

## **Laboratório 4**

---

### **1 Objetivo**

O objetivo desta atividade consiste na prática de Hierarquias de generalização/especialização através de herança simples e sobrescrita de métodos.

Os seguintes passos **devem** ser tomados para a criação do projeto:

1. Abra o Eclipse.
2. Crie um novo projeto (File -> New -> Java Project).
3. Digite o nome do projeto **Lab4RAxxxxxx** onde **xxxxxx** deve ser substituído pelo seu RA.
4. Não inclua a opção "Create modulo-info.java file"(desmarque se estiver marcado).
5. Clique em Finish.

### **2 Atividade**

Nesta atividade o principal foco será a familiarização com generalização e herança em java e a programação de algumas classes: **Carta**, **Baralho**, e **Magia**.

#### **2.1 Primeiramente...**

1. Crie o projeto do Lab4 e, dentro dele, os pacotes *base* e *util*.
2. Reutilize as classes **BaralhoArrayList** e **CartaLacaio** criadas nos laboratórios anteriores, colando-as no pacote *base*. Renomeie a classe **BaralhoArrayList** para **Baralho** e **CartaLacaio** para **Lacaio**.
3. Acrescente uma classe vazia **Magia** no pacote *base* e copie a classe **Util** já criada para o pacote *util*.
4. **NÃO UTILIZE ACENTOS EM NUNHUM PRINT E EM NENHUMA PARTE DOS SEUS ARQUIVOS**

#### **2.2 A classe base Carta**

1. Crie a classe **Carta** no pacote *base* com atributos:
  - **private UUID id;** (identificador único de cada carta a ser setado no construtor com a chamada **UUID.randomUUID()**). Você deve fazer o import da biblioteca **import java.util.UUID;**
  - **private String nome**
  - **private int custoMana**
2. E métodos

- Métodos getters e setters para todos atributos, exceto o id que não será alterado após ser instanciado um objeto.
- Método **toString()** da classe **Carta** deve retornar um objeto da classe **String** contendo uma descrição geral dos atributos da carta.
- A classe **Carta** deverá possuir um método **usar(Carta alvo)** vazio que será sobrescrito nas subclasses para modificar os atributos de uma carta **alvo**.

## 2.3 A classe Lacaio

1. Faça **Lacaio** estender **Carta**. Remova todos os atributos repetidos (i.e., todos os atributos de **Lacaio** que já foram implementados por **Carta**).
2. Atributos privados de Lacaio: **int ataque**, **int vidaAtual** e **int vidaMaxima**
3. Faça os métodos setters e getters destes atributos.
4. Modifique os construtores de **Lacaio** de tal forma que estes reutilizem um ou mais construtores da superclasse **Carta**.
5. Implemente o método **toString()** em **Lacaio**. Esta sobrescrita deve retornar uma descrição de todos os atributos do lacaio, reutilizando o método **toString()** da superclasse.
6. Sobrescreva o método **Carta.usar(Carta alvo)** em **Lacaio**. Na classe **Lacaio**, este método deve aplicar dano sobre um objeto alvo: deverá diminuir a *vidaAtual* da carta alvo pelo *ataque* do objeto que tem esse método invocado. Você vai precisar fazer um cast explícito do objeto **alvo** para o tipo **Lacaio** como **Lacaio aux = (Lacaio) alvo**; e aí poderá chamar os métodos de Lacaio a partir de **aux**.

## 2.4 A classe Magia

1. Faça **Magia** estender **Carta**. Remova todos os atributos repetidos (i.e., todos os atributos de **Lacaio** que já foram implementados por **Carta**).
2. Atributos privados de Magia: **String descrição**, uma descrição do que a magia faz.
3. Faça os métodos setters e getters destes atributos.
4. Crie os construtores de **Magia** de tal forma que estes reutilizem um ou mais construtores da superclasse **Carta**.
5. Implemente o método **toString()** em **Magia**. Esta sobrescrita deve retornar uma descrição de todos os atributos da magia, reutilizando o método **toString()** da superclasse.

## 3 Baralho com objetos do tipo Carta, Lacaio, e Magia

Vamos ilustrar um cenário onde herança faça sentido: refatore **Baralho** para que este contenha qualquer tipo de cartas.

## 4 Tarefas

Responda as questões a seguir sucintamente em um arquivo texto que deverá ser submetido juntamente com o código:

1. Crie uma nova classe **Main** que contenha o método **main**. Instancie um **Baralho** e adicione várias cartas de diferentes tipo à ele.
2. Faça a chamada do método para embaralhar o baralho e apresente na tela as informações de todas as cartas atualmente no baralho da primeira para a última.
3. É possível acessar o atributo nome diretamente da classe **Lacaio**? Explique.

## 5 Submissão

Submeta no google classroom a pasta do projeto comprimida como um arquivo zip, de tal forma que o arquivo se chame Lab2RAxxxxxx.zip (xxxxxx deve ser substituído pelo seu RA). A pasta deve ter sido preparada conforme especificado. Não se esqueça de incluir na pasta o arquivo respostas.txt.