



**Universidade Federal de Ouro Preto - UFOP**  
**Instituto de Ciências Exatas e Biológicas - ICEB**  
**Departamento de Computação - DECOM**  
**Disciplina: Programação Orientada a Objetos**  
**Professor: Guillermo Cámara Chávez**

### **Orientações para a prática:**

- Esta é uma prática avaliativa e deve ser feita em grupos de 5 membros.
- É permitido consultar os materiais de aula e também materiais disponíveis na internet.
- Todos os programas devem ser implementados considerando os conceitos de Orientação a Objetos estudados até o momento.
- Cada exercício deverá ser feito em um projeto e ao final, os códigos fonte deverão ser disponibilizados neste documento.
- Todos os exercícios devem ser feitos utilizando o NetBeans em um projeto.
- Deve ser incluído um vídeo onde seja mostrado o uso do programa

### **Programação Orientada a Objetos – Parte I**

O trabalho consiste em desenvolver uma aplicação desktop utilizando a linguagem de programação Java. A aplicação proposta nesta prática tem como objetivo o gerenciamento de empréstimos de livros em uma biblioteca.

O sistema deverá estar apto a realizar o cadastro de funcionários, livros, usuários e empréstimos, além disso, deverá ser possível realizar o cadastro dos autores e também as categorias dos livros, como por exemplo, terror, ficção, infantil entre outros. Um livro poderá ser escrito por um ou mais autores e também poderá estar classificado como uma ou mais categorias. As operações de cadastro somente poderão ser realizadas por um usuário administrador identificado, o qual não precisa estar cadastrado no sistema, a operação de empréstimo somente poderá ser realizada por um funcionário a um usuário comum, o usuário comum poderá realizar a consulta de livros. Todas as operações são

simuladas, portanto não é necessário realizar a persistência dos dados após o encerramento da aplicação.

As classes da aplicação devem estar distribuídas em pacotes de acordo com suas funções, os nomes dos pacotes estão listados a seguir: Principal, Modelo, DAO e Dados.

O pacote Principal conterá apenas a classe principal da sua aplicação, aquela que possui o método `main(..)`, responsável pela execução e gerenciamento da aplicação, nessa classe poderá ser implementado o menu que irá invocar as interfaces de entradas de dados, além do login do administrador, funcionário ou usuário comum.

No pacote Modelo devem estar as classes que representam as entidades do negócio, nesse caso, considere como exemplos as classes no diagrama da Figura 1. Nessa figura é possível observar relacionamentos de herança e composição. As classes Funcionário, Usuário e Autor tem um relacionamento de herança com a classe Pessoa. Além disso existem relacionamentos de composição entre as classes Livro em relação a

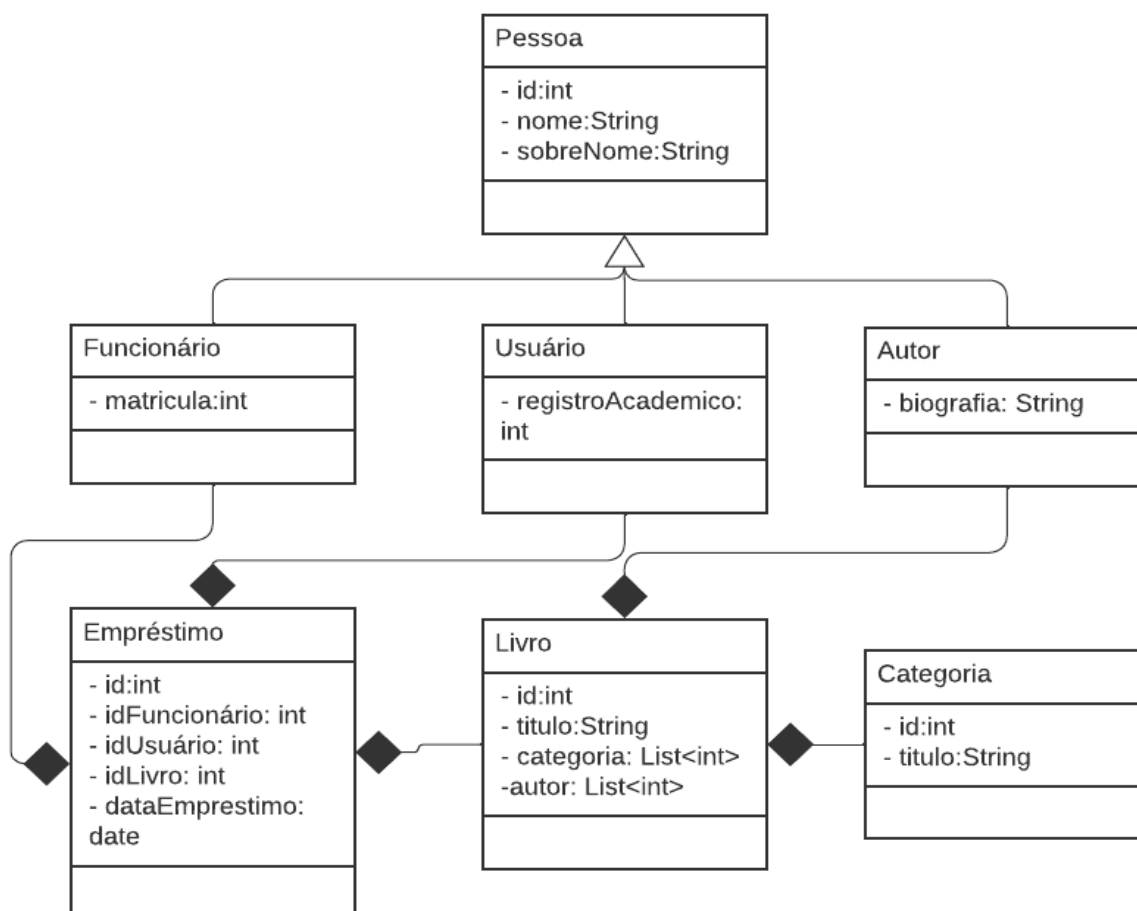


Figura 1 - Diagrama de Classes do pacote Modelo

Autor e Categoria, a classe Empréstimo também possui relacionamento de composição junto as classes Livro, Usuário e Funcionário.

Os atributos de todas classes deverão estar encapsulados, ou seja, deverão ter o modificador de visibilidade privado e deverão ter os respectivos métodos *get* e *set*, exceto os atributos da classe Dados, esses poderão ser acessados diretamente.

No pacote Dados deverá ser criada uma classe responsável pelo armazenamento dos dados da aplicação, ela pode ser nomeada como Dados. Para cada uma das classes das entidades do pacote modelo deverá ser criada uma lista estática correspondente, como atributo da classe Dados. Cada uma dessas listas representa, de forma análoga, uma tabela em um banco de dados relacional e armazenará os objetos das entidades do mundo real. Como pode ser observado na imagem da Figura 2, todos os atributos poderão ser públicos, com isso não são necessários métodos auxiliares para acesso.

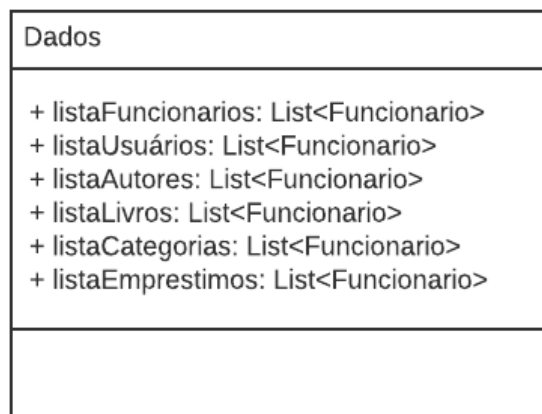


Figura 2 - Classe Dados

No pacote DAO deverão ser criadas classes, seguindo o padrão DAO (*Data Access Object*), para manipular os objetos instanciados a partir de cada uma das classes do pacote Modelo. Essas classes serão responsáveis pela criação, manipulação e exclusão dos objetos armazenados nas listas da classe Dados. Com o objetivo de garantir que todas as classes do pacote DAO implementem todos os métodos para manipulação dos objetos armazenados é importante criar uma Interface DAO, cuja especificação desses métodos esteja definida, e garantir que todas as classes implementem essa interface. A Figura 3 apresenta um exemplo de uma classe DAO para Funcionários, além da Interface.

Para cada uma das classes que representam as entidades do negócio, aquelas do pacote Modelo, deverão ser criadas classes DAO para manipular seus objetos.

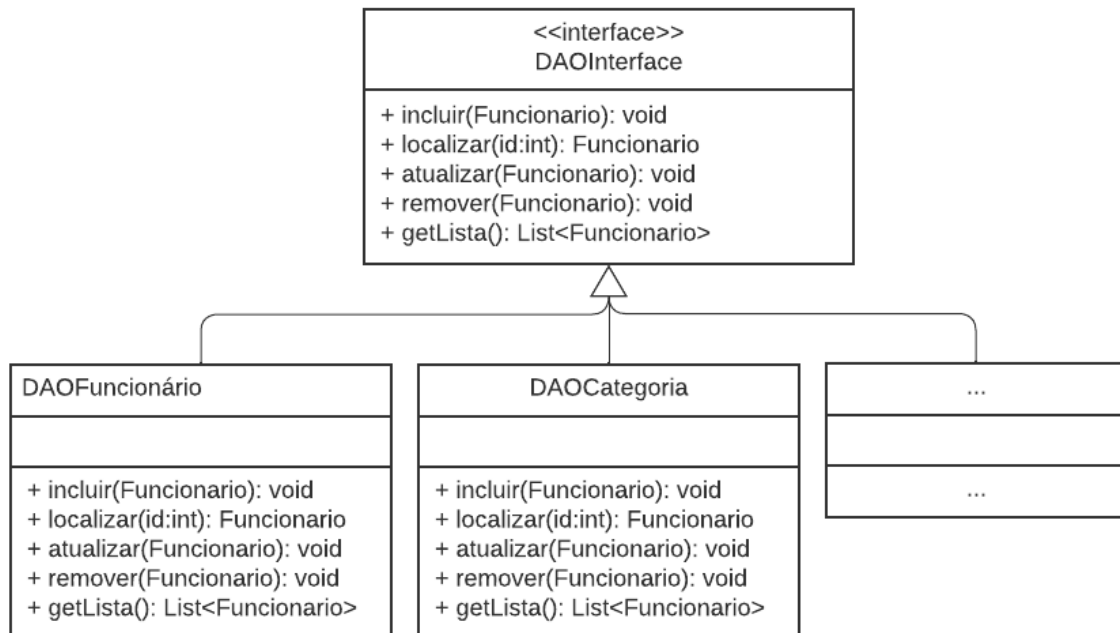


Figura 3 - Classes DAO

Esta é a primeira parte da aplicação, ao final dessa prática espera-se ter uma aplicação funcional com a definição das classes de modelo, as classes DAO para manipulação dos dados, uma classe contendo listas para simular um banco de dados (persistência em memória) e a classe principal da aplicação, para execução e gerenciamento da aplicação. A entrada de dados poderá ser realizada neste momento através do Scanner.

➤ A entrega deve ter os seguintes artefatos:

- Relatório técnico conforme o modelo “\_Prática Modelo de Entrega.docx”.
- O código fonte em formato Zip, exportado pelo NetBeans.
- Serão considerados como critérios de avaliação.
  - Códigos que não compilarem e tiverem *warnings* terão nota reduzida;
  - Código devidamente comentados;

- iii. Indentação do código e boas práticas de programação;
- iv. Uso dos conceitos de Orientação a Objetos como: Classes, Objetos, Polimorfismo, Herança, Tratamento de Exceções, Coleções entre outros.

1. Apresentar a especificação de requisitos do seu projeto conforme modelo de relatório em anexo. (0,25)
2. Implementação do projeto em Java. (0,75)