

# Lista de Exercícios 2

## Prof.: Caio César de Melo e Silva

Fornecido o contexto abaixo, apresente soluções para as questões que seguem.

Imagine que você é desenvolvedor da Apple, e está participando do desenvolvimento do projeto de um aplicativo que virá nas próximas versões do sistema Mac OS. Esse aplicativo tem como principal função cadastrar e sincronizar os contatos que estão armazenados em iPhone's, iPad's e iMac's (iMac é o PC da apple). Ou seja, ao cadastrar um novo contato no *smartphone*, *tablet* ou PC, o sincronizador automaticamente se encarrega de passar as informações para os outros dispositivos, mantendo todos com os mesmos contatos.

Dado esse contexto, crie um **projeto** chamado **SincronizadorApple**. Insira nesse projeto os **pacotes de modelo, controle e visão**. E apresente soluções para as questões abaixo.

- a) Declare, pelo menos 3, possíveis classes que poderiam estar nesse programa. Declare também os atributos das classes. (Dica: Uma das classes deve se chamar **Contato**)
- b) Para as classes criadas no item **a)**, implemente os métodos para recuperar e informar valor dos atributos.
- c) Para as classes criadas no item **a)**, implemente os métodos construtores das classes.
- d) Crie uma classe chamada Main, que possua o método “**public static void** main(String[] args) {}”. Nesse método, construa um novo objeto para cada uma das classes criadas no item **a)**, e apresente os valores dos atributos de cada objeto utilizando uma JOptionPane.
- e) Implemente o método que tem como assinatura “**public void** armazenaContato(Contato c) {}”. O objetivo desse método é receber um contato e inseri-lo na memória (entenda a memória como um array de contatos). (Dica: esse método deve estar em uma classe de controle)
- f) Implemente um método que imprima todos os nomes e telefones de contatos que estão armazenados em memória.
- g) Implemente um método que retorna a quantidade dos contatos que já foram cadastrados.
- h) Implemente um método de assinatura “**public void** removeContato(String nome) {}”, que recebe o nome do contato como parâmetro e exclui o mesmo da memória (remove do array).
- i) Implemente um método que ordene os contatos que estão na memória em ordem alfabética, e apresente os mesmos na tela.

**Desafio)** Crie um mini-programa com as classes de interface. O programa deve conter uma tela com campos de texto (representando atributos da classe “Contato”) e um botão (“Confirma”). Quando o botão for clicado os valores que estão nos campos de texto, devem ser apresentados em uma nova janela.