

Avaliação Parcial 02

ATENÇÃO! Para esta avaliação é preciso que você:

- Separe todos os códigos e textos que vai utilizar na prova. Você pode utilizar todo e qualquer código e texto, contanto que esteja no seu computador. Não acesse a internet para resolver a prova. Quer encontrar alguma função do Processing que esqueceu o nome? Não se preocupe, ele já vem com a documentação, acesse Help -> Reference. Precisa acessar a documentação do Java? Verifique em <https://docs.oracle.com/en/java/javase/15/docs/api/index.html> apenas.
- Separe 4 horas para a resolução da prova. Após estas 4 horas, compile um arquivo com as respostas e as envie no Solar. Envie o código desenvolvido, NÃO VALE enviar link para projeto no Repl.it, por exemplo.
- Seja **honesto** consigo mesmo: estamos em um momento de pandemia e sua prova está sendo feita de forma remota. Conseguir fazer a prova seguindo as regras e obter uma boa nota faz parte do processo, indica que você conseguiu capturar os conceitos discutidos em sala. Não conseguir fazer a prova seguindo as regras e obter uma nota não tão boa também faz parte do processo, indica que talvez você tenha que aprofundar os estudos. Se utilizar de outro meio para conseguir uma boa nota é um auto-engano que vai reverberar mais pra frente na sua graduação, agora pode parecer interessante mas depois você vai sentir falta do conteúdo não estudado.

Questão 01 (1 ponto) - Diferencie um array comum de um ArrayList. Em que cenário é mais adequada a utilização de um ou de outro? Construa um método estático que concatena dois ArrayList que armazenam Strings passados como parâmetro.

Questão 02 (4 pontos) - Tratamento de exceção é uma tarefa bastante importante no desenvolvimento de software. Você foi contratado pela SimplesMente Dados para construir um substituto ao Gmail. Sua tarefa é construir uma classe Email contendo um método que recebe como parâmetro uma String e verifica se a String recebida é um email válido ou não. Como é a primeira tarefa na companhia, você vai desenvolver apenas uma versão simplificada deste método: ele deve verificar se a String contém um caractere '@' e se contém um ".com" ou ".com.br". Caso a String possua essas características, deve ser apresentado no Console uma mensagem "Email válido". Caso a String não possua essas características, deve ser lançada uma exceção "Email inválido". Construa um programa que apresenta os diferentes resultados da execução deste método.

Questão 03 (1 ponto) - Construa uma classe Pessoa cujo estado é definido pelos atributos nome e idade e cujo comportamento é definido pelos seus getters() e setters(). Nesta classe, construa também um método estático maisNovo() que recebe como parâmetro 2

objetos do tipo Pessoa e retorna aquele com o menor valor de idade. ATENÇÃO: seu método deve retornar uma Pessoa, qualquer retorno diferente disso será desconsiderado (um inteiro, uma String, por exemplo).

Questão 04 (4 pontos) - Utilizando a classe construída na questão anterior, desenvolva um projeto Java com uma GUI composta de dois JTextField e um botão. O usuário vai digitar o nome da pessoa em um dos *text fields* e a idade em outro. A partir desses dados e quando ele clicar o botão, você vai instanciar uma Pessoa e armazená-la em um ArrayList. Caso o usuário não digite nada e clique o botão, você vai apresentar os dados da Pessoa mais velha e os dados da Pessoa mais nova presentes no ArrayList.