Arquivos

Leitura e Escrita de Fluxos de Dados

Prof. Mateus M. Luna[1]

Prof. André L. Moura[2] **Prof. Dirson S. de Campos**[3]

1. mateus_m_luna@ufg.br
2. andre_moura@ufg.br
3. dirson_campos@ufg.br

Material elaborado em parceria com os professores **Nádia F. F. da Silva, Juliana P. Félix, Guilherme S. Marques** e **Reinaldo de S. Júnior**.

2022/2

INF
INSTITUTO DE INFORMÁTICA

UFG
UNIVERSIDADE FEDERAL DE GOIÁS

Instruções

Resolva as questões que não envolvem código em um único arquivo chamado **exercicio.txt**. Lembre-se de identificá-las apropriadamente para facilitar a correção, colocando números e quebra de linha para cada uma. Também no conteúdo deste arquivo, coloque seu nome e número de matrícula.

Para as questões que envolvem códigos, deve ser enviado:

- O pacote completo criado pelo Eclipse OU;
- Uma pasta com as classes usadas.

Se houver mais de uma questão de código, lembre-se de separar em pastas ou identificar no nome dos arquivos a qual se refere. Crie o hábito de identificar o código, bem como seu nome e número de matrícula nos comentários antes das classes.

Junte todos os arquivos finais em um único **.zip,** de preferência com seu nome para submeter no SIGAA.

Exercício 1

Preencha as lacunas em cada seguimento, de acordo com o que aprendemos:

1. Para termos em nosso código uma referência a uma pasta que existe no nosso Sistema Operacional criamos um objeto da classe _____.
2. O método _____ serve para se criar uma nova pasta.
3. Ao chamarmos o construtor da classe `FileInputStream`, passando o endereço ou nome do arquivo que queremos acessar, somos obrigados a tratar uma exceção do tipo _____.
4. Uma das diferenças entre se usar um `InputStream` e um `FileReader` para se ler dados de arquivos é que o primeiro trabalha com a leitura em _____ enquanto segundo com a leitura em caracteres.
5. Para se escrever dados em arquivos na forma de `Strings`, podemos usar o método _____ da classe `BufferedWriter`, seguido de uma chamada do método `flush()`.

Exercício 2

Esta atividade terá como código inicial o que fizemos na atividade de Coleções (aquele com as classes `Baralho` e `Cartas`). Você pode usar seu código enviado caso tenha conseguido ou mesmo usar o arquivo de demonstração que foi colocado no Sigaa.

Sua missão é implementar a lógica de carregamento de dados das cartas no método `carregar()` da classe `Baralho`. A diferença é que desta vez as informações serão provenientes de arquivos **.csv**. No mesmo tópico desta atividade, estão quatro arquivos que podem ser usados como exemplo. Os detalhes do funcionamento da função estão no slide seguinte.

Lembre-se de colocar os arquivos **.csv** na mesma pasta do seu projeto do Eclipse (ou outra IDE). Eles devem estar no mesmo nível da pasta `src` e `bin`. Se você precisar de ajuda com o processo de leitura, consulte o *exemplo3* do código de Arquivos que foi enviado no Sigaa.

Exercício 2

O método `carregar()` deverá, baseando-se no `tema` definido para o `Baralho`:

- Definir qual arquivo será carregado (note que o `tema` pode ter acentos e letra maiúscula, como `"Aviões"`, enquanto o arquivo é apenas `"avioes.csv"` - não precisa fazer esta conversão de forma *"automática"*, um `switch/case` basta).
- Carregar o arquivo, usando um `FileInputStream`.
- Ler o arquivo, **linha por linha**, usando um `BufferedReader`, ignorando-se a primeira linha que contém o "cabeçalho" do **.csv**.
- Ao ler cada linha, deve-se obter as informações de *`nome`* e *`código`* da carta, para então passá-las para o construtor de `Carta`. Note que no **.csv**, estas informações estão separadas por vírgula, portanto será necessário "quebrar" esta `String` da linha em partes. Isto pode ser feito usando o método `split()` da `String`, que espera um parâmetro separador, como `","`.
- Trate todas as exceções necessárias dentro do próprio método e lembre-se de fechar os fluxos e leitores no bloco `finally`.

Arquivos

Leitura e Escrita de Fluxos de Dados

Prof. Mateus M. Luna[1]

Prof. André L. Moura[2] **Prof. Dirson S. de Campos**[3]

1. mateus_m_luna@ufg.br
2. andre_moura@ufg.br
3. dirson_campos@ufg.br

Material elaborado em parceria com os professores **Nádia F. F. da Silva, Juliana P. Félix, Guilherme S. Marques** e **Reinaldo de S. Júnior**.

2022/2

INF
INSTITUTO DE INFORMÁTICA

UFG
UNIVERSIDADE FEDERAL DE GOIÁS