

Soal Tugas Praktikum Modul 12

Dasar Pemrograman Berorientasi Objek – CCK1IAB3

Collection

Petunjuk Umum

1. Tugas praktikum bersifat individu, sehingga praktikan tidak diperbolehkan untuk berdiskusi satu sama lain, berdiri, pindah tempat duduk ataupun membuat kegaduhan selama praktikum.
2. Tugas praktikum modul ini terdiri dari sesi terbimbing, mohon membaca teknis dengan baik!
 - Asisten Praktikum (AsPrak) memberikan tutorial di depan kelas.
 - Praktikan mengikuti dan bertanya terkait tutorial yang diberikan
 - Praktikan mengerjakan soal terbimbing lainnya dibantu oleh AsPrak, baik secara logika maupun secara sintak.
3. Pengumpulan tugas melalui LMS mengikut petunjuk yang diberikan oleh asisten praktikum.

Jurnal

Mari kita menulis kelas yang berhubungan dengan buku menggunakan konsep pewarisan dan membuat array menggunakan polimorfisme.

Dan mari kita buat sistem untuk mencari informasi buku menggunakan array buku.

Sample Run

```
=== << Program manajemen pelanggan >> ===
1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem
=====
## input menu : 1
----- Informasi pelanggan -----
1. nama : Lee  usia : 28  No telepon : [HP] 010-5678-1234
2. nama : Park  usia : 31  No telepon : [HP] 010-4567-9876
3. nama : Choi  usia : 25  No telepon : [OFFICE] 010-1111-2222
-----
=== << Program manajemen pelanggan >> ===
1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem
=====
## input menu : 2
-----
tambahkan informasi pelanggan baru.
masukkan informasi pelanggan baru
-----
> nama : Hong
> usia : 27
> No telepon [HP] : 010-4679-8520
> No telepon [OFFICE] : 010-4679-8522
> No perwakilan (HP/OFFICE) : OFFICE
@ informasi pelanggan sudah ditambahkan
-----
```

=== << Program manajemen pelanggan >> ===

1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem

=====

input menu : 1

----- informasi pelanggan -----

1. nama : Lee usia : 28 No telepon : [HP] 010-5678-1234
2. nama : Park usia : 31 No telepon : [HP] 010-4567-9876
3. nama : Choi usia : 25 No telepon : [OFFICE] 010-1111-2222
4. nama : Hong usia : 27 No telepon : [OFFICE] 010-4679-8522

=== << program manajemen pelanggan >> ===

1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem

=====

input menu : 3

edit usia pelanggan.

- > nomor pelanggan yang akan di edit : 2
- > informasi usia yang akan di edit : 34
- @ informasi pelanggan berhasil diubah.

=== << Program manajemen pelanggan >> ===

1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem

=====

input menu : 1

----- informasi pelanggan -----

1. nama : Lee usia : 28 No telepon : [HP] 010-5678-1234
2. nama : Park usia : 34 No telepon : [HP] 010-4567-9876
3. nama : Choi usia : 25 No telepon : [OFFICE] 010-1111-2222
4. nama : Hong usia : 27 No telepon : [OFFICE] 010-4679-8522

=== << Program Manajemen Pelanggan >> ===

1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem

=====

Input menu : 4

ubah No telepon pelanggan.

> Nomor pelanggan yang akan di edit : 3
> No telepon yang akan di edit (HP) : 010-8888-7777
> No telepon yang akan di edit (OFFICE) : 010-1111-7788
> No perwakilan (HP/OFFICE) : HP
@ informasi pelanggan telah diubah.

=====

=== << Program manajemen pelanggan >> ===

1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem

=====

input menu : 1

----- informasi pelanggan -----

1. nama : Lee	usia : 28	No telepon : [HP] 010-5678-1234
2. nama : Park	usia : 34	No telepon : [HP] 010-4567-9876
3. nama : Choi	usia : 25	No telepon : [HP] 010-8888-7777
4. nama : Hong	usia : 27	No telepon : [OFFICE] 010-4679-8522

=== << Program manajemen pelanggan>> ===

1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem

=====

input menu : 5

Menghapus informasi pelanggan.

> Nomor pelanggan yang akan dihapus : 1

@ informasi pelanggan telah dihapus.

=== << Program Manajemen Pelanggan >> ===

1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit No telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem

=====

Input menu : 1

----- informasi pelanggan -----

1. nama : Park usia : 34 No telepon : [HP] 010-4567-9876
2. nama : Choi usia : 25 No telepon : [HP] 010-8888-7777
3. nama : Hong usia : 27 No telepon : [OFFICE] 010-4679-8522

=== << Program Manajemen Pelanggan >> ===

1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem

=====

Input menu : 9

End. Bye~ Bye~

Sample Run (Kasus)

Note: *Tidak menggunakan exception

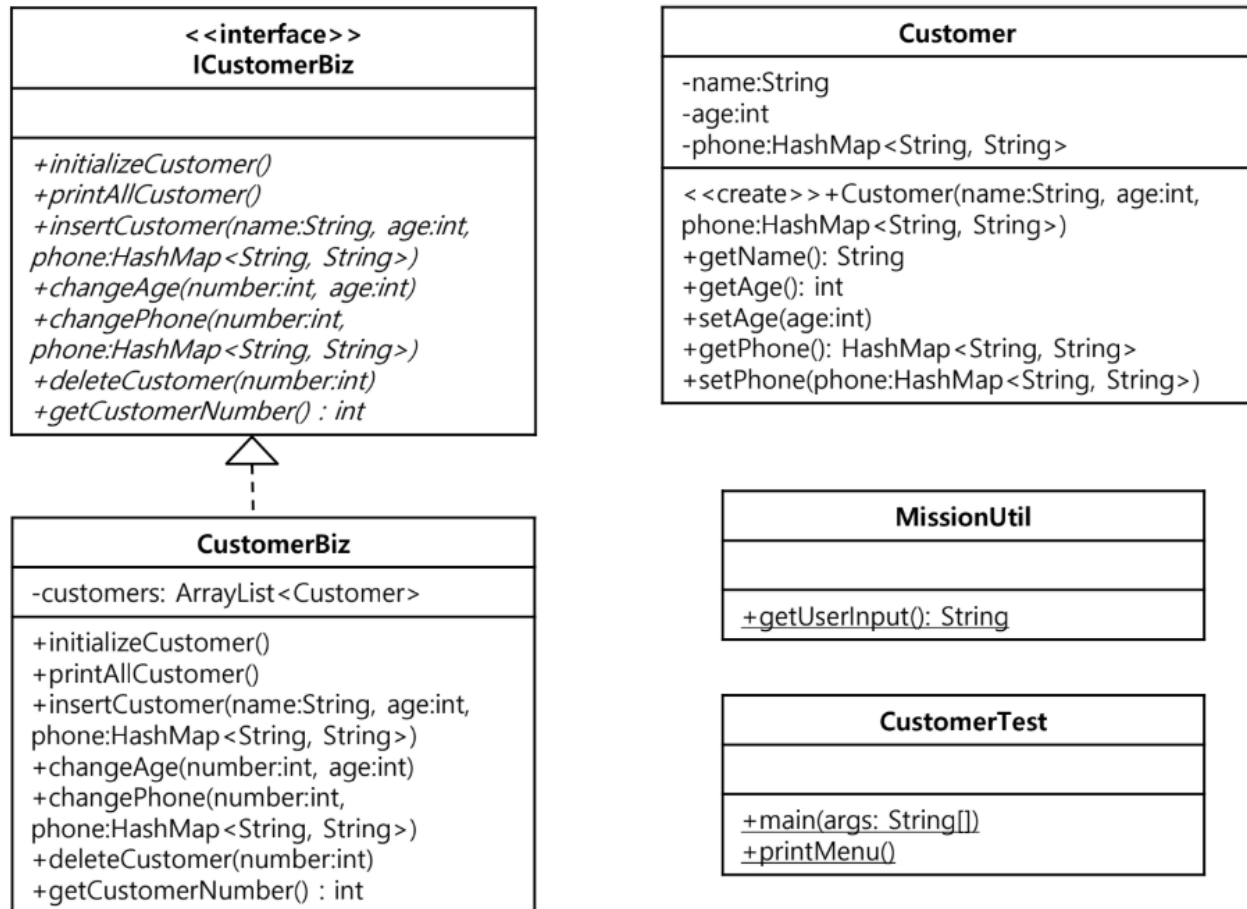
```
== << Program Manajemen Pelanggan >> ==
1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem
=====
## Input menu : 3
-----
edit usia pelanggan.
-----
> Nomor pelanggan yang akan di edit : 5
[Error] tidak bisa menemukan nomor pelanggan
yang sesuai.
```

```
== << Program Manajemen Pelanggan >> ==
1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem
=====
## Input menu : 3
-----
edit usia pelanggan.
-----
> No informasi pelanggan yang akan di edit : 2
> Informasi usia yang akan diubah : -6
[error] usia tidak bisa di input negative.
```

```
=== << Program Manajemen Pelanggan >> ===
1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem
=====
## Input menu : 4
-----
edit No telepon pelanggan.
-----
> No informasi pelanggan yang akan di ubah : -5
[Error] tidak bisa menemukan nomor pelanggan
yang sesuai.
```

```
=== << Program Manajemen Pelanggan >> ===
1. Mendapat informasi semua pelanggan
2. Tambahkan informasi pelanggan
3. Edit usia pelanggan
4. Edit nomor telepon pelanggan
5. Hapus informasi pelanggan
9. Keluar sistem
=====
## Input menu : 5
-----
hapus informasi pelanggan.
-----
> Nomor pelanggan yang akan dihapus : 9
[error] tidak bisa menemukan nomor pelanggan
yang sesuai.
```

Class Diagram



Customer.java: Class yang mengandung informasi pelanggan.

1. Nama package adalah "customer"
2. Field, constructor, dan getter/setter mengacu pada Class Diagram

MissionUtil.java: Utility yang digunakan

1. Nama package adalah "util"
2. get userInput() : Metode untuk mengembalikan nilai yang dimasukkan melalui standard input (keyboard)
(jika anda perlu memasukkan angka, gunakan metode ini tanpa membuat metode terpisah.)

ICustomerBiz.java: Interface yang mendefinisikan method yang dibutuhkan

1. Nama package adalah "customer.biz"
2. Mengacu pada Class Diagram untuk membuat abstract method untuk fungsi search, add, modify, dan delete informasi customer.

CustomerBiz.java: Class yang berisikan logic yang diperlukan

1. Nama package adalah “customer.biz”
2. Field mengacu pada Class Diagram
3. `InitializaCustomer()`: metode untuk mengatur data informasi pelanggan awal di field `ArrayList`.
 - Buat objek field `ArrayList`.
 - Mengacu pada sample data, tambahkan informasi pelanggan ke `ArrayList` secara berurutan. Pada fase ini No telepon dan informasi terkait dikelola dalam format `HashMap`. Pada `HashMap` terdapat tiga Key yang digunakan adalah No telepon(HP), No telepon(OFFICE), No perwakilan(HP/OFFICE), nilai dari Key bisa ditentukan secara bebas.
4. `printAllCustomer()`: metode untuk mencetak semua informasi pelanggan yang terdapat dalam `ArrayList`.
 - Gunakan loop, lalu cetak informasi nama, usia dan No telepon seluruh pelanggan yang terdapat pada kolom Array List.
 - Saat mencetak, lihat Sample Run’ dan sesuaikan format agar outputnya sama.
 - Menampilkan nomor yang ditunjuk sebagai nomor perwakilan di antara informasi nomor telepon yang disimpan di `HashMap`.
5. `insertCustomer(String name, int age, HashMap<String, String> phone)` : tambahkan method informasi pelanggan baru.
 - Buatlah objek informasi pelanggan lalu tambahkan ke field `ArrayList`.
6. `changeAge(int number, int age)` : Method untuk mengganti usia dengan informasi yang diinput oleh user.
 - Ubahlah usia dari customer yang sesuai dengan nomor yang diterima sebagai Parameter.
7. `changePhone(int number, HashMap<String, String> phone)` : Method untuk mengubah No telepon dengan informasi yang diinput oleh pengguna.
 - Ubah No telepon pelanggan sesuai dengan nomor yang diterima sebagai Parameter.
8. `deleteCustomer(int number)` : Method untuk menghapus isi informasi pelanggan yang diinput user.
 - Hapus pelanggan yang sesuai dengan nomor yang diterima sebagai parameter dari `ArrayList`.
9. `getCustomerNumber()` : Metode mengembalikan jumlah pelanggan yang terdapat dalam `ArrayList`
 - Mengembalikan nilai dengan menghitung jumlah pelanggan yang terdapat dalam `ArrayList`.

CustomerTest.java: Class untuk mengeluarkan menu Test.

1. Nama package adalah “customer.test”
2. `Main()`: method yang memproses menu dan memanggil masukkan dari user.
 - Jika tidak ada nomor yang sesuai dengan pelanggan, pesan error akan ditampilkan dengan mengacu pada setiap kasus.

3. buat objek CustomerBiz dengan ICustomerBiz sebagai tipe data referensi dan panggil metode untuk menginisialisasi informasi pelanggan.
4. Gunakan loop untuk memungkinkan pemilihan menu berlanjut hingga menu nomor 9 diinput.
5. Jika menu nomor 1 dimasukkan, metode yang sesuai dipanggil untuk menanyakan seluruh informasi pelanggan.
6. Jika menu nomor 2 dimasukkan, tambahkan Method untuk menambahkan informasi pelanggan yang diterima oleh user.
 - Jika ada parameter yang diperlukan saat memanggil Method, User akan menerima input.
 - Jika usia di masukan nomor negative, akan muncul pesan error mengacu pada Sample Run.
7. Jika menu nomor 3 diinput, maka Method akan dipanggil untuk mengubah usia pelanggan.
 - Jika ada parameter yang diperlukan saat memanggil suatu metode, input diterima dari user.
 - Jika tidak ada nomor yang sesuai dengan pelanggan, pesan kesalahan ditampilkan dengan pengoperasian Sample Run.
 - Jika usia diinput angka negative, pesan kesalahan ditampilkan mengacu pada sample run.
8. Jika menu nomor 4 diinput, maka Method akan dipanggil untuk mengubah No telepon pelanggan.
 - Jika ada parameter yang diperlukan saat memanggil Method, masupada Sample Run.
9. Jika menu nomor 5 diinput, metode untuk menghapus informasi pelanggan dipanggil.
 - Jika ada parameter yang diperlukan saat memanggil suatu metode, input akan diterima dari pengguna.
 - Jika tidak ada nomor yang sesuai dengan pelanggan, pesan eror akan muncul mengacu pada Sample Run.
10. Jika menu nomor 9 diinput, pesan yang sesuai akan dikeluarkan dan program berakhir. (Lihat Sample Run)
11. Jika angka selain 1~5 atau 9 diinput, pesan error akan muncul mengacu pada Sample Run.
 - Setelah mencetak pesan eror, menu tersebut muncul kembali sehingga anda dapat memilih menu baru.
12. printMenu(): Method untuk mencetak menu.

Sample Data

Data nomor telepon dikelola dengan HashMap

nama	usia	No telepon [HP]	No telepon [OFFICE]	No perwakilan	catatan
Lee	28	010-5678-1234	010-2132-9876	HP	
Park	31	010-4567-9876	010-1234-5678	HP	
Choi	25	010-8888-9999	010-1111-2222	OFFICE	
Hong	27	010-4679-8520	010-4679-8522	OFFICE	Data saat menambahkan pelanggan