UNIVERSIDADE REGIONAL DE BLUMENAU - FURB
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
DEPARTAMENTO DE SISTEMAS E COMPUTAÇÃO
DISCIPLINA: Programação I

Prof. Aurélio Hoppe

LISTA DE EXERCÍCIO – ARRAYLIST e HASHMAP

1. Crie uma classe chamada TesteString e dentro da mesma faça o seguinte:

   a) Dentro do main, crie 10 variáveis do tipo String com nomes de times de futebol;
   b) Armazene estas strings em um ArrayList;
   c) Remova o elemento que está na SEXTA (6) posição do ArrayList;
   d) Utilize o método get() para obter cada um dos elementos do ArrayList, imprimindo-os na tela.

2. Implementar um programa para ler o cadastro de um laboratório de informática. No laboratório existem N computadores. Cada computador possui as seguintes informações (diagrama de classe):

Computador
número serial capacidade do HD quantidade de memória

Para tanto, faz-se necessário o desenvolvimento dos seguintes métodos:
   a) Construtor para inicializar os atributos da classe;
   b) Métodos acessores (*get*) e modificadores (*set*) dos atributos da classe;
   c) Um método adiciona computador, que recebe a lista (ArrayList) de computadores e um objeto tipo computador como parâmetro;
   d) Um método imprime, que recebe a lista (ArrayList) de computadores e mostre o total de memória e os dados de todos os computadores

3. Crie uma classe chamada Pessoa contendo os seguintes atributos: Nome e CPF, ambos do tipo String. Para testar esta classe, crie uma classe chamada TestePessoa. Dentro da classe TestePessoa, faça o seguinte:

   a) Crie vários objetos da classe Pessoa (Nome e CPF)
   b) Crie um objeto tipo HashMap que armazenará objetos do tipo Pessoa
   c) Dentro do main (TestePessoa) crie um método chamado inserir que recebe como parâmetro um CPF, um objeto da classe Pessoa e o objeto HashMap. Dentro deste método insira o par CPF-Pessoa no HashMap
   d) Dentro do main (TestePessoa) faça um método que imprima na tela o CPF e Nome das pessoas armazenadas no HashMap.

4. O objetivo desta questão é implementar um controle de acesso a sites. Para isto crie duas classes: Site.java, e DicionarioDeIP.java (main).

Na classe Site faça o seguinte:
   a) Crie três atributos privados: nome do site (String), endereço IP do site (String), status do site (acesso livre ou bloqueado) (boolean);
   b) Construtor para inicializar os atributos da classe;
   c) Crie os métodos getters e setters para cada um dos atributos;
   d) Sobrescreva o método toString() da classe Object para gerar uma String com os três atributos da classe.

Na classe DicionarioDeIP (main) faça o seguinte:
   a) Crie vários objetos da classe Site (Nome, endereço IP e status de acesso) ;
   b) Crie um objeto tipo HashMap que armazenará objetos do tipo Site, onde a chave corresponde ao nome do site;
   c) Dentro do main (DicionarioDeIP) crie um método chamado *inserirSite* que recebe como parâmetro um objeto da classe Site e o objeto HashMap. Dentro deste método insira o par [Nome do site] – [Site] no HashMap
   d) Dentro do main (DicionarioDeIP) crie um método chamado *acessarSite* que recebe como parâmetro uma String com o [Nome do site] e o objeto HashMap. Se o status do site estiver definido como acesso livre, retorne o objeto Site, mostrando as informações [nome e endereço IP do site] ao usuário. Agora se o status do site estiver definido como bloqueado, retorne uma mensagem de erro, informando ao usuário sobre a indisponibilidade de uso do site (site bloqueado);
   e) Dentro do main (DicionarioDeIP) faça um método que imprima na tela o Nome, endereço IP e status do site armazenadas no HashMap.

Exemplo de execução:

ITEM	Exemplo de entrada	Saída para o exemplo de entrada
a)	Site s = new Site(“www.inf.furb.br”, “201.54.201.5”, true); Site s1 = new Site(“www.orkut.com”, “64.233.163.86”, false);	
c)	inserirSite(s , mapaSite); inserirSite(s1, mapaSite);	
d)	System.out.println(acessarSite(“www.inf.furb.br”, mapaSite));	Site: www.inf.furb.br IP: 201.54.201.5
	System.out.println(acessarSite(“www.orkut.com”, mapaSite));	Site bloqueado
e)	System.out.println(imprimirSite(mapaSite));	Site: www.inf.furb.br IP: 201.54.201.5 Acesso: Permitido Site: www.orkut.com IP: 64.233.163.86 Acesso: Bloqueado

5. Para efetuar o recolhimento do Imposto de Renda a Receita Federal tem o NOME, CPF e RENDA ANUAL de cada contribuinte, durante o ano. Exemplo: Nome: João da Silva CPF: 123.456.789-00 RendaAnual: R$10.000

Nível de Renda Anual	Alíquota
0 a 4.000	0%
4.001 a 9.000	5,8%
9.001 a 25.000	15%
25.001 a 35.000	27,5%
Acima de 35.000	30%

Para o cálculo do imposto a pagar de cada contribuinte, considere o seguinte:

Sendo assim, deve-se calcular o imposto a pagar do seguinte modo:
Imposto a pagar = Renda Anual * Alíquota

Faça um programa Java orientado a objetos que leia as informações a serem digitadas pela Receita, até que o usuário escolha a opção “sair”. Ao final do programa será possível:

a) Digitar o CPF de algum contribuinte e ver seus dados e o imposto a pagar;
b) Saber os dados e o imposto a pagar do contribuinte que tem o maior imposto a pagar.

6. O objetivo deste exercício é implementar a relação entre uma pessoa e seus endereços. Para isto crie três classes: Pessoa.java, Endereco.java e TestePessoa.java.

Na classe Pessoa faça o seguinte:

a) Crie dois atributos privados: Um do tipo String para o nome da pessoa e outro do tipo ArrayList para armazenar os vários endereços da Pessoa;
b) Crie todos os getters e setters para estes atributos;
c) Crie um construtor alternativo para iniciar o atributo nome;
d) Crie um método chamado "imprimirEnderecos" e dentro do mesmo utilize o método get() do ArrayList para obter cada um dos endereços da Pessoa e imprima os mesmos na tela. Utilize toString().

Na classe Endereco faça o seguinte:

a) Crie três atributos privados: um do tipo String para o logradouro e outro também do tipo String para o complemento e um do tipo int para o número;
b) Crie os métodos getters e setters para cada um dos atributos;
c) Sobrescreva o método toString() da classe Object para gerar uma String com os três atributos da classe.

Na classe TestePessoa faça o seguinte:

a) Dentro do main crie um objeto a partir da classe Pessoa;
b) Crie vários objetos a partir da classe Endereco, armazenando-os no ArrayList da classe Pessoa que foi criado anteriormente;
c) Chame o método imprimirEnderecos da classe Pessoa para mostrar o nome da pessoa e todos os seus possíveis endereços de contato.

7. O objetivo deste exercício é implementar o controle de caixa/estoque de uma livraria. Para isto crie três classes: Livro.java, EstoqueLivraria.java e TesteLivraria.java.

Na classe Livro faça o seguinte:

a) Crie seis atributos privados: titulo, autor e ISBN (String), estoque (int) e pCompra e pVenda (double);
b) Crie um construtor para iniciar os atributos;
c) Crie todos os getters e setters para estes atributos;
d) Sobrescreva o método toString() da classe Object para gerar uma String com os seis atributos da classe.

Na classe EstoqueLivraria faça o seguinte:

a) Crie dois atributos privados: saldo do caixa (double) e estoque (HashMap);
b) Crie um construtor para iniciar os atributos;
c) Crie os métodos getters e setters para cada um dos atributos;
d) Crie um método para realizar a aquisição/compra de um novo livro (compraDeLivros). A compra do livro deve estar condicionada ao saldo do caixa (saldo do caixa deve ser maior ou igual ao valor de compra do livro). Ao efetivar a compra, o programa deve incrementar a quantidade em estoque do livro;
e) Crie um método para realizar a venda de um livro (vendaDeLivros). A venda está condicionada a existência do livro no acervo da livraria e a disponibilidade do livro em estoque. Ao efetivar a venda, o programa deve decrementar a quantidade em estoque do livro;
f) Crie um método para imprimir os livros existentes no acervo da livraria (imprimeEstoque).

Na classe TesteLivraria faça o seguinte:

a) Dentro do main crie um objeto a partir da classe EstoqueLivraria;
b) Crie vários objetos a partir da classe Livro, incluindo-os no estoque da livraria (compraDeLivros);
c) Efetue a vendas de alguns livros (vendaDeLivros);
d) Chame o método imprimeEstoque para mostrar os livros existentes no acervo da livraria.