

Soal Tugas Praktikum Modul 11

Dasar Pemrograman Berorientasi Objek – CCK1IAB3

Assesment / Case Based III

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 assesment atau mandiri, mohon membaca teknis dengan baik!
 - Praktikan mengerjakan soal secara mandiri tanpa dibantu ataupun dibimbing oleh asisten praktikum.
 - Asisten praktikum hanya diperbolehkan membantu praktikan apabila ada error yang disebabkan oleh kesalahan sintak (bukan logika).
3. Pengumpulan tugas melalui LMS mengikut petunjuk yang diberikan oleh asisten praktikum.

Assesment / Case Based III

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 Informasi Buku >> =====
1. Mencari seluruh informasi buku
2. Mencari nama penulis novel
3. Mencari harga novel (harga terendah ke harga tertinggi)
9. Keluar
=====
## Pilih menu: 1

----- Informasi Buku -----
[Judul] Cooking Light           [Harga] 15,000 won           [Kategori] living, cooking [Catatan] America Cooking
[Judul] Auto Bild               [Harga] 16,000 won           [Kategori] science, car    [Catatan] Germany Car
[Judul] The Confession          [Harga] 10,500 won           [Penulis] Grisham, John
[Judul] Les Miserables          [Harga] 17,500 won           [Penulis] Hugo, Victor
[Judul] Breakthrough            [Harga] 47,200 won           [Penulis] Ifill, Gwen
[Judul] The Racketeer           [Harga] 38,000 won           [Penulis] Grisham, John

===== << Program Informasi Buku >> =====
1. Mencari seluruh informasi buku
2. Mencari nama penulis novel
3. Mencari harga novel (harga terendah ke harga tertinggi)
9. Keluar
=====
## Pilih menu : 2
>> Masukkan nama penulis yang ingin dicari : Grisham, John

----- Informasi Buku -----
[Judul] The Confession          [Harga] 10,500 won           [Penulis] Grisham, John
[Judul] The Racketeer           [Harga] 38,000 won           [Penulis] Grisham, John
-----
```

```
===== << Program Informasi Buku >> =====
1. Mencari seluruh informasi buku
2. Mencari nama penulis novel
3. Mencari harga novel (harga terendah ke harga tertinggi)
9. Keluar
=====
## Pilih menu : 3
>> Masukkan harga minimum untuk novel yang ingin dicari : 15000
>> Masukkan harga maksimum untuk novel yang ingin dicari : 40000
----- Informasi Buku -----
[Judul] Les Miserables          [Harga] 17,500 won          [Penulis] Hugo, Victor
[Judul] The Racketeer           [Harga] 38,000 won          [Penulis] Grisham, John
-----
===== << Program Informasi Buku >> =====
1. Mencari seluruh informasi buku
2. Mencari nama penulis novel
3. Mencari harga novel (harga terendah ke harga tertinggi)
9. Keluar
=====
## Pilih menu : 9
>> Berhenti. Bye~
```

Sample Run (Kasus 1) : Salah masukkan nomor menu

Note: *Kasus 1-3 tidak menggunakan exception

```
===== << Program Informasi Buku >> =====
1. Mencari seluruh informasi buku
2. Mencari nama penulis novel
3. Mencari harga novel (harga terendah ke harga tertinggi)
9. Keluar
=====
## Pilih menu : 7
[Error] Anda salah memasukkan menu. Silakan masukkan lagi.
```

Sample Run (Kasus 2): Nilai min/maks dimasukkan sebagai angka negative

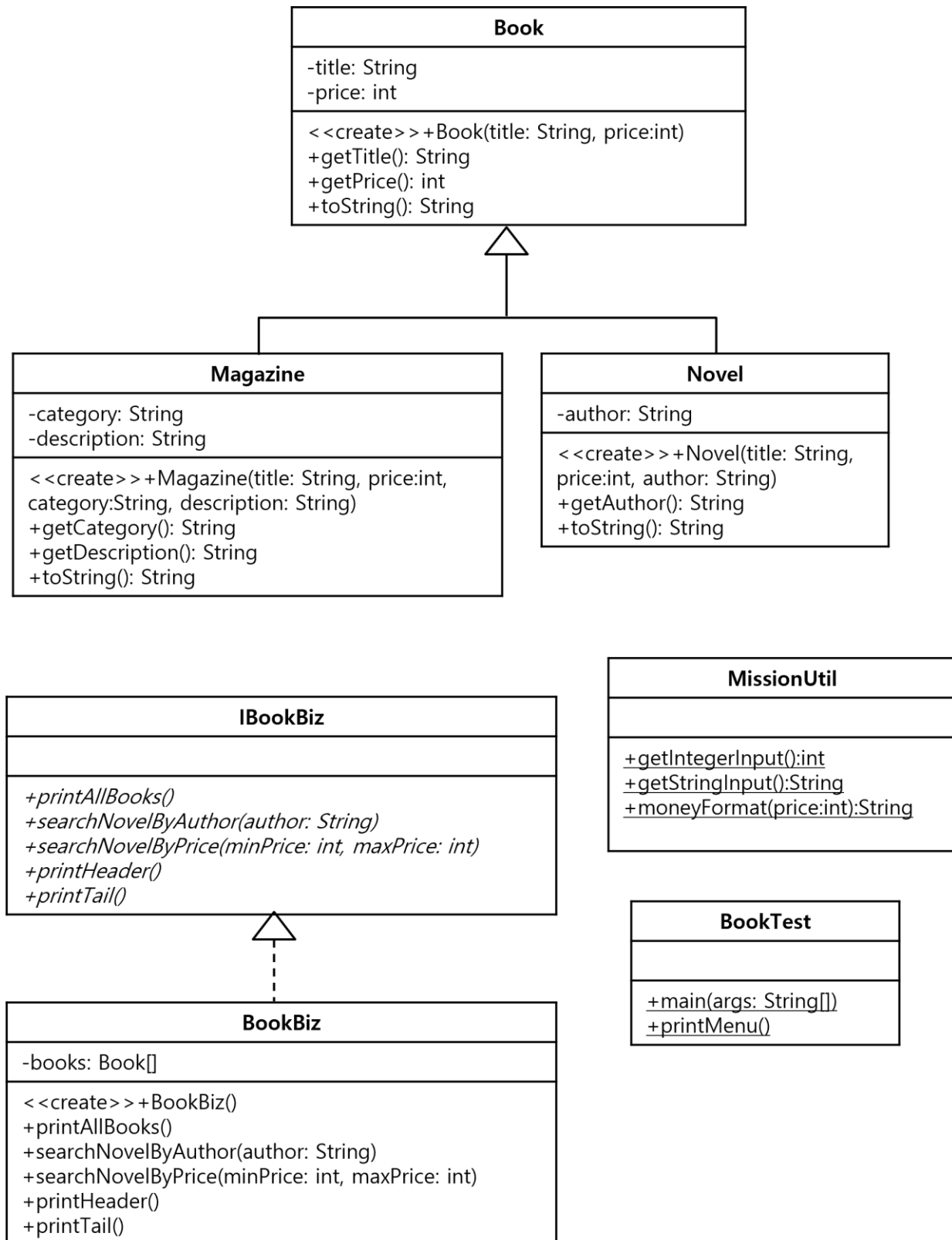
```
===== << Program Informasi Buku >> =====
1. Mencari seluruh informasi buku
2. Mencari nama penulis novel
3. Mencari harga novel (harga terendah ke harga tertinggi)
9. Keluar
=====
## Pilih menu : 3
>> Masukkan harga minimum untuk novel yang ingin dicari : -1000
[Error] Informasi harga tidak boleh berisi angka negatif. Silakan masukkan lagi.
```

```
===== << Program Informasi Buku >> =====
1. Mencari seluruh informasi buku
2. Mencari nama penulis novel
3. Mencari harga novel (harga terendah ke harga tertinggi)
9. Keluar
=====
## Pilih menu : 3
>> Masukkan harga minimum untuk novel yang ingin dicari : 15000
>> Masukkan harga maksimum untuk novel yang ingin dicari : -1000
[Error] Informasi harga tidak boleh berisi angka negatif. Silakan masukkan lagi.
```

Sample Run (Kasus 3): Nilai yang dimasukkan salah

```
===== << Program Informasi Buku >> =====
1. Mencari seluruh informasi buku
2. Mencari nama penulis novel
3. Mencari harga novel (harga terendah ke harga tertinggi)
9. Keluar
=====
## Pilih menu : 3
>> Masukkan harga minimum untuk novel yang ingin dicari : 13000
>> Masukkan harga maksimum untuk novel yang ingin dicari : 2000
Nilai maksimum tidak boleh kurang dari nilai minimum. Silakan masuk lagi.
```

Class Diagram



Book.java: Class yang berisi informasi dasar buku (disediakan).

1. Nama package adalah “book”.
2. Field dan metode getter yang harus dimiliki kelas Book telah ditentukan.
3. toString(): Mengembalikan informasi dasar buku dengan menggabungkannya menjadi sebuah string. Contoh: “[Judul] Cooking Light [Harga] 15.000”
 - a. Gunakan moneyFormat dari util untuk menampilkan simbol mata uang.

Magazine.java: Class yang menyimpan informasi department

1. Nama package adalah “book”
2. Lihat Diagram Kelas untuk mewarisi kelas Book dan menentukan field yang harus dimiliki kelas Magazine.
3. Magazine(String title, int price, String category, String description) : Menginisialisasi nilai field yang diterima sebagai parameter. Untuk judul dan harga majalah, gunakan konstruktor kelas induk.
4. Tulis metode getter untuk field tersebut.
5. toString(): Mengembalikan informasi majalah dengan menggabungkannya menjadi sebuah string. Untuk menggabungkan string judul dan harga majalah, gunakan metode toString() dari kelas induk. Contoh: “[Judul] Cooking Light [Harga] 15,000 won [Kategori] hidup, memasak [Keterangan] Masakan Amerika”

Novel.java: Class yang berisi informasi novel.

1. Nama package adalah “book”.
2. Lihat Diagram Kelas untuk mewarisi kelas Book dan menentukan field yang harus dimiliki kelas Novel.
3. Novel(String title, int price, String author) : Menginisialisasi nilai field yang diterima sebagai parameter. Untuk judul dan harga novel, gunakan konstruktor kelas induk.
4. Tulis metode getter untuk field tersebut.
5. toString(): Mengembalikan informasi novel dengan menggabungkannya menjadi sebuah string. Untuk menggabungkan string judul dan harga novel, gunakan metode toString() dari kelas induk.
Contoh: “[Judul] Pengakuan [Harga] \10,500 [Penulis] Grisham, John”

IBookBiz.java: Class yang berisi informasi novel.

1. Nama package adalah “book.biz”.
2. Tentukan metode abstrak yang harus dimiliki antarmuka IBookBiz, mengacu pada Diagram Kelas.

BookBiz.java: Kelas yang berisi logika yang diperlukan untuk sistem pencarian buku.

1. Nama package adalah “book.biz”.
2. Lihat Diagram Kelas dan tentukan field yang harus dimiliki kelas BookBiz.
 - a. Field buku dideklarasikan sebagai tipe array dari kelas Book.
 - b. Untuk setiap nilai dalam array buku, 2 objek Majalah dan 4 objek Novel dibuat dan disimpan dalam urutan tersebut, mengacu pada Sample Data.
3. printAllBooks(): Mencetak informasi semua buku.
 - a. Cetak semua informasi buku yang disimpan dalam array menggunakan loop.
 - b. Saat mencetak informasi buku, gunakan metode toString().
4. searchNovelByAuthor(String author): Mencari novel menggunakan nama penulis.
 - a. Di antara informasi buku yang disimpan di field buku, cari nama penulis di antara indeks 2 sampai 5 yang sesuai dengan novel tersebut.
 - b. Pencarian dan keluaran novel karya penulis dimasukkan sebagai parameter. Ini hanya ditampilkan jika nama penulis yang dimasukkan sebagai parameter sama persis dengan nama penulis novel.
5. searchNovelByPrice(int minPrice, int maxPrice): Mencari novel menggunakan harga.
 - a. Di antara informasi buku yang disimpan di field buku, cari harga di antara indeks 2 hingga 5 yang sesuai dengan novel.
 - b. Cari novel dengan harga antara nilai minimum dan maksimum yang dimasukkan sebagai parameter dan keluarannya. Pencarian termasuk boundary value.
6. printHeader(): Mencetak bagian atas daftar informasi buku.
 - a. Lihat Sample Run dan cetak “---Informasi Buku---” di bagian atas daftar informasi buku.
7. printTail(): Mencetak bagian bawah daftar informasi buku.
 - a. Lihat Sample Run dan cetak “-----” di bawah daftar informasi buku.

BookTest.java: Kelas untuk menguji penanganan menu dan sistem pencarian buku

1. Nama package adalah “book.test”
2. main(): Pilih menu untuk menjalankan program.
 - a. Buat objek kelas BookBiz. (Membuat objek dengan tipe referensi IBookBiz.)
 - b. Menerima input menu dari pengguna.
 - c. Ulangi terus menerus hingga menu '9' (keluar) diinput.
 - d. Jika menu '1' (Mencari seluruh informasi buku) dipilih, metode yang sesuai akan dipanggil di kelas BookBiz.
 - e. Ketika menu '2' (Mencari nama penulis novel) dipilih, metode yang sesuai akan dipanggil di kelas BookBiz.

- Jika ada parameter yang diperlukan untuk memanggil metode, lihat Sample Run untuk menerima input.
 - f. Ketika menu '3' (Mencari harga novel) dipilih, metode yang sesuai akan dipanggil di kelas BookBiz.
 - Jika ada parameter yang diperlukan untuk memanggil metode, lihat Sample Run untuk menerima input.
 - Jika informasi harga baru dimasukkan sebagai angka negatif, rujuk ke Sample Run untuk menampilkan pesan kesalahan.
 - Jika nilai minimum harga lebih besar dari nilai maksimum harga, lihat Sample Run untuk menampilkan pesan kesalahan.
 - g. Jika Anda memasukkan nomor selain menu 1, 2, 3, atau 9, lihat ke Sample Run dan keluarkan pesan kesalahan yang sesuai.
3. printMenu(): Mencetak menu.

Sample Data

<u>Judul</u>	<u>Kategori</u>	<u>Penulis</u>	Harga	<u>Catatan</u>
Cooking Light	living, cooking		15000	America Cooking
Auto Bild	science, car		16000	Germany Car
The Confession		Grisham, John	10500	
Les Miserables		Hugo, Victor	17500	
Breakthrough		Ifill, Gwen	47200	
The Racketeer		Grisham, John	38000	