

MC322 – Programação Orientada a Objetos

Laboratório 06 – 2s2023

Prof. Dr. Bruno Cafeo (Professor)

Fillipe dos Santos Silva

Jesamin Zevallos Quispe

Vinicius Leme Soares

Discord: <https://discord.gg/mHcUFRDGgQ>

1 Coleções

Uma Biblioteca Universitária possui uma grande variedade de itens multimídia como livros, CDs, DVDs, entre outros. Para gerenciar esses itens e as operações de empréstimo, reserva e renovação, a biblioteca precisa de um sistema eficiente. Uma das formas de organizar e acessar rapidamente as informações é através do uso de Coleções.

1. Membros da Biblioteca:

- Utilize uma `List<Membro>` para armazenar todos os membros. Permita adição, remoção e busca por `id`.

2. Itens da Biblioteca:

- Utilize um `Map<Integer, Item>` para armazenar todos os itens da biblioteca, usando o `id` do item como chave.

3. Empréstimos:

- Utilize um `Set<Emprestimo>` para garantir que um item não seja emprestado para dois membros ao mesmo tempo.

4. Reservas:

- Utilize uma `List<Reserva>` para armazenar as reservas, garantindo que a ordem seja mantida.

5. Categorias:

- Utilize um `Set<Categoria>` para armazenar todas as categorias únicas disponíveis na biblioteca. Por exemplo, “Ciências”, “Artes”, “Tecnologia”, etc.

6. Eventos:

- Para armazenar todos os membros que estão participando de um determinado evento, use um `List<Membro>` dentro da classe `Evento`.

7. Comentários:

- Implemente a possibilidade de membros deixarem comentários sobre itens da biblioteca. Para isso, crie uma classe `Comentario` com atributos como `membroId`, `itemId` e `texto`.
- Dentro da classe `Item`, use um `List<Comentario>` para armazenar todos os comentários relacionados a um item específico.

8. Histórico de Empréstimos:

- Mantenha um registro de todos os empréstimos anteriores de um membro. Para isso, na classe `Membro`, use um `List<Emprestimo>` para armazenar o histórico.

Pontos a serem pensados e implementados:

- a) Escreva um método para adicionar um novo membro à lista de membros. Se um membro com o mesmo `id` já existir, retorne uma mensagem de erro.
- b) Escreva um método para emprestar um item a um membro. Se o item já estiver emprestado, adicione a requisição à lista de reservas.
- c) Escreva um método para retornar um item. Se houver reservas para o item, garanta que o próximo membro na fila de reservas o receba.
- d) Raciocine sobre a coleção que você usaria para armazenar todos os itens que um membro já pegou emprestado
- e) Raciocine sobre como você modificaria o sistema para lidar com casos em que um membro não devolve um item no prazo estipulado

2 Avaliação

- Entrega realizada dentro do prazo estipulado.
- Execução do código.
- Qualidade do código desenvolvido (saída dos dados na tela, tabulação, comentários).
- Desenvolvimento correto das Coleções.
- **Entregar no diretório files do repositório o diagrama de classes UML completo e atualizado em relação ao laboratório anterior.**

3 Entrega

- Você não precisa criar outro repositório ou outro projeto, a ideia é avançar sobre o repositório que você já tem.
- A entrega do Laboratório é realizada exclusivamente via **Github**. Para a submissão no Github, gere um release (tag) com a identificação do laboratório no formato <lab06-RA>. Por exemplo, para o aluno com RA 123456, a tag será: lab06-123456.
- Evite criar *releases* enquanto não tiver certeza que seu código está funcionando como esperado.
- Utilize os horários de laboratório e atendimentos para tirar eventuais dúvidas de submissão e também relacionadas ao desenvolvimento do laboratório.
- Data da entrega: **17/09/2023** até às **23h59**.