

Programa de Bolsas AWS Digital Commerce –

Turma Java Spring Boot

Atividade 2 – Exercícios Java

Durante as semanas 2 e 3 dos Cursos de Java, você aprendeu como desenvolver utilizando a tecnologia Java, e para aplicar ainda mais o conhecimento que é repassado durante os vídeos estamos lançando uma série de exercícios para que você possa aplicar ainda mais os conhecimentos adquiridos.

- Foram selecionados 30 exercícios para cada Squad (é possível visualizar os exercícios abaixo);
- Cabe à Squad decidir de que forma irão resolver os exercícios, seja individual ou em grupo.
- Cada Squad deverá criar um repositório no Github para que os exercícios sejam disponibilizados, dessa forma todos poderão contribuir com as resoluções.
- É importante que todos da Squad saibam como é realizada a resolução dos exercícios da sua Squad.
- O prazo para resolução dos exercícios é de 28/12/2023 até 02/01/2024.
- No dia 03/01/2024 será a apresentação de alguns exercícios, nesse dia, sortearmos um exercício para cada um e será necessário explicar o que foi necessário para a resolução. Portanto, todos tem que saber como resolver todos os exercícios da sua Squad.
- A apresentação será gravada.
- A participação na resolução das atividades (commits) e domínio de explicar o código na apresentação serão avaliados.
- Necessário disponibilizar o link do repositório para os Supervisores.

Squad 1:

1. Crie um programa que receba dois números e exiba a soma, subtração, multiplicação e divisão entre eles.
2. Desenvolva um programa que determine se um número é par ou ímpar.
3. Escreva um programa que leia o nome completo de uma pessoa e exiba o número de caracteres no nome.
4. Elabore um programa que inverta uma string fornecida pelo usuário.
5. Implemente um programa que conte o número de vogais em uma string informada.
6. Crie um programa que verifique se uma palavra é um palíndromo.
7. Desenvolva um programa que calcule a média de uma lista de números fornecidos pelo usuário.
8. Escreva um programa que encontre o maior elemento em um vetor.
9. Implemente um programa que determine se um número é primo.
10. Crie um programa que calcule a raiz quadrada de um número fornecido.
11. Desenvolva um programa que receba uma lista de números e exiba apenas os números pares.
12. Elabore um programa que realize a ordenação crescente de um vetor.
13. Implemente um programa que leia uma string e substitua todas as ocorrências de uma letra especificada por outra.
14. Escreva um programa que calcule a potência de um número.
15. Crie um programa que conte o número de vezes que uma determinada letra aparece em uma string.
16. Desenvolva um programa que calcule a média ponderada de três notas.
17. Elabore um programa que remova os espaços em branco de uma string.
18. Implemente um programa que encontre o menor elemento em um vetor.
19. Escreva um programa que converta uma string para letras maiúsculas.
20. Crie um programa que determine se um ano é bissexto.
21. Desenvolva um programa que calcule a média aritmética de uma lista de números.
22. Elabore um programa que verifique se uma string é um palíndromo.
23. Implemente um programa que leia uma lista de números e exiba apenas os números ímpares.
24. Escreva um programa que calcule a soma dos dígitos de um número inteiro.
25. Crie um programa que encontre o segundo maior elemento em um vetor.
26. Desenvolva um programa que conte o número de palavras em uma string.
27. Elabore um programa que substitua todas as vogais em uma string por asteriscos.
28. Implemente um programa que calcule o fatorial de um número usando recursividade.

29. Escreva um programa que inverta a ordem dos elementos em um vetor.
30. Crie um programa que verifique se uma string é um palíndromo ignorando espaços em branco e maiúsculas/minúsculas.

Squad 2:

1. Desenvolva um programa que calcule a média geométrica de uma lista de números.
2. Crie um programa que determine se uma string é um palíndromo, ignorando espaços em branco e diferenciação entre maiúsculas e minúsculas.
3. Elabore um programa que leia uma lista de números e exiba apenas os números que são divisíveis por 3 e 5.
4. Implemente um programa que encontre a posição de um elemento específico em um vetor.
5. Escreva um programa que converta uma string para letras minúsculas.
6. Desenvolva um programa que verifique se um número é um quadrado perfeito.
7. Crie um programa que calcule a soma dos dígitos de um número usando um loop.
8. Elabore um programa que ordene um vetor em ordem decrescente.
9. Implemente um programa que determine se uma string é um pangrama (contém todas as letras do alfabeto).
10. Escreva um programa que calcule a média harmônica de uma lista de números.
11. Desenvolva um programa que remova os elementos duplicados de um vetor.
12. Crie um programa que determine se um número é positivo, negativo ou zero.
13. Elabore um programa que substitua todas as ocorrências de uma palavra por outra em uma string.
14. Implemente um programa que calcule a distância euclidiana entre dois pontos em um plano cartesiano.
15. Escreva um programa que encontre o menor número primo maior que um número fornecido.
16. Desenvolva um programa que conte o número de palavras palíndromas em uma lista de strings.
17. Desenvolva um programa que calcule a soma dos dígitos de um número em uma base específica (além da decimal).
18. Elabore um programa que verifique se um número é uma potência de 2.
19. Implemente um programa que calcule o resto da divisão de um número por outro.
20. Escreva um programa que determine se uma string é um anagrama de outra.
21. Desenvolva um programa que leia uma lista de números e exiba apenas os números primos.
22. Crie um programa que calcule o produto escalar de dois vetores.
23. Elabore um programa que inverta as palavras em uma string.
24. Implemente um programa que calcule a média ponderada de duas notas com pesos diferentes.
25. Escreva um programa que determine se um número é triangular.

26. Desenvolva um programa que conte o número de caracteres alfabéticos em uma string.
27. Crie um programa que verifique se uma string é um número válido em notação científica.
28. Elabore um programa que verifique se um número é um número de Fibonacci.
29. Implemente um programa que calcule a mediana de uma lista de números.
30. Escreva um programa que determine se uma string é uma subsequência de outra.

Squad 3:

1. Desenvolva um programa que calcule o cosseno de um ângulo em graus.
2. Elabore um programa que leia uma lista de palavras e retorne a palavra com o menor número de caracteres.
3. Elabore um programa que leia uma lista de palavras e retorne a palavra de maior comprimento.
4. Implemente um programa que determine se uma string é um número válido (inteiro ou decimal).
5. Escreva um programa que conte o número de ocorrências de cada palavra em uma string.
6. Desenvolva um programa que determine se um número é um quadrado mágico.
7. Crie um programa que remova os elementos repetidos de uma lista encadeada.
8. Elabore um programa que converta um número decimal para binário.
9. Implemente um programa que conte o número de anagramas de uma palavra em uma lista de palavras.
10. Implemente um programa que determine se uma string é um palíndromo, considerando apenas os caracteres alfanuméricos.
11. Escreva um programa que conte o número de anagramas de uma palavra em uma lista de palavras.
12. Crie um programa que verifique se uma string é um palíndromo, considerando apenas os caracteres alfabéticos.
13. Elabore um programa que leia uma lista de números e exiba apenas os números que formam uma sequência crescente.
14. Implemente um programa que calcule a média ponderada de três notas com pesos diferentes.
15. Escreva um programa que ordene uma lista encadeada de números.
16. Desenvolva um programa que determine se uma string é uma sequência de Fibonacci.
17. Crie um programa que encontre o segundo menor elemento em um vetor.
18. Elabore um programa que calcule a raiz cúbica de um número.
19. Implemente um programa que conte o número de caracteres não alfabéticos em uma string.
20. Desenvolva um programa que determine se um número é um número de Kaprekar.
21. Desenvolva um programa que calcule a média harmônica ponderada de uma lista de números.
22. Crie um programa que verifique se uma string é um acrônimo.
23. Elabore um programa que leia uma lista de números e retorne o produto dos números negativos.
24. Implemente um programa que determine se um número é primo de Mersenne.
25. Escreva um programa que conte o número de elementos em comum entre dois vetores.
26. Desenvolva um programa que calcule a soma dos quadrados dos números em uma lista.

27. Crie um programa que remova os espaços em branco extras de uma string.
28. Elabore um programa que verifique se uma lista de números é uma permutação de outra lista.
29. Implemente um programa que calcule a média ponderada de uma lista de números.
30. Escreva um programa que determine se uma string é um número romano válido.

Squad 4:

1. Desenvolva um programa que calcule o fatorial duplo de um número.
2. Elabore um programa que converta um número octal para decimal.
3. Elabore um programa que leia uma lista de palavras e retorne a palavra que aparece com mais frequência.
4. Implemente um programa que determine se uma string é um palíndromo, considerando apenas os dígitos numéricos.
5. Escreva um programa que conte o número de subtrings palíndromas em uma string.
6. Desenvolva um programa que determine se um número é um número feliz.
7. Crie um programa que remova os elementos duplicados de um vetor sem usar estruturas de dados adicionais.
8. Elabore um programa que converta um número binário para decimal.
9. Implemente um programa que encontre o número mais frequente em um vetor.
10. Escreva um programa que ordene uma lista de strings por tamanho.
11. Desenvolva um programa que calcule o mínimo múltiplo comum (MMC) de dois números.
12. Implemente um programa que encontre o elemento que mais se repete em um vetor.
13. Elabore um programa que leia uma lista de números e retorne a soma dos números primos.
14. Implemente um programa que calcule a média ponderada de uma lista de números, onde os pesos são fornecidos pelo usuário.
15. Escreva um programa que inverta a ordem das palavras em uma string.
16. Escreva um programa que ordene uma lista de strings por ordem lexicográfica, ignorando maiúsculas e minúsculas.
17. Crie um programa que encontre a posição da última ocorrência de um elemento em um vetor.
18. Elabore um programa que calcule a soma dos dígitos de um número usando recursividade.
19. Implemente um programa que determine se um número é um cubo perfeito.
20. Escreva um programa que conte o número de palavras em uma lista de strings, excluindo palavras repetidas.
21. Desenvolva um programa que calcule o número de combinações possíveis de um conjunto de elementos.
22. Desenvolva um programa que calcule o produto escalar de dois vetores tridimensionais.
23. Elabore um programa que leia uma lista de números e retorne a média dos números pares.
24. Implemente um programa que calcule o máximo divisor comum (MDC) de dois números.
25. Escreva um programa que determine se uma string é um pangrama perfeito.
26. Desenvolva um programa que inverta a ordem dos elementos em uma lista encadeada.

27. Crie um programa que verifique se uma string é um palíndromo perfeito.
28. Elabore um programa que verifique se uma string é uma subsequência de outra, ignorando espaços em branco.
29. Implemente um programa que calcule a média ponderada de uma lista de notas com pesos distintos para cada nota.
30. Escreva um programa que determine se um número é um número de Lucas.

Squad 5:

1. Desenvolva um programa que calcule o coeficiente binomial de dois números inteiros.
2. Elabore um programa que leia uma lista de números e retorne a média dos números pares e ímpares separadamente.
3. Elabore um programa que leia uma lista de palavras e retorne a palavra que contém o maior número de vogais.
4. Implemente um programa que determine se uma string é um palíndromo de frase (considerando pontuações e espaços).
5. Escreva um programa que conte o número de substrings distintas em uma string.
6. Desenvolva um programa que determine se um número é um número de Armstrong.
7. Crie um programa que remova os elementos duplicados de um vetor sem alterar a ordem original.
8. Elabore um programa que converta um número decimal para octal.
9. Implemente um programa que encontre o número menos frequente em um vetor.
10. Escreva um programa que ordene uma lista de strings por ordem lexicográfica reversa.
11. Desenvolva um programa que calcule a série harmônica de um número.
12. Implemente um programa que calcule o n-ésimo termo de uma sequência de Fibonacci.
13. Elabore um programa que leia uma lista de números e retorne a soma dos números ímpares.
14. Implemente um programa que calcule a média geométrica ponderada de uma lista de números.
15. Escreva um programa que inverta a ordem dos elementos em uma pilha.
16. Escreva um programa que inverta a ordem dos elementos em uma lista ligada simples.
17. Crie um programa que encontre a posição da primeira ocorrência de um elemento em uma lista encadeada.
18. Elabore um programa que calcule a soma dos algarismos primos de um número.
19. Implemente um programa que determine se um número é um quadrado mágico.
20. Escreva um programa que conte o número de palavras em uma lista de strings, considerando apenas palavras distintas.
21. Desenvolva um programa que calcule o coeficiente binomial usando a fórmula de Pascal.
22. Desenvolva um programa que determine se um número é um número triangular centrado.
23. Elabore um programa que leia uma lista de números e retorne a média dos números negativos.
24. Implemente um programa que calcule a soma dos quadrados dos números ímpares.

25. Escreva um programa que determine se um número é um número de Catalan.
26. Desenvolva um programa que inverta a ordem dos elementos em uma fila.
27. Crie um programa que encontre o maior número primo menor que um número fornecido.
28. Elabore um programa que verifique se uma string é um palíndromo de palavra (desconsiderando espaços e pontuações).
29. Implemente um programa que calcule a média ponderada de uma lista de notas, onde os pesos são gerados aleatoriamente.
30. Escreva um programa que determine se um número é um número feliz.