan beberapa menit menjelang batas waktu pengumpulan karena ada kemungkinan pengumpulan gagal dilakukan atau koneksi Intenet terputus!

Bank Hogwarts

Di satu hari yang cerah, Sofia sedang menjalankan aktivitasnya sebagai manajer Bank Hogwarts di dunia paralel, kemudian ia mendapatkan surat dari bos-nya yang bernama Dipsi dengan isi sebagai berikut:

ソフィアさん

今日は　私が　忙しい　だから。私は今　ボゴール　へ　行って、子供たちが　大学試験　がある。私が　あそこ　いる間に　データ　を　チェックを　してくれませんか。あ。明日の　日本語能力試験　を　がんばって。

ディプシ

Terjemahan (Untuk yang penasaran):

Untuk Sofia

Hari ini saya sibuk, saya hari ini pergi ke Bogor, karena anak - anak sedang ada ujian masuk universitas. Selama saya pergi, tolong periksa data di sana ya. Oh iya, semangat ya untuk ujian bahasa jepangnya besok !

Dipsi

Isi surat tersebut mengatakan bahwa Sofia ditugaskan untuk memeriksa data di Bank Nasional Hogwarts. Sofia akhirnya mencoba menjelajahi cara kerja sistem bank tersebut. Ternyata, *prototype* dari cara kerja bank tersebut dapat diimplementasikan menggunakan Python dengan menggunakan *class* dan *method* sederhana. Sofia meminta bantuan Anda sebagai pegawai di divisi IT yang sangat mahir dalam melakukan pemrograman dengan Python untuk mengimplementasikan cara kerja sistem Bank Hogwarts. Bantulah Sofia!

Program akan mengolah *file* **.in** berdasarkan spesifikasi yang akan dijelaskan, lalu *output* langsung ditampilkan ke *console* (seperti Tugas Pemrograman 2). Setelah program membaca *file* dan mengeluarkan semua *output,* program akan mengeluarkan cetakan "**Terima Kasih Telah Menggunakan!**". Berikut spesifikasi *input* yang diterima oleh program:

- **DAFTAR <nama> <jenis tabungan> <saldo awal>**
  Mendaftarkan akun baru dengan pemilik, jenis akun, dan saldo awal yang diberikan oleh pendaftar.

- **SETOR <nama> <jumlah uang>**
  Menambahkan uang ke akun pengguna.

- **TARIK <nama> <jumlah uang>**
  Melakukan tarikan tunai dengan mengurangi saldo pengguna.

- **TRANSFER <nama pengirim> <nama penerima> <jumlah uang>**
  Melakukan transfer sebanyak yang diinginkan pengirim ke penerima.

- **INFO SEMUA**
  Mencetak semua nama nasabah beserta saldo terakhirnya.

- **INFO KEUANGAN**
  Mencetak total keuntungan dari bank Hogwarts

Bank mendapat keuntungan dari mana ? Silakan baca aturan di tabel yang ada di bawah

Validasi:

1. Saldo hanya boleh **0 <= saldo <= limit**. Jika saldo negatif, ubahlah menjadi nol.
2. Setiap transaksi dijamin selalu bilangan positif dan TIDAK ADA biaya transaksi.
3. Jika saldo awal saat DAFTAR < 5.000, cetak: **Maaf, Saldo Anda kurang!**. Jika saldo awal melebihi limit, cetak **Maaf, Saldo Anda melebihi kapasitas!**.
   Jika berhasil mendaftar, cetak:
   [Nama nasabah] telah terdaftar dengan paket [Nama Paket]
4. **Setiap nasabah dijamin memiliki nama yang unik dan tidak ada yang sama, dan operasi yang dilakukan oleh nama nasabah yang di tes dalam *file* .in dijamin ada! (tidak perlu memikirkan jika nama nasabah tidak terdaftar sebelumnya)**
5. Limit saldo dari setiap jenis tabungan adalah sebagai berikut:
   - Pelajar : 200.000
   - Regular : 250.000
   - Bisnis : 500.000
   - Elit : 1.000.000

Class yang dibutuhkan

<table>
<tr><td colspan="2">Nasabah
Sebuah class yang merepresentasikan data nasabah di Bank Hogwarts

• nama str : Nama nasabah
• jenis tabungan str : Jenis tabungan yang didaftarkan oleh nasabah
• Saldo int : Saldo yang disetorkan oleh nasabah</td></tr>
<tr><td>setor()</td><td>Sebuah method yang menerima 1 buah parameter jumlah uang yang hendak disetorkan. Dijalankan dari perintah SETOR

Saat method ini dijalankan, jika total uang setelah di setor melebihi limit, SETOR tetap berhasil, tetapi saldo yang akan masuk ke pengguna hanya sebagian (hingga limit).

Contoh: saldo sekarang 175.000, limit saldo 200.000, setor 100.000, maka yang masuk hanya akan 25.000, sedangkan 75.000 sisanya menjadi keuntungan bank

Jika berhasil, cetak dengan:
Akun telah bertambah sebesar [Jumlah Uang]
Jumlah uang adalah jumlah uang bersih yang masuk, dengan contoh kasus di atas akan mencetak:
Akun telah bertambah sebesar 25000</td></tr>
<tr><td>tarik()</td><td>Sebuah method yang menerima 1 buah paramer berupa jumlah uang yang hendak di tarik. Dijalankan dari perintah TARIK

Saat dijalankan, jika jumlah uang yang di tarik > saldo, cetak:
Transaksi gagal! Saldo tidak cukup!
dan tidak ada perubahan pada saldo
==========
Jika berhasil, cetak:
Berhasil menarik sebesar [Jumlah Uang]</td></tr>
<tr><td>transfer()</td><td>Sebuah method yang menerima 2 buah parameter yaitu nama penerima & uang yang mau di transfer. Dijalankan dari perintah TRANSFER.

Saat dijalankan, jika uang yang ditransfer < saldo pengirim, cetak:
Transaksi gagal! Saldo tidak cukup!</td></tr>
</table>

	Apabila memungkinkan untuk ditransfer, **TETAPI** saldo penerima + uang ditransfer melebihi limit akun penerima, maka transfer TETAP BERHASIL, uang pengirim berkurang, uang penerima bertambah hingga mencapai limit, dan **sisa uang yang tidak masuk akan menjadi keuntungan bank**. Contoh: Saldo pengirim : 100.000 Saldo penerima : 150.000 , dan limit saldo penerima : 200.000 **Pengirim hendak transfer 75.000, transaksi pun sukses.** Saldo pengirim berkurang 75.000 Saldo penerima hanya bertambah 50.000 Sisa 25.000 menjadi keuntungan bank Jika uang yang di transfer <= saldo pengirim DAN saldo penerima + uang diterima <= limit penerima, cetak: **Berhasil transfer sebesar [Jumlah Uang] kepada [Penerima]**

infoSemua()	Sebuah *method* yang tidak menerima parameter (cukup *self* saja), *method* ini akan mencetak semua saldo akhir dari para nasabah dari keyword INFO SEMUA **Cetak dengan:** **[Nama Nasabah-1] [Saldo Nasabah-1]** **[Nama Nasabah-2] [Saldo Nasabah-2]** **[Nama Nasabah-3] [Saldo Nasabah-3]** **...** Dan seterusnya sebanyak jumlah nasabah yang telah didaftarkan
infoUang()	Sebuah *method* yang tidak menerima parameter (cukup *self* saja), *method* ini akan mencetak total keuntungan bank dari keyword **INFO KEUANGAN** **Cetak dengan:** **BANK Mendapat Keuntungan sebesar [Jumlah uang]**

Template tidak wajib untuk digunakan, tetapi disarankan untuk digunakan demi mempermudah pengerjaan

omponen Penilaian

- 40% : *Output* sesuai yang diharapkan
- 30% : Mengimplementasikan seluruh hal yang dibutuhkan program dengan baik
- 20% : Penggunaan tipe data yang tepat
- 5% : Kerapihan kode
- 5% : Dokumentasi & Penamaan file