

---

**Started on** Monday, 2 June 2025, 8:20 AM

---

**State** Finished

---

**Completed on** Monday, 2 June 2025, 9:50 AM

---

**Time taken** 1 hour 29 mins

---

**Grade** **40.00** out of 300.00 (**13%**)

## Question 1

Partially correct

Mark 40.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

# Inventory Management

Kalian ditugaskan untuk membuat suatu program yang bisa mengatur inventory suatu toko yang menjual handphone dan baju. Program menyimpan data dalam format katalog barang yang berbentuk (**barang**, **jumlah**) dengan barang merupakan barang yang disimpan dan jumlah adalah jumlah barang tersebut dalam inventory.

Terdapat beberapa fungsi yang harus kalian buat untuk membantu toko tersebut menjalankan bisnisnya yaitu :

- **addCatalog** : menambahkan katalog barang baru ke dalam inventory
- **removeCatalog** : menghilangkan katalog barang terakhir dalam list  
apabila inventory kosong akan mencetak "Inventory Kosong"
- **printTotalType** : mencetak total katalog barang tiap tipe barang
- **getTotalAmount** : mendapatkan total jumlah (amount) suatu barang dengan tipe tertentu dalam inventory
- **getTotalPrice** : mendapatkan total jumlah harga semua barang dalam inventory
- **printInventory** : mencetak isi inventory

untuk penjelasan mendalam tiap fungsi, harap lihat di [attachment!](#) untuk soal ini submit **InventoryManager.java**

## CONTOH PROGRAM

Berikut penjelasan untuk fungsi **printTotalType**, **getTotalAmount**, **getTotalPrice**, **printInventory** Misalnya inventory berisi :

- Kaos putih (tipe baju) seharga 50 sebanyak 10 buah
- Kaos hitam (tipe baju) seharga 10 sebanyak 5 buah
- HP2003 (tipe handphone) seharga 100 sebanyak 1 buah

**printTotalType()** akan mencetak

```
[Total Baju : 2]
[Total Handphone : 1]
```

**getTotalAmount("Baju")** akan mengembalikan

```
15
```

karena total jumlah baju adalah jumlah kaos putih + Jumlah kaos hitam =  $10 + 5 = 15$

**getTotalPrice()** akan mengembalikan

```
650
```

dari  $(50 * 10) + (10 * 5) + (100 * 1) = 650$

**printInventory()** akan mencetak

```
[Kaos putih(50) : 10]
[Kaos hitam(10) : 5]
[HP2003(100) : 1]
```

## HINT

- **add(object)** : menambahkan object ke dalam list
- **remove(n)** : menghilangkan elemen indeks n dari list
- **get(n)** : mendapatkan elemen dengan indeks n
- **size()** : memberikan jumlah elemen dalam list
- **getClass().getName()** : mendapatkan nama kelas

Java 8

 [InventoryManager.java](#)

Score: 40

Blackbox

Score: 40

Verdict: Wrong answer

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.06 sec, 28.99 MB
2	20	Accepted	0.08 sec, 30.48 MB
3	0	Wrong answer	0.10 sec, 28.83 MB
4	0	Wrong answer	0.08 sec, 27.94 MB
5	0	Wrong answer	0.06 sec, 26.22 MB

Time limit	1 s
Memory limit	64 MB

## Goodbye Olympia!

"Olympia tuh website yang bagus banget ya!"

Kata-kata di atas pasti tidak pernah terucap dari mulut satupun mahasiswa IF dan STI. Pasti salah satu dari kalian pernah mengalami error, hang, bug atau bahkan crash ketika sedang menggunakan website tersebut. Bahkan, untuk melakukan kompilasi dan grading saja Olympia membutuhkan waktu yang lama!

Tuan Bil, pembenci nomor X Olympia, memutuskan untuk membuat baru sistem Olympia namun dengan lebih cepat. Ia sudah menuliskan kode yang dapat melakukan kompilasi dan grading untuk sebuah soal. Sistem tersebut berjalan dengan lancar... sampai ada yang memasukan data aneh. Karena kebingungan, Tuan Bil bertanya ke kalian, para ahli Java, bagaimana cara menyelesaikan masalah tersebut. Untungnya, kalian memahami konsep **Exception** sehingga kalian bisa membantu me-refactor kode Tuan Bil dan menuliskan runner BukanOlympia-nya!

Tugas kalian adalah:

1. Menambahkan exception handling untuk method **getBahasaRunner()** serta **compile()** dan **grade()** untuk seluruh bahasa!
2. Membuat main file untuk menjalankan sebuah *grading* dengan contoh input dibawah.

Diberikan [kode lama Tuan Bil](#) dan submit kembali seluruh file tersebut!

## Klarifikasi Error Message

- Error `WrongExtentionException` akan mengembalikan error dalam format `Expected extention "{supposedExtention}" for {filename}`.

Contoh: `Expected extention ".java" for Main.bukanjava`

- Error `TestcaseMismatchException` akan mengembalikan error dalam format `Input testcase for {inputFilename} got output filename of {outputFilename}`. Contoh: `Input testcase for 1.in got output filename of not1.out`

- Error `BahasaNotExistsException` akan mengembalikan error dengan pesan `Bahasa doesn't exists!`

## Aturan Exception

- Method **getBahasaRunner()** harus memastikan apakah nama bahasa adalah "Jawa", "C" atau "Ular". Selain ketiganya, maka kembalikan error *BahasaNotExistsException*.
- Method **compile()** harus memastikan apakah seluruh nama file dalam *files* memiliki extention yang sesuai dengan bahasanya. Jika tidak, maka kembalikan error *WrongExtentionException*.
- Method **grade()** harus memastikan apakah *inputFilename* memiliki extention ".in". Jika tidak, maka kembalikan error *WrongExtentionException*.
- Method **grade()** harus memastikan apakah *outputFilename* memiliki extention ".out". Jika tidak, maka kembalikan error *WrongExtentionException*.
- Method **grade()** harus memastikan apakah nama file *outputFilename* dan *inputFilename* (tanpa extention) sama. Jika tidak, maka kembalikan error *TestcaseMismatchException*.

## Aturan Main

- Panggil **getBahasaRunner()**.
- Jika bahasa tidak tersedia, maka langsung keluar dari program dan cetak error message.
- Panggil metode **compile()**.
- Jika terjadi error, maka langsung keluar dari program dan cetak error message.
- Loop seluruh testcase dan panggil metode **grade()** untuk setiap testcase.
- Jika terjadi error, cetak error message dan lanjut ke testcase selanjutnya.

## Format Input

```
Nama Bahasa
Jumlah File (integer)
File-1
File-2
...
Jumlah Testcase (integer)
Input-1 Output-1
Input-2 Output-2
...
```

## Contoh Input 1

```
Jawa
1
Main.jawa
1
1.in 1.out
```

## Contoh Output 1

```
javac Main.jawa
java Main < 1.in > 1.out
```

## Contoh Input 2

```
C
2
Main.c
ADT.c
4
1.in 1.out
2.bukanin 2.out
3.in bukan3.out
4.in 4.out
```

## Contoh Output 2

```
gcc -o Main Main.c ADT.c
./Main < 1.in > 1.out
Expected extention ".in" for 2.bukanin
Input testcase for 3.in got output filename of bukan3.out
./Main < 4.in > 4.out
```

Java 8

 [NotePurry.java](#)

Score: 0

Blackbox

Score: 0

Verdict: Compile error

Evaluator: Exact

No	Score	Verdict	Description
	0	Compile error	make: *** No rule to make target 'BukanOlympia.java', needed by 'BukanOlympia.class'. Stop.

Question **3**

Incorrect

Mark 0.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

## NotePurry

Beberapa hari lalu, kalian telah melewati sebuah **tantangan epik**. Kalian bikin game nanam-nanam, mancing ikan, pdkt NPC, dan banyak hal menarik lainnya. Hasilnya? Banyak yang **keren banget**. Tapi sayangnya, tidak semua berjalan mulus. **Lihat saja di X**, timeline isinya banyak yang mengeluh tentang tubes ini.

Setelah tugas ini selesai, pasti kalian berharap soal 3 praktikum selanjutnya **ringan** dong. Oleh karena itu, kali ini kalian diminta untuk membantu Tuan Bil yang sedang **lelah meng-TA**.

Tuan Bil lagi pengen menyelesaikan sebuah side quest. Sebuah text editor yang bisa menyelesaikan perang abadi antara **Vim dan Emacs**. Padahal dia sendiri... pakainya Nano. Tuan Bil pastinya pengen text editor ini memiliki banyak fitur. Sayangnya, dia terlalu sibuk **procrastinating** dan sekarang dia butuh bantuan kalian untuk bikin bare minimum text editor yang bisa dipakai secepatnya. Untuk mencapainya, kalian cukup **melengkapi kode** yang sudah disediakan di bawah pada file bernama **NotePurry.java**.

Semangat menyelesaikan soal praktikum **terakhir** ini! ...Or, who knows, maybe this is just **a new beginning**.

## Notes

- Diperbolehkan membuat fungsi tambahan

## Attachments

Attachments: [NotePurry.java](#)

Java 8

 [NotePurry.java](#)

Score: 0

Blackbox

Score: 0

Verdict: Wrong answer

Evaluator: Exact

No	Score	Verdict	Description
1	0	Wrong answer	0.07 sec, 28.27 MB
2	0	Wrong answer	0.09 sec, 29.07 MB
3	0	Wrong answer	0.07 sec, 28.97 MB
4	0	Wrong answer	0.10 sec, 28.31 MB
5	0	Wrong answer	0.21 sec, 28.93 MB

◀ Pra-Kuis 2

Jump to...

