

Exercício (1/3)

- Baixe o arquivo `solucao.zip` e extraia no seu projeto
http://www.cin.ufpe.br/~scbs/solucaoExercicios/cin_2010_06_14/
- Crie a exceção `br.ufpe.cin.dados.RepositorioException`
- Adicione a mesma nos métodos da interface `RepositorioContas`
- Faça a classe `ContaAbstrata` implementar `java.io.Serializable`
- Crie uma classe que implementa a interface `RepositorioContas` em um arquivo binário

DICAS:

- Use a classe `RepositorioContasArray` como uma classe auxiliar
- No construtor do repositório leia os objetos do arquivo binário e insira os objetos no repositório auxiliar (como uma cache)
- Utilize a classe `File` para criar o arquivo, caso esse não exista
- Utilize `FileInputStream` e `ObjectInputStream` para ler objetos
- Utilize `FileOutputStream` e `ObjectOutputStream` para escrever objetos
- Sempre que o array mudar grave os objetos do array
- Teste o projeto usando o repositório com o `FrameBanco`

Exercício (2/3)

- **Crie um repositório baseado em arquivo texto**
 - O código será muito similar ao repositório baseado em Array. Portanto, copie ele e passe a modificá-lo
 - O construtor do repositório deverá usar um método chamado `LerArquivoTxt` para ler o conteúdo de um arquivo texto e copiá-lo para um array
 - cuidado! o arquivo pode estar vazio!
 - Utilize a classe `File` para criar o arquivo, caso esse não exista e para testar se o arquivo está vazio
 - Utilize `Scanner` para implementar o método `LerArquivoTxt`
 - Implemente um método chamado `GravarArquivoTxt` para sobrescrever o conteúdo do arquivo com o conteúdo do array
 - Utilize `FileWriter` e `BufferedWriter` para implementar o método `EscreverArquivoTxt`
 - Ao gravar um objeto `ContaAbstrata`, lembre-se de incluir no arquivo texto o tipo da conta (do contrário, essa informação será perdida; ela será útil para a leitura)
 - Esse método deve ser chamado sempre que uma conta for removida ou quando houver uma atualização em uma conta - também deve ser chamado antes de terminar o programa, para que as operações feitas sobre o array possa ter efeito

Exercício (3/3)

- Note que para “montar” cada uma das contas a serem armazenadas no array de trabalho, você deverá ter métodos `setSaldo` e `setBonus`
- Para evitar criar esses métodos, crie um construtor para inicializar o objeto com esses valores lidos do arquivo texto
- Isso não foi necessário para o arquivo binário, pois ele já monta (*deserialize*) o objeto para você
- Teste a aplicação com esse novo repositório