

Started on	Sunday, 9 March 2025, 4:46 PM
State	Finished
Completed on	Monday, 10 March 2025, 12:19 AM
Time taken	7 hours 33 mins
Grade	300.00 out of 300.00 (100%)

Time limit	1 s
Memory limit	64 MB

# Cimol Bojot Kunci Sukses Kehidupan

## Deskripsi

Nona Sherra butuh uang cepat tapi tetap halal. Melihat tumpah-ruahnya antrean Cimol Bojot Ganyang, Nona Sherra melihat kesempatan: memebuka franchise baru Cimol Bojot di dalam Kantin ITB. Bantu Nona Sherra menghitung hasil penjualan Cimol Bojotnya :D

Kelas **CimolBojot** merupakan sebuah kelas yang memodelkan Cimol Bojot yang dijual oleh Toko Cimol Sherra. Cimol bojot ini akan memiliki 3 atribut, yaitu:

- **namaMenu** yaitu nama menu/rasa dari cimol bojot
- **harga** yaitu harga dari menu cimol bojot
- **stok** yaitu stok yang tersedia

Kelas **Toko** merupakan sebuah kelas yang memodelkan toko cimol bojot yang dimiliki oleh Nona Sherra. Kelas ini akan memiliki 4 atribut, yaitu:

- **cimolBojotArray** yaitu daftar cimol bojot yang dimiliki
- **kas** yaitu jumlah uang di kasir
- **MAX\_CIMOL** yaitu jumlah maksimal jenis cimol yang dimiliki, bernilai 3 (input dipastikan 3 jenis cimol berbeda)
- **cimolCount** yaitu jumlah cimol yang dimiliki

## Batasan

1. Nama cimol bojot dipastikan tidak ada spasi
2. Cimol bojot yang dimiliki oleh sebuah toko maksimal hanya 3
3. Seluruh input angka dipastikan >= 0
4. Input tanpa <>

## Program Utama

Program utama (**Main**) akan memodelkan proses penjualan toko cimol bojot. Input:

- Baris pertama, kedua, ketiga: <namaCimol> <hargaCimol> <stokCimol>
- Baris keempat: <kasAwal>
- Baris kelima: nama cimol yang akan dijual
- Baris keenam: jumlah cimol yang akan dijual

Output:

- Jika nama cimol yang dibeli tidak dijual: **Kita gak jual itu kaka**
- Jika cimol yang ingin dibeli lebih banyak daripada stok: **Waduh stoknya gak cukup**
- Jika pembelian cimol berhasil (terdiri dari 2 baris):  
    **Berhasil menjual cimol <namaMenu> sebanyak <quantity>**  
    **Sekarang sisa: <stok>**
- Baris terakhir (output disertai newline): **Kas kasir sekarang: <kas>**

## Contoh Masukan 1

```
balado 10000 10
bbq 15000 10
keju 20000 20
50000
keju
5
```

## Contoh Keluaran 1

```
Berhasil menjual cimol keju sebanyak 5
Sekarang sisa: 15
Kas kasir sekarang: 150000
```

## Contoh Masukan 2

balado 10000 10  
bbq 15000 10  
keju 20000 20  
50000  
ayam  
5

## Contoh Keluaran 2

Kita gak jual itu kaka  
Kas kasir sekarang: 50000

Kumpulkan file **Main.java**  
Lengkapi file [CimolBojot.java](#) dan [Toko.java](#) untuk membantu menjawab soal

Java 8 ▾

 [Main.java](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	25	Accepted	0.07 sec, 28.92 MB
2	25	Accepted	0.07 sec, 28.02 MB
3	25	Accepted	0.06 sec, 28.71 MB
4	25	Accepted	0.06 sec, 27.60 MB

Time limit	1 s
Memory limit	64 MB

# Balalalala

Kelas **Kartu** merupakan sebuah kelas yang memodelkan Kartu dalam permainan Balala. Kartu ini akan memiliki 3 atribut, yaitu:

- **suit** yaitu simbol **diamond, club, heart, spade**
- **rank** yaitu nilai kartu dengan urutan **ace, 2, 3, 4, 5, 6, 7, 8, 9, 10, jack, queen, king**
- **multiplier** yaitu pengali dari nilai kartu

Nilai dari atribut-atribut **Kartu** akan diinisialisasikan dalam 2 buah konstruktor. Kontruktor pertama akan menerima multiplier sedangkan konstruktor kedua tidak akan menerima multiplier. Terdapat method-method yang harus kalian implementasi sesuai arahan comment pada method.

Kelas **Blind** merupakan sebuah kelas yang memodelkan satu stage dalam permainan Balala. Pada blind, nilai kartu akan dijumlahkan kemudian dikalikan sesuai dengan tipe joker yang dimiliki kemudian nilai dibandingkan dengan target. Kelas ini memiliki atribut:


- **kartu-n** yaitu kartu 1 sampai 5
- **targetValue** yaitu nilai yang harus dicapai untuk memenangkan blind
- **jokerType** yaitu tipe joker yang dimiliki; Joker valid adalah joker 1, 2, dan 3

Nilai dari atribut-atribut **Blind** akan diinisialisasi di konstruktor dan akan ada method-method yang harus kalian implementasi sesuai arahan comment pada method.

Lengkapi file [Kartu.java](#) dan [Blind.java](#)

Kumpulkan file Kartu.java dan Blind.java dalam satu file zip dengan penamaan bebas!

Java 8 ▾

 [2.zip](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	2.5	Accepted	0.06 sec, 28.33 MB
2	2.5	Accepted	0.06 sec, 28.30 MB
3	2.5	Accepted	0.06 sec, 26.27 MB
4	2.5	Accepted	0.06 sec, 30.80 MB
5	2.5	Accepted	0.06 sec, 27.80 MB
6	2.5	Accepted	0.06 sec, 28.27 MB
7	2.5	Accepted	0.06 sec, 28.95 MB
8	2.5	Accepted	0.06 sec, 30.41 MB
9	2.5	Accepted	0.06 sec, 28.97 MB
10	2.5	Accepted	0.06 sec, 27.97 MB

No	Score	Verdict	Description
11	2.5	Accepted	0.06 sec, 27.98 MB
12	2.5	Accepted	0.06 sec, 28.90 MB
13	2.5	Accepted	0.07 sec, 28.76 MB
14	2.5	Accepted	0.06 sec, 28.92 MB
15	2.5	Accepted	0.06 sec, 27.88 MB
16	2.5	Accepted	0.06 sec, 28.53 MB
17	2.5	Accepted	0.06 sec, 28.92 MB
18	2.5	Accepted	0.06 sec, 28.41 MB
19	2.5	Accepted	0.06 sec, 30.34 MB
20	2.5	Accepted	0.06 sec, 27.86 MB
21	2.5	Accepted	0.06 sec, 26.20 MB
22	2.5	Accepted	0.06 sec, 27.90 MB
23	2.5	Accepted	0.06 sec, 28.36 MB
24	2.5	Accepted	0.06 sec, 27.82 MB
25	2.5	Accepted	0.06 sec, 30.20 MB
26	2.5	Accepted	0.06 sec, 28.29 MB
27	2.5	Accepted	0.05 sec, 27.83 MB
28	2.5	Accepted	0.06 sec, 28.90 MB
29	2.5	Accepted	0.06 sec, 28.54 MB
30	2.5	Accepted	0.06 sec, 29.57 MB
31	2.5	Accepted	0.07 sec, 30.43 MB
32	2.5	Accepted	0.16 sec, 28.39 MB
33	2.5	Accepted	0.10 sec, 28.95 MB
34	2.5	Accepted	0.10 sec, 28.91 MB
35	2.5	Accepted	0.09 sec, 28.50 MB
36	2.5	Accepted	0.07 sec, 28.99 MB
37	2.5	Accepted	0.06 sec, 29.10 MB
38	2.5	Accepted	0.06 sec, 28.05 MB
39	2.5	Accepted	0.06 sec, 28.53 MB
40	2.5	Accepted	0.06 sec, 28.16 MB

Time limit	1 s
Memory limit	64 MB

## Main Balala

Tuan Bil membutuhkan program yang akan menghitung skor akhir dari setiap blind permainan balala dan memberi tau dirinya apakah blind tersebut akan menang atau kalah.

Buatlah program `MainBalala.java` dan gunakan kelas / file `Kartu.java` dan `Blind.java` yang sudah dibuat sebelumnya untuk mengimplementasikan program tersebut!

## Format Masukan

- Baris 1-3 akan meminta suit, rank dan multiplier untuk kartu pertama. Misalnya : diamond 12 10 adalah kartu diamond dengan rank 12 dan multiplier 10.
- Ulang aksi tersebut untuk kartu 2-5, dengan :
- baris 4-6 untuk kartu 2
- baris 7-9 untuk kartu 3
- baris 10-12 untuk kartu 4
- baris 13-15 untuk kartu 5
- Setelah semua kartu dibuat, program akan meminta nilai joker type yang digunakan blind. Nilai joker merupakan nilai integer (lihat ketentuan di `Blind.java`)
- Kemudian, program akan meminta nilai target yang dimiliki blind tersebut. (lihat ketentuan di `Blind.java`)

## Format Keluaran

- Baris pertama hingga kelima akan mengeluarkan informasi mengenai kartu pertama dalam format : `Kartu N punya suit S dengan rank R dan multiplier M` dengan :
- N = urutan kartu (pertama, kedua, ketiga, keempat, kelima)
- S = suit kartu
- R = rank kartu
- M = multiplier kartu
- Skor akhir yang dituliskan dalam format : `Skor akhir : 13800`
- Baris terakhir dapat berupa `MENANG!` atau `KALAH!`, sesuai dengan kondisi kemenangan blind tersebut dengan newline di akhir

## Contoh Masukan 1

```
diamond
king
100
club
10
10
heart
jack
10
spade
8
10
diamond
10
10
1
100
```

# Contoh Keluaran 1

Kartu pertama punya suit diamond dengan rank 10 dan multiplier 100  
Kartu kedua punya suit club dengan rank 10 dan multiplier 10  
Kartu ketiga punya suit heart dengan rank 10 dan multiplier 10  
Kartu keempat punya suit spade dengan rank 8 dan multiplier 10  
Kartu kelima punya suit diamond dengan rank 10 dan multiplier 10  
Skor akhir : 13800  
MENANG!

## Penjelasan

diamond  
king  
100  
club  
10  
10  
heart  
jack  
10  
spade  
8  
10  
diamond  
10  
10


Ini merupakan informasi mengenai kartu yang akan dimainkan

1  
100

Ini merupakan informasi tipe joker (1) dan target blind (100)

Submit file **MainBalala.java**

Java 8 ▾

 [MainBalala.java](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.06 sec, 28.77 MB
2	10	Accepted	0.06 sec, 28.27 MB
3	10	Accepted	0.06 sec, 27.90 MB
4	10	Accepted	0.06 sec, 28.42 MB
5	10	Accepted	0.06 sec, 28.70 MB
6	10	Accepted	0.06 sec, 29.10 MB
7	10	Accepted	0.06 sec, 27.84 MB
8	10	Accepted	0.06 sec, 27.75 MB
9	10	Accepted	0.06 sec, 28.11 MB
10	10	Accepted	0.06 sec, 28.37 MB

