

## Soal Lab 8

### Kiospedia

Pada Lab kali ini, Anda diminta untuk menggunakan salah satu konsep pada OOP yaitu *Polymorphism*. Anda diminta untuk mengimplementasikan beberapa kelas dengan konsep tersebut yang juga memiliki *overriding method* dan *overloading method*.

Anda saat ini membangun sebuah sistem *e-commerce* sederhana bernama Kiospedia. Kiospedia merupakan salah satu *e-commerce* yang menggunakan *text-based* dalam pelayanannya. Kiospedia menyediakan berbagai macam jenis barang yang dijual. Terdapat beberapa kategori barang yang tersedia di Kiospedia, yaitu barang Elektronik, Pakaian, serta Makanan. Dengan memanfaatkan ilmu DDP2, Anda akan membuat programnya secara *object oriented* dengan menggunakan bahasa pemrograman Java dan memanfaatkan konsep *polymorphism*.

Anda memberikan promo diskon pada barang-barang yang dijual di Kiospedia. Akan tetapi, diskon yang diberikan berbeda-beda untuk setiap kategori barang yang dijual. Berikut deskripsi besar diskon untuk setiap kategori barang:

Item	Besar Diskon
Elektronik	15%
Pakaian	10%
Makanan	5%

Dengan begitu, ketika *user* melakukan *checkout* pembelian, maka harga tiap barang akan dipotong sesuai dengan diskon yang tertera di atas.

Selain dengan promo diskon di atas, *user* juga bisa menggunakan kode promo yang akan memotong harga sesuai persen kode promo. Akan tetapi, jika *user* menggunakan kode promo, maka promo diskon yang di atas tidak lagi berlaku.

Berikut adalah rumus untuk menghitung harga setelah diskon:

$\text{harga diskon} = \text{harga semula} - (\% \text{diskon} / 100) \times \text{harga semula}$
---



### Format Masukan

Masukan dibaca melalui input standar hingga **End of File (EOF)**, di mana setiap masukan berupa perintah-perintah sebagai berikut:

Format Perintah	Keterangan
<b>TAMBAH</b> <spasi><kategori barang><spasi><nama barang><spasi><harga normal>	<p>Menambahkan barang ke dalam <i>cart</i> belanjaan. Setiap barang memiliki kategori, nama, serta harga normal sebelum diskon.</p> <p>Kategori barang hanya salah satu dari tiga kategori berikut: Elektronik, Pakaian, dan Makanan.</p> <p>Contoh: <b>TAMBAH Elektronik Laptop 5000000</b> (menambahkan barang bernama Laptop dengan kategori Elektronik seharga 5000000 sebelum diskon)</p>
<b>DETAIL</b>	Menampilkan detail barang-barang yang ada pada <i>cart</i> belanjaan.
<b>CHECKOUT</b>	Melakukan proses pembelian semua barang yang ada pada <i>cart</i> belanjaan serta mengosongkan isi <i>cart</i> .
<b>CHECKOUT</b> <spasi><besar diskon>	<p>Melakukan proses pembelian semua barang yang ada pada <i>cart</i> belanjaan dengan menggunakan kode promo dengan besar diskon tertentu (dalam persen) serta mengosongkan isi <i>cart</i>.</p> <p>Contoh: <b>CHECKOUT 20</b> (membeli semua barang pada <i>cart</i>, semua harga barang akan dipotong diskon sebesar 20%, contohnya jika harga barang 50000, maka menjadi 40000)</p>
<b>CHECKOUT</b> <spasi> <b>SORTED</b>	Melakukan proses pembelian semua barang secara terurut secara leksikografis dari nama barangnya yang ada pada <i>cart</i> belanjaan serta mengosongkan isi <i>cart</i> .



## Format Keluaran

<b>TAMBAH</b>	<p>Jika kategorinya Elektronik:  <b>"Menambahkan &lt;nama barang&gt; seharga &lt;harga normal&gt; biar canggih!"</b></p> <p>Jika kategorinya Pakaian:  <b>"Menambahkan &lt;nama barang&gt; seharga &lt;harga normal&gt; agar tampil lebih modis!"</b></p> <p>Jika kategorinya Makanan:  <b>"Menambahkan &lt;nama barang&gt; seharga &lt;harga normal&gt;. Enak!"</b></p>
<b>DETAIL</b>	<p>Mencetak secara iteratif detail setiap barang yang ada pada <i>cart</i> dengan format sebagai berikut:</p> <p>Jika kategorinya Elektronik:  <b>"&lt;nama barang&gt; adalah peralatan elektronik. Mendapat diskon 15%."</b></p> <p>Jika kategorinya Pakaian:  <b>"Pakaian &lt;nama barang&gt; dengan diskon 10%."</b></p> <p>Jika kategorinya Makanan:  <b>"&lt;nama barang&gt; bisa dimakan. Diskon 5%."</b></p>
<b>CHECKOUT</b>	<p>Mencetak secara iteratif setiap barang yang ada pada <i>cart</i> dengan format sebagai berikut:</p> <p><b>&lt;nama barang&gt; berhasil dibeli. Harga: &lt;harga setelah diskon&gt;.</b>  <b>&lt;nama barang&gt; berhasil dibeli. Harga: &lt;harga setelah diskon&gt;.</b>  ...  <b>&lt;nama barang&gt; berhasil dibeli. Harga: &lt;harga setelah diskon&gt;.</b>  <b>--- Total harga: &lt;total seluruh harga setelah diskon&gt;</b></p>
<b>CHECKOUT &lt;besar diskon&gt;</b>	<p>Mencetak secara iteratif setiap barang yang ada pada <i>cart</i> dengan format sebagai berikut:</p> <p><b>&lt;nama barang&gt; berhasil dibeli. Harga: &lt;harga setelah diskon&gt;.</b>  <b>&lt;nama barang&gt; berhasil dibeli. Harga: &lt;harga setelah diskon&gt;.</b>  ...  <b>&lt;nama barang&gt; berhasil dibeli. Harga: &lt;harga setelah diskon&gt;.</b>  <b>--- Total harga: &lt;total seluruh harga setelah diskon&gt;</b></p>
<b>CHECKOUT SORTED</b>	<p>Mencetak secara iteratif dan terurut secara leksikografis</p>



	<p>dari namanya untuk setiap barang yang ada pada <i>cart</i> dengan format sebagai berikut:</p> <p>Contoh dengan barang pada <i>cart</i> memiliki nama: [Mangga, Handphone, Jas]</p> <p><b>Handphone berhasil dibeli. Harga: &lt;harga setelah diskon&gt;.</b></p> <p><b>Jas berhasil dibeli. Harga: &lt;harga setelah diskon&gt;.</b></p> <p><b>Mangga berhasil dibeli. Harga: &lt;harga setelah diskon&gt;.</b></p> <p><b>--- Total harga: &lt;total seluruh harga setelah diskon&gt;</b></p>
--	--

### Contoh Masukan

```

TAMBAH Elektronik TV 1530000
TAMBAH Pakaian Baju 80000
TAMBAH Pakaian Celana 65000
TAMBAH Makanan Indomie 85000
TAMBAH Elektronik Handphone 1970000
DETAIL
CHECKOUT
TAMBAH Makanan Samyang 530000
TAMBAH Pakaian FlareSkirt 87500
TAMBAH Pakaian Blazer 165000
TAMBAH Makanan Kurma 60000
TAMBAH Elektronik PS4 3800000
CHECKOUT SORTED
TAMBAH Makanan Ovomaltine 50000
TAMBAH Pakaian Batik 109000
TAMBAH Elektronik NintendoSwitch 6450000
CHECKOUT 50

```

### Contoh Keluaran

```

Menambahkan TV seharga 1530000.0 biar canggih!
Menambahkan Baju seharga 80000.0 agar tampil lebih modis!
Menambahkan Celana seharga 65000.0 agar tampil lebih modis!
Menambahkan Indomie seharga 85000.0. Enak!
Menambahkan Handphone seharga 1970000.0 biar canggih!
TV adalah peralatan elektronik. Mendapat diskon sebesar 15%.
Pakaian Baju dengan diskon 10%.
Pakaian Celana dengan diskon 10%.
Indomie bisa dimakan. Diskon 5%.
Handphone adalah peralatan elektronik. Mendapat diskon sebesar 15%.
TV berhasil dibeli. Harga: 1300500.0.
Baju berhasil dibeli. Harga: 72000.0.
Celana berhasil dibeli. Harga: 58500.0.
Indomie berhasil dibeli. Harga: 80750.0.
Handphone berhasil dibeli. Harga: 1674500.0.
--- Total harga: 3186250.0
Menambahkan Samyang seharga 530000.0. Enak!
Menambahkan FlareSkirt seharga 87500.0 agar tampil lebih modis!
Menambahkan Blazer seharga 165000.0 agar tampil lebih modis!
Menambahkan Kurma seharga 60000.0. Enak!

```





```
Menambahkan PS4 seharga 3800000.0 biar canggih!
Blazer berhasil dibeli. Harga: 148500.0.
FlareSkirt berhasil dibeli. Harga: 78750.0.
Kurma berhasil dibeli. Harga: 57000.0.
PS4 berhasil dibeli. Harga: 3230000.0.
Samyang berhasil dibeli. Harga: 503500.0.
--- Total harga: 4017750.0
Menambahkan Ovomaltine seharga 50000.0. Enak!
Menambahkan Batik seharga 109000.0 agar tampil lebih modis!
Menambahkan NintendoSwitch seharga 6450000.0 biar canggih!
Ovomaltine berhasil dibeli. Harga: 25000.0.
Batik berhasil dibeli. Harga: 54500.0.
NintendoSwitch berhasil dibeli. Harga: 3225000.0.
--- Total harga: 3304500.0
```

### Petunjuk Pengerjaan

Terdapat *template* program berupa *class* utama beserta *class-class* lain yang diperlukan. Tugas Anda adalah mempelajari jalannya program pada *class* utama, melengkapi semua yang dibutuhkan pada *class-class* yang belum lengkap. Anda diminta untuk **TIDAK MENGUBAH KODE PADA CLASS UTAMA**. Silakan gunakan konsep *method overriding* dan *method overloading* untuk memberikan perlakuan yang berbeda dari hasil pemanggilan *method* pada *class* utama.

Untuk membuat objek agar bisa di-sort. Lakukan:

- *Implements* interface *Comparable<T>* pada *class parent*.
- Buat *method* *compareTo(T other)* pada *class* tersebut.
- Implementasikan *compareTo* dengan melakukan *compareTo* pada namanya.

Jangan lupa melakukan *extend* ke *class parent* pada setiap *class* yang membutuhkan.

### Format Pengumpulan

- **KELAS\_NPM\_TUTORIAL8.zip** contoh: B\_1606123456\_TUTORIAL8.zip
- Isi file KELAS\_NPM\_TUTORIAL8.zip:
  - Lab8.java (main)
  - Item.java
  - Elektronik.java
  - Pakaian.java
  - Makanan.java
  - *File-file* lain yang dibutuhkan

