

Você foi contratada(o) para implementar o sistema de um **restaurante**. O sistema deve cadastrar refeições, importar dados de um arquivo e fornecer várias formas de pagamento.



a) Escreva o encapsulamento para a **refeição do restaurante**. Cada refeição possui um **nome**, um **preço por pessoa** e uma **porção base**. Por exemplo, a refeição Carne de sol, custa **R\$15.00 por pessoa** e alimenta, idealmente, **3 pessoas**. Porém, se um grupo de pessoas maior (exemplo 5 pessoas) chegar ao restaurante eles podem fazer o pedido, e o preço é automaticamente ajustado (detalhes abaixo). **Não permita** que uma refeição possua porção base negativa ou nula, lançando o erro com a mensagem: "Porcao base deve ser maior que zero.". Complete também a classe Restaurante para que este **armazene** um uma lista de refeições.

b) Escreva o código para **ler um arquivo csv** (comma-separated value). Um arquivo csv possui Strings separadas por vírgula que se repetem em cada linha. Cada linha contém a informação: `<nome>,<precoPorPessoa>,<porcaoBase>`

```
Carne de Sol,15.0,3
Torta de Frango,5.0,1
Lasanha de voinha,10.0,6
```

Para a verificação da leitura, escreva no restaurante a lógica para o método `buscaRefeicao(String nome)`, que retorna a refeição com o nome especificado. **Se a refeição não existir** no Restaurante, lance um erro com a mensagem: "A refeicao <nome> nao esta no cardapio."

c) O restaurante fornece **dois tipos distintos de refeição**: Refeição **Executiva** e Refeição **Mundiça**. Cada tipo de refeição possui um comportamento específico com relação à descontos e a forma como é impressa no cardápio. O que determina o tipo da refeição é a **porção base**. Uma refeição **Executiva** possui uma **porção base de até 5 pessoas**, enquanto que a refeição **Mundiça** é aquela cuja **porção base é maior que 5 pessoas** (certamente a refeição Mundiça *acompanha refrigerante de 2 litros!!*).

Escreva o método `redefinePorcaoBase(String nome, int novaPorcao)` que **muda a porção base da refeição** com nome especificado (se ela existir). Se a refeição não existir, lance um erro com a mensagem: "Nao foi possível redefinir porcao. A refeicao <nome> não esta no cardapio". Trate também os casos em que a nova **porção é nula ou negativa**, lançando o erro com a mensagem: "Nao foi possível redefinir porcao. Porcao base deve ser maior que zero."

d) Cada tipo de refeição oferece uma **redução no preço** de acordo com a **quantidade de pessoas que fazem o pedido**. Escreva o método `fazPedido(String nome, int qtdPessoas)` que retorna o preço da refeição de nome especificado para a quantidade de pessoas. Note que **o resultado depende do tipo da refeição**.

- **Executivo:** Acima de 3 pessoas, cada pessoa adicional recebe um desconto de 25% no preço individual (ou seja, $0.75 * \text{precoBase}$). Em resumo:

```
Se: qtdPessoas > 3: (precoBase * 3) + (precoBase * adicional * 0.75)
Senão: precoBase * qtdPessoas
```

- **Mundiça:** Para servir mais pessoas todas as refeições mundiça possuem 10% de desconto (ou seja, $0.9 * \text{precoBase}$) para cada pessoa que fez o pedido. Em resumo:

```
total = qtdPessoas * 0.9 * precoBase
```

e) Além do preço, cada tipo de refeição é impresso de forma distinta ao consultar o cardápio. Implemente o `toString()` da Refeição para exibir as seguinte mensagens:

Executiva:

```
<nome> (Executivo): para <porcaoBase> pessoa(s). Por pessoa:
R$<precoBase>
```

Mundiça:

```
<nome> (Mundica - Acompanha refrigerante de 2L): para <porcaoBase>
pessoa(s). Por pessoa: R$<precoBase>
```

f) O restaurante deve calcular a estimativa para o **contrato de buffet**. A chamada `gastoComBuffet(int qtdPessoas)` retorna o gasto total para servir **todas as refeições do restaurante** para a **quantidade de pessoas especificada**. Isso implica que cada refeição deve calcular **automaticamente** o seu preço e o gasto com o buffet será a soma total gasta (já considerando o tipo de cada refeição).

Extra (1,0): Mais testes, Javadoc, ordenar cardápio por nome, indicar uso de composição, polimorfismo, tipos de fluxos, forwarding e etc. como comentário no código.

Link para o Arquivo de Teste: <https://goo.gl/QRD1TW>

Link para o Restaurante: <https://goo.gl/yALVBx>

Link para o arquivo CSV: <https://goo.gl/obG9d1>

Instruções para a entrega:

A entrega será para o e-mail: juiz.lp2@gmail.com

Nomeie seu projeto de acordo com o formato: **Nome_Sobrenome_Matricula**

Envie um zip do seu projeto. Cuidado para empacotar a pasta correta. Confira.

Formato: MATRICULA_NOME_SOBRENOME_TURMA_3_PROVA_3.zip

No assunto do e-mail coloque: **[Prova 3 - 2015.1] Nome Completo**