

## Lista 4 - Introdução a POO. Objetos e classes.

**Prazo de entrega:** 17/05/2024 até 23:59 (Sexta)

**Instruções de entrega:** Você deve ter um repositório em seu github para a disciplina de POO chamado **Programacao-Orientado-Objetos**.

Crie uma pasta dentro dele chamada Listas e dentro dela crie uma pasta chamada Lista4.

Para cada exercício da lista deve ser criado um projeto java.

Ao terminar a lista, suba seus exercícios no github e envie o link no formulário abaixo.

**Link para o formulário de envio:** <https://forms.gle/N9w59MAFxq8nh3236>

**MONITORIAS:** Serão realizados quatro dias de monitorias para auxiliar com a resolução dos exercícios. As datas e horários das monitorias são:

Dia	Horários		
Segunda-Feira	19h até 21h		
Terça-Feira	19h até 21h	-	-
Quarta-Feira	09h até 11h	19h até 21h	-
Quinta-Feira	19h até 21h	-	-
Sexta-Feira	09h até 11h	19h até 21h	22h até 23:59

**OBS:** Se você não conseguir participar das monitorias nos dias e horários propostos, você pode solicitar uma monitoria em uma data e horário diferente, desde que seja comunicado com pelo menos um dia de antecedência 😊

## EXERCÍCIO



### Criando Classes

Desenvolva um programa que contenha três classes para representar objetos. Duas devem representar coisas concretas e uma algo abstrato. Cada classe deve conter pelo menos três atributos e três métodos (um dos métodos deve imprimir todos os atributos da classe).

#### Instruções:

Exemplo de objetos concretos: carro, pessoa, conta bancária, animal, produto, etc...

Exemplo de objetos abstratos: corrida, conversa, forma geométrica, luta, aula, etc...

Você poderá escolher o que quiser para representar em uma classe.

Cada classe deve ter pelo menos três atributos que descrevem suas características principais.

Implemente pelo menos três métodos em cada classe (um dos métodos deve imprimir todos os atributos da classe).

Na função main da classe App, instancie um objeto de cada classe criada e chame o método que imprime os dados do objeto.

**Tente ser criativo ao escolher os objetos a serem representados pelas classes.**

