



CURSO | TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Professor

Ivan Carlos Alcântara de Oliveira

Componente

Estrutura de dados

Aula

Aula 6

Aplicando conhecimento

Tendo por base o conteúdo encontrado no arquivo compactado “_Aula6_ED_ListasLigadas.zip”, desenvolva o solicitado para a “_Aula6_ED”.

O arquivo compactado “_Aula6_ED_ListasLigadas.zip” contém:

- *2024.2_TADS_EstruturasDeDados_6.pdf: este documento com as orientações da atividade.*
- *O projeto Java no Eclipse compactado (_Aula6_ED.zip) contendo as classes:*
 - *_Aula6_ED: encarregada de executar alguns testes/funcionalidades com objetos da classe “LinkedList”. Nesta classe, há um (1) método que não foi criado e que deve ser elaborado nesta atividade. Esta classe cria um banco de dados de Trabalhadores, para isso, faz uso da LinkedList (lista encadeada);*
 - *LinkedList.java: classe LinkedList contém métodos previamente desenvolvidos para uma estrutura de dados Lista Ligada genérica, ou seja, que pode armazenar objetos de qualquer tipo. Nela, há um (1) método que não foi desenvolvido e que deve ser criado por você;*
 - *Node.java: classe Node contém os atributos dado e prox, e representa um nó (node) da lista ligada. O “dado” é o elemento do tipo genérico armazenado e “prox” é o ponteiro para o próximo Node (nó);*
 - *Trabalhador: classe relativa a um Trabalhador contendo os atributos id, idade, sexo, salário e cargo.*
- *ADS_ED_AulaVI_template.docx: arquivo do Word no qual deve ser publicado os códigos fontes para os métodos desenvolvidos e os testes realizados.*

Como sugestão, entenda todos os códigos fontes previamente desenvolvidos antes de elaborar o solicitado nesta atividade, ok!

Os dois métodos que devem ser construídos são:

- **public static void demiteFuncionarios(LinkedList<Trabalhador> func):** método da classe “_Aula6_ED” encarregado de remover funcionários



demitidos da `LinkedList` `func` que armazena objetos do tipo `Trabalhador`. Enquanto o usuário desejar, você deve solicitar o número do funcionário (id) a ser demitido (removido da lista), se o funcionário existir, deve ser removido da `LinkedList` e apresentada a mensagem “Funcionário ID encontrado e removido com sucesso!”. Caso não exista na lista, apresenta a mensagem “Funcionário ID não foi encontrado!”. Para melhor compreender o solicitado, veja um exemplo de execução apresentado a seguir.

- **`public void concatena(LinkedList<T> lista)`**: método interno à classe `LinkedList` responsável por concatenar a “lista” passada como parâmetro ao final da lista original da classe `LinkedList`. Para melhor compreender o solicitado, veja um exemplo de execução apresentado a seguir.

Ao realizar o solicitado, publique os códigos fontes criados (basta colocar os métodos elaborados por você) no template “ADS_ED_AulaVI_template.docx” anexo com os testes realizados. Depois, entregar o template convertido para PDF com sua solução e testes junto aos códigos fonte em Java do programa completo desenvolvido.

Tendo por base o trecho do banco de dados e o main da classe `_Aula6_ED` abaixo:

```
// bancoDeDados(LinkedList<Trabalhador> func): realiza o cadastro de 10
// trabalhadores na lista ligada (LinkedList) func.
public static void bancoDeDados(LinkedList<Trabalhador> func) {
    func.addLast(new Trabalhador(1, 25, 'M', 5000.0f, "Analista Jr.));
    func.addLast(new Trabalhador(2, 22, 'M', 3000.0f, "Auxiliar Administrativo.));
    func.addLast(new Trabalhador(3, 27, 'F', 5000.0f, "Analista Jr.));
    func.addLast(new Trabalhador(4, 35, 'M', 7000.0f, "Programador Sr.));
    func.addLast(new Trabalhador(5, 45, 'F', 7000.0f, "Gerente Jr.));
    func.addLast(new Trabalhador(6, 51, 'M', 7500.0f, "Gerente Pleno));
    func.addLast(new Trabalhador(7, 60, 'M', 8000.0f, "Vice-Presidente));
    func.addLast(new Trabalhador(8, 34, 'F', 7000.0f, "Analista Pleno));
    func.addLast(new Trabalhador(9, 43, 'F', 8000.0f, "Programador Sr.));
    func.addLast(new Trabalhador(10, 63, 'F', 10000.0f, "Presidente));
}
```



```
public static void main(String[] args) {
    LinkedList<Trabalhador> func = new LinkedList<Trabalhador>();

    bancoDeDados(func);
    System.out.println(func);

    LinkedList<Trabalhador> contratados = new LinkedList<Trabalhador>();

    contratados.addLast(new Trabalhador(11, 23, 'M', 3900.0f, "Programador Jr.));
    contratados.addLast(new Trabalhador(12, 29, 'F', 4500.0f, "Secretária Bilíngue"));
    contratados.addLast(new Trabalhador(13, 35, 'M', 6000.0f, "Programador Pleno"));
    contratados.addLast(new Trabalhador(14, 53, 'M', 5000.0f, "Programador Jr.));
    contratados.addLast(new Trabalhador(15, 63, 'F', 8000.0f, "Analista Sr.));
    contratados.addLast(new Trabalhador(16, 56, 'M', 9000.0f, "Analista Sr.));

    func.concatena(contratados);
    System.out.println(func);

    demiteFuncionarios(func);
    System.out.println(func);
}
```

Exemplo de testes realizados com a execução do projeto Java no Eclipse contendo todos os métodos desenvolvidos é apresentado abaixo:

```
[Lista]
L: [
[Trabalhador] Id: 1, sexo: M, idade: 25, cargo: Analista Jr., salário: 5000.0
[Trabalhador] Id: 2, sexo: M, idade: 22, cargo: Auxiliar Administrativo., salário: 3000.0
[Trabalhador] Id: 3, sexo: F, idade: 27, cargo: Analista Jr., salário: 5000.0
[Trabalhador] Id: 4, sexo: M, idade: 35, cargo: Programador Sr., salário: 7000.0
[Trabalhador] Id: 5, sexo: F, idade: 45, cargo: Gerente Jr., salário: 7000.0
[Trabalhador] Id: 6, sexo: M, idade: 51, cargo: Gerente Pleno, salário: 7500.0
[Trabalhador] Id: 7, sexo: M, idade: 60, cargo: Vice-Presidente, salário: 8000.0
[Trabalhador] Id: 8, sexo: F, idade: 34, cargo: Analista Pleno, salário: 7000.0
[Trabalhador] Id: 9, sexo: F, idade: 43, cargo: Programador Sr., salário: 8000.0
[Trabalhador] Id: 10, sexo: F, idade: 63, cargo: Presidente, salário: 10000.0 ]
Qtde.: 10

[Lista]
L: [
[Trabalhador] Id: 1, sexo: M, idade: 25, cargo: Analista Jr., salário: 5000.0
[Trabalhador] Id: 2, sexo: M, idade: 22, cargo: Auxiliar Administrativo., salário: 3000.0
[Trabalhador] Id: 3, sexo: F, idade: 27, cargo: Analista Jr., salário: 5000.0
[Trabalhador] Id: 4, sexo: M, idade: 35, cargo: Programador Sr., salário: 7000.0
[Trabalhador] Id: 5, sexo: F, idade: 45, cargo: Gerente Jr., salário: 7000.0
[Trabalhador] Id: 6, sexo: M, idade: 51, cargo: Gerente Pleno, salário: 7500.0
[Trabalhador] Id: 7, sexo: M, idade: 60, cargo: Vice-Presidente, salário: 8000.0
[Trabalhador] Id: 8, sexo: F, idade: 34, cargo: Analista Pleno, salário: 7000.0
[Trabalhador] Id: 9, sexo: F, idade: 43, cargo: Programador Sr., salário: 8000.0
[Trabalhador] Id: 10, sexo: F, idade: 63, cargo: Presidente, salário: 10000.0
[Trabalhador] Id: 11, sexo: M, idade: 23, cargo: Programador Jr., salário: 3900.0
[Trabalhador] Id: 12, sexo: F, idade: 29, cargo: Secretária Bilíngue, salário: 4500.0
[Trabalhador] Id: 13, sexo: M, idade: 35, cargo: Programador Pleno, salário: 6000.0
[Trabalhador] Id: 14, sexo: M, idade: 53, cargo: Programador Jr., salário: 5000.0
[Trabalhador] Id: 15, sexo: F, idade: 63, cargo: Analista Sr., salário: 8000.0
[Trabalhador] Id: 16, sexo: M, idade: 56, cargo: Analista Sr., salário: 9000.0 ]
Qtde.: 16
```



Deseja demitir um funcionário (S/N)?

s

Forneça o número do funcionário:

10

Funcionário 10 encontrado e removido com sucesso!

Deseja continuar a demitir um funcionário (S/N)?

s

Forneça o número do funcionário:

5

Funcionário 5 encontrado e removido com sucesso!

Deseja continuar a demitir um funcionário (S/N)?

s

Forneça o número do funcionário:

8

Funcionário 8 encontrado e removido com sucesso!

Deseja continuar a demitir um funcionário (S/N)?

s

Forneça o número do funcionário:

5

Funcionário 5 NÃO foi encontrado!

Deseja continuar a demitir um funcionário (S/N)?

n

[Lista]

L: [

[Trabalhador] Id: 1, sexo: M, idade: 25, cargo: Analista Jr., salário: 5000.0
[Trabalhador] Id: 2, sexo: M, idade: 22, cargo: Auxiliar Administrativo., salário: 3000.0
[Trabalhador] Id: 3, sexo: F, idade: 27, cargo: Analista Jr., salário: 5000.0
[Trabalhador] Id: 4, sexo: M, idade: 35, cargo: Programador Sr., salário: 7000.0
[Trabalhador] Id: 6, sexo: M, idade: 51, cargo: Gerente Pleno, salário: 7500.0
[Trabalhador] Id: 7, sexo: M, idade: 60, cargo: Vice-Presidente, salário: 8000.0
[Trabalhador] Id: 9, sexo: F, idade: 43, cargo: Programador Sr., salário: 8000.0
[Trabalhador] Id: 11, sexo: M, idade: 23, cargo: Programador Jr., salário: 3900.0
[Trabalhador] Id: 12, sexo: F, idade: 29, cargo: Secretária Bilíngue, salário: 4500.0
[Trabalhador] Id: 13, sexo: M, idade: 35, cargo: Programador Pleno, salário: 6000.0
[Trabalhador] Id: 14, sexo: M, idade: 53, cargo: Programador Jr., salário: 5000.0
[Trabalhador] Id: 15, sexo: F, idade: 63, cargo: Analista Sr., salário: 8000.0
[Trabalhador] Id: 16, sexo: M, idade: 56, cargo: Analista Sr., salário: 9000.0]

Qtde.: 13