

### Instruções gerais:

1. Utilize o Eclipse ou o STS para desenvolver os algoritmos.
2. Ao concluir os exercícios, envie todos os códigos criados no Eclipse ou no STS para o Repositório criado na sua conta pessoal do Github, em uma pasta identificada com o tema da sessão
3. Caso seja solicitado, adicione o **link do Github** ou os **prints** dos arquivos .JAVA, indicados pelo instrutor na Plataforma Canvas.

**Mantenha as entregas das Atividades em dia na Plataforma Canvas**

## EXERCÍCIOS

### Boas práticas:

1. Resolva **1 exercício da lista 01 - Collection ArrayList** e **1 exercício da lista 02 - Collection Set**.
2. Leia o enunciado do exercício com atenção
3. Observe as indicações de Entrada e Saída esperadas em cada exercício
4. Observe com atenção os desenhos e diagramas inseridos nos exercícios para facilitar a compreensão
5. Utilize o Cookbook, os Vídeos da Plataforma e os Códigos guia como referências para a resolução dos exercícios
6. Na entrega das atividades na Plataforma Canvas, efetue o **envio do print do exercício resolvido de cada lista**.
7. Caso ainda fique alguma dúvida, consulte os instrutores da sua turma pelo Discord

## Lista 01 – Collection List (Resolva 1 Exercício)

1) Escreva um programa Java para criar uma **Collection ArrayList** de Objetos da **Classe String**. O programa deverá solicitar ao usuário, que ele digite via teclado 5 cores e deverá adicioná-las individualmente no ArrayList. Em seguida, faça o que se pede:

- Mostre na tela todas as cores que foram adicionadas.
- Mostre na tela todas as cores que foram adicionadas ordenadas em ordem crescente.

Veja o exemplo abaixo:

ENTRADA	SAÍDA
Azul Verde Amarelo Branco Laranja	<b>Listar todas as cores:</b>  Azul Verde Amarelo Branco Laranja  <b>Ordenar as cores:</b>  Amarelo Azul Branco Laranja Verde

Na construção do Algoritmo, utilize os seguintes conteúdos:

- Entrada e Saída de dados
- Laços de Repetição
- **Collection ArrayList**

- 2) Escreva um programa Java para criar uma **Collection ArrayList** de Objetos da **Classe Wrapper Integer**, inicializada com 10 valores inteiros. O programa deverá solicitar ao usuário, que ele digite via teclado 1 número inteiro e caso ele seja encontrado no ArrayList, exiba na tela a posição deste número na Collection. Caso o número não seja encontrado, ele deverá exibir na tela a mensagem: **O número NN não foi encontrado!**

Veja o exemplo abaixo:

ENTRADA					SAÍDA				
<div> <div> 0123456789 </div> <div> List <div> 25134978106 </div> </div> </div> <p>Digite o número que você deseja encontrar: 7</p>					<p>O número 7 está localizado na posição: 6</p>				
<div> <div> 0123456789 </div> <div> List <div> 25134978106 </div> </div> </div> <p>Digite o número que você deseja encontrar: 40</p>					<p>O número 40 não foi encontrado!</p>				

*\* Os números em vermelho representam o índice (posição) na Collection.*

Na construção do Algoritmo, utilize os seguintes conteúdos:

- Entrada e Saída de dados
- Laços Condicionais
- Laços de Repetição
- **Collection ArrayList**

## Lista 02 – Collection Set (Resolva 1 Exercício)

3) Escreva um programa Java para criar uma **Collection Set** de Objetos da **Classe Wrapper Integer**. O programa deverá solicitar ao usuário, que ele digite via teclado 10 valores inteiros não repetidos e adicione-os individualmente na Collection Set. Em seguida, faça o que se pede:

- Mostre na tela todos os elementos da Collection Set, utilizando a **Classe Iterator**.

Veja o exemplo abaixo:

ENTRADA	SAÍDA
Set [2, 5, 1, 3, 4, 9, 7, 8, 10, 6]	Listar dados do Set: 1 2 3 4 5 6 7 8 9 10
Set [2, 5, 10, 3, 4, 2, 2, 3, 10, 5]	Listar dados do Set: 2 3 4 5 10

Na construção do Algoritmo, utilize os seguintes conteúdos:

- Entrada e Saída de dados
- **Collection Set**
- **Classe Iterator**

- 4) Escreva um programa Java para criar uma **Collection Set** de Objetos da **Classe Wrapper Integer**, inicializada com 10 valores inteiros. O programa deverá solicitar ao usuário, que ele digite via teclado 1 número inteiro e caso ele seja encontrado no Set, exiba na tela a mensagem: **Número 00 Encontrado!** Caso o número não seja encontrado, ele deverá exibir na tela a mensagem: **O número NN não foi encontrado!**

Veja o exemplo abaixo:

ENTRADA	SAÍDA
<div><div>0123456789</div><div>List<div>25134978106</div></div></div> <div>Digite o número que você deseja encontrar: 7</div>	O número 7 foi encontrado!
<div><div>0123456789</div><div>List<div>25134978106</div></div></div> <div>Digite o número que você deseja encontrar: 40</div>	O número 40 não foi encontrado!

Na construção do Algoritmo, utilize os seguintes conteúdos:

- Entrada e Saída de dados
- Laços Condicionais
- Laços de Repetição
- **Collection Set**