Laporan TP2

Nama : Oki Priyadi

NIM : 1806280552

Pada TP tiga kali ini saya membuat tree untuk menampung struktur krusty crab. Kemudian semua tingkatan dibungkus dalam satu buah Hashmap agar memudahkan mengakses barisan tersebut. Hashmap key adalah nama dari masing-masing tingkatan, sehingga pada tingakatan samudra, hashmapkey adalah nama samudra, pada tingkatan region hasmap key adalah nama region dan seterusnya.

Untuk menu karena dibutuhkan dua tipe menu yang itu menu sesuai dengan urutan menu itu diinput dan yang kedua menu dengan urutan sesuai abjad menggunakan **orderlist**, maka dibuat dua objek berisi kedua urutan menu diatas.

Untuk setiap perintah saya buat saya klasifikasikan berdasarkankan perintahnya.

1. **Perintah Tambah**

Perintah tambah cukup sederhana, kita hanya butuh untuk mengambil stok yang lama kemudian kita tambahkan dengan tambahan stok yang diinputkan. Hasilnya kembali dimasukan ke dalam objek Cabang. Kemudian update jumlah stok pada region dan samudra tempat cabang itu berada.

1. **Perintah Kurang**

Perintah kurang juga cukup sederhana, kita hanya butuh untuk mengambil stok yang lama kemudian kita kurangkan dengan stok terjual yang dari hasil input. Hasilnya kembali dimasukan kedalam objek Cabang. Kemudan update jumlah stok pada region dan samudra tempat cabang itu berada.

Selain itu kita juga harus mengupdate data terjual dengan cara mengambil data terjual yang lama keduian tambahkan dengan stok terjual yang sama dengan hasil inputan diatas. Hasilnya kembali di masukan ke dalam objek Cabang. Kemudian update jumlah data terjual pada region dan samudra tempat cabang itu berada.

1. **Perintah Transfer.**

Pada perintah ini pertama-tama ambil stok lama dari cabang asal kemudian dikurangi dengan stok yang ditransfer hasil dari inputan. Hasilnya kembali dimasukan kedalam objek Cabang. Kemudan update jumlah stok pada region dan samudra tempat cabang itu berada.

Selain itu kita juga harus mengambil stok lama dari cabang tujuan namun kali ini ditambahkan dengan stok yang ditransfer hasil dari inputan. Hasilnya kembali dimasukan kedalam objek Cabang. Kemudan update jumlah stok pada region dan samudra tempat cabang itu berada.

1. **Jenis Terlaris**

Pada perintah terlaris, pertama – tama saya definisikan nilai defult terlebih dahulu

String namaMenu = "ZZZ";

int jumlahTerjualMaks = -1;

harapannya namaMenu akan terganti dengan menu yang terlaris dengan cara: dapatkankan terlebih dahuku jumlah terjual dari masing-masing menu pada cabang yang diinginkan, kemudian bandingkan antara jumlah terjual dengan variabel jumlahTerjualMaks diatas. Jika jumlah terjual lebih besar maka ubah jumlahTerjualMkas dengan nilai jumlah terjual dan ubah namaMenu dengan nama menu pada jumlah terjual tersebut. Dengan demikian dipdatkan nama menu menu dan juga jumlah terjual maksimal pada cabang tertentu

1. **Jenis Teruntung**

pertama – tama saya definisikan nilai defult terlebih dahulu

String namaMenu = "ZZZ";

int jumlahKeuntunganMaks = -1;

Untuk setiap cabang pada region yang diinginkan lakukan perulangan berikut:

1. Lakukan perulangan pada setiap menu dan totalkan seluruh keuntungan menu sehingga didapati jumlah Keutungan percabang .
2. kemudian bandingkan antara jumlah keuntungan percabang dengan variabel jumlahKeuntunganMaks diatas. Jika jumlah keuntungan percabang lebih besar maka ubah jumlahKeutunganMaks dengan nilai jumlah keuntungan percabang dan ubah namaMenu dengan nama menu pada jumlah terjual tersebut. Jika hasilnya sama maka cek apakah nama cabang sudah terurut secara alfabetik.

Dengan demikian dipdatkan nama cabang dan juga jumlah keuntungan maksimal pada region tertentu

1. **Persentase Penjualan**

Untuk mendapatkan persentase penjualan rasanya cukup mudah, namun ternyata pada test case di scele banyak yang salah, dugaan saya karna kekurang tepatan dua angka dibelakang koma. Namun karena saya tidak berhasil meresolve kesalahan tersebut maka saya tuliskan apa adanya cara yang saya pakai. Caranya adalah :

1. Print nama samudra
2. Untuk setiap menu lakukan perulangan berikut :
   1. Definisikan variabel terjual dengan tipe float dan inisiasi nilainya dengan jumlah keseluruhan menu tersebut dijual pada samudra yang ditetapkan
   2. Definisikan variabel stok dengan tipe float dan inisiasi nilainya dengan jumlah keseluruhan stok menu pada samudra yang ditetapkan
   3. Definisikan variabel hasil dengan tipe float dan inisiasi nilainya dengan (terjual/(terjual+stok))\*100;
   4. Format variabel hasil sehingga menghasilkan nilai dengan precisisi dua angka dibelakang koma
   5. Print hasilnya dengan format :

out.println(namaMenu + " " + formattedString +"%");

1. **Untung Minimal**
2. Pertama-tama definisikan dahulu totalJumlahCabangUntung = 0.
3. Untuk setiap cabang lakukan perulangan untuk mendapakan jumlahTotalKeutunganCabang.
4. Jika jumlahTotalKeutunganCabang lebih besar atau samadengan untungMinimal yang ditetapkan maka tambahkan satu pada totalJumlahCabangUntung
5. Dengan demikian didapat berapa jumlah cabang yang minimal mempunya untung yang ditetapkan