

3. Simple Store App

Jadi akan dibuat aplikasi dimana terdapat *user* dan *store* untuk melakukan pembelian *produk*.

User

- a. Kelas User memiliki attribut ***name (String)*** untuk menampung nama pengguna dan ***balance (int)*** untuk menampung uang milik user.
- b. Terdapat ***getter*** untuk mengakses masing-masing attribute
- c. Terdapat ***constructor*** untuk menambahkan nilai ke semua attribut pada saat instance objek.

Product

- a. Kelas Product memiliki attribut ***name (String)*** untuk menampung nama produk, ***price (int)*** untuk menampung harga produk, dan ***stock (int)*** untuk menampung jumlah stock produk
- b. Terdapat ***getter*** untuk mengakses masing-masing attribute
- c. Terdapat ***behavior*** untuk mengurangi jumlah stock sebesar satu saat produk terjual
- d. Instance objek produk dapat dilakukan dengan dua cara :
 - 1) Constructor biasa untuk semua attribute sebagai parameter dengan tipe data yang sama
 - 2) Constructor dimana semua attribute sebagai parameter namun attribute price paramaternya berupa String, dengan conversi inputan "k" = 1000. Misalnya, "1k" maka akan tersimpan di attribute sebagai 1000

Store

- a. Kelas Store memiliki attribute ***name (String)*** untuk menampung nama toko, dan ***products*** untuk menampung list produk.
- b. Terdapat Constructor dengan parameter name, sehingga nama toko akan langsung ditambahkan saat objek di instance

- c. Terdapat behavior untuk menambahkan satu produk ke attribut products
- d. Terdapat behavior untuk menampilkan list produk yang tersedia di toko tersebut
- e. Terdapat behavior untuk memproses transaksi, jadi di uang user akan berkurang sesuai harga produk, stok produk akan berkurang satu, dan jika stok produk sisa 0 maka produk akan dihapus dari list produk. Silahkan handle pula saat uang user tidak cukup untuk membeli produk.

Instance Objek dilakukan di Main Method, yang dibuat di kelas tersendiri (tidak boleh include di kelas user, product, ataupun store).

Untuk mengakses attribute tiap kelas harus menggunakan method getter yang telah dibuat dan untuk mengatur nilai harus menggunakan constructor, setter, ataupun behavior lain. Intinya tidak diperkenankan mengakses attribute secara langsung seperti `user.name = "sample"` ataupun `System.out.println(user.name)`.

Untuk contoh tampilan program seperti berikut:

1. Pertama akan muncul detail uang user, detail store, dan daftar produk yang tersedia

```
-----CARD-----
Ikhsan
3000
-----
Selamat Datang di Sisfo Computer
Daftar Produk:
1. HP Pavilion Gaming 15 - $2000 | Stok 15
2. ASUS VivoBook 15 A516MAO N4020 - $4000 | Stok 4
3. Lenovo IdeaPad S145 - $1500 | Stok 2
0. Keluar
> 
```

2. Ada konfirmasi sebelum membeli produk

```
Apakah anda yakin ingin membeli:  
HP Pavilion Gaming 15, dengan Harga $2000  
1. Konfirmasi  
2. Batal  
> 1
```

3. Akan muncul status pembelian apakah berhasil atau tidak, disertai pertanyaan apakah lanjut atau keluar

```
Ikhsan berhasil membeli HP Pavilion Gaming 15.  
-----  
Lanjutkan Transaksi?  
1. Iya  
2. Tidak  
> 1
```

4. Saat memilih lanjut maka akan muncul kembali halaman pertama dengan jumlah balance berkurang sesuai dengan harga produk yang dibeli
5. Saat stok produk habis maka tidak akan muncul di daftar produk store.

Silahkan tambahkan komentar ke kode untuk memperjelas tiap bagian.

Silahkan dikerjakan semampunya jangan memaksakan diri, jika tidak paham langsung asistensi kan saja