

## AT3/N2 - Atividade prática coletiva - Bimestre N2

Crie um projeto (**em Java 17**) para representar um servidor utilizando *sockets*. O servidor terá a função de controlar um registro/cadastro de livros de uma biblioteca, sendo capaz de executar as seguintes funcionalidades:

- Listagem dos livros;
- Aluguel e devolução de livros;
- Cadastro de livros.

Os livros devem ser representados por uma classe, que deve conter no mínimo os seguintes atributos:

- Autor;
- Nome;
- Gênero;
- Número de exemplares.

Os livros deverão ficar armazenados em um arquivo JSON (divulgado previamente pelo professor), que terá 10 livros inicialmente. **As alterações aplicadas pelo usuário (cadastro e aluguel) devem refletir no arquivo, pois o mesmo representará uma “base de dados” da biblioteca.**

As operações devem ser realizadas por um **cliente socket**, que também deve ser implementado pelo grupo. **Ou seja, deve haver o envio e recebimento de dados entre o cliente e o servidor.**

### Observações:

- O arquivo JSON deverá manter sua estrutura correta (conforme explicado [aqui](#)) após a aplicação das operações;
- A comunicação entre cliente e servidor deve ser feita **obrigatoriamente** por meio de *sockets*.

## Critérios de avaliação

O projeto vale no máximo 4 pontos na nota do bimestre (N1), sendo que a avaliação será baseada em dos tipos de critérios:

### Coletivos (3 pontos):

- **Criação da classe do livro:** 0.1;
  - Correta implementação de cada classe, com atributos e métodos;
- **Implementação das operações :** 0.9 (0.3 por operação);
  - **Serão considerados a lógica utilizada e a corretude das operações.**
  - **Também será checado se as operações atualizam o arquivo JSON corretamente.**
- **Criação do servidor:** 0.7
  - Correta implementação, usando as classes necessárias e realizando operações de entrada e saída corretamente.
- **Criação do cliente:** 0.7
  - Correta implementação, usando as classes necessárias e realizando operações de entrada e saída corretamente.
- **Estruturação e organização do código:** 0.5;
  - Divisão do projeto em classes;
  - Modularização do código, utilizando métodos sempre que possível;
  - Organização/clareza do código (nomes significativos de variáveis, indentação, etc.);
- **Organização do GitHub (README, comentários e tamanho dos commits, etc.)** (0.1);

### Individuais (1 ponto):

- Participação no desenvolvimento do projeto (quantidade e qualidade dos commits e proporção de commits feitos por cada membro) (0.5);
- Participação na apresentação final do projeto (0.5).

## Observações

- **O trabalho deve ser feito em grupo, mas qualquer tipo de plágio/cola será penalizado (o projeto receberá nota 0).**
- **Projetos com erro de sintaxe ou que não possam ser executados irão receber nota 0 também**
- A entrega será dia 17/06, até às 23:59, pelo AVA.
- O envio deve conter apenas um link para o repositório (que deve estar público) com o código do projeto.
  - **NÃO ENVIEM UM ZIP OU COLOQUEM O PROJETO ZIPADO NO GITHUB!**