



FUNDAMENTOS DE PROGRAMAÇÃO ORIENTADA A OBJETOS

---

# Praticando a persistência de objetos em uma aplicação com interação por eventos

ATIVIDADE SOMATIVA 2 – EM GRUPOS

---



## Sumário

APRESENTAÇÃO .....	2
QUESITOS A SER AVALIADOS .....	2
ORIENTAÇÕES GERAIS .....	3
ORIENTAÇÃO PARA IMPLEMENTAÇÃO DA INTERFACE GRÁFICA .....	4
SUGESTÕES DE DOMÍNIO DE APLICAÇÃO .....	6

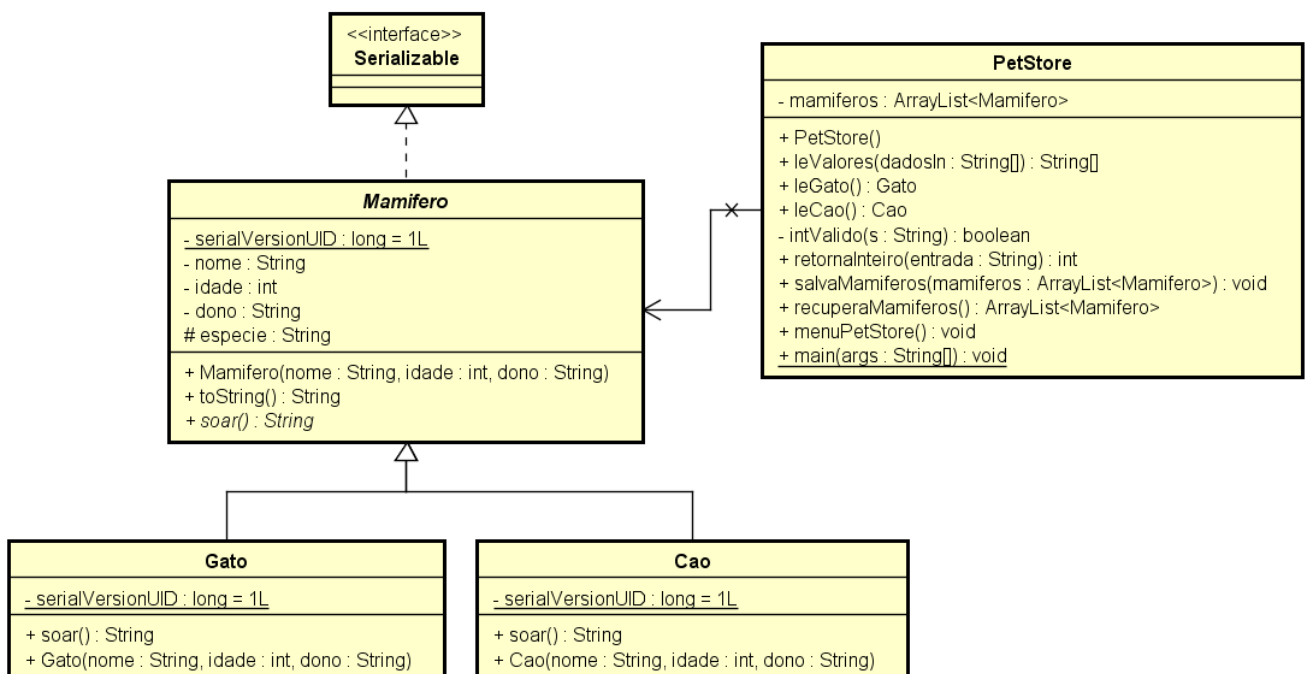


## APRESENTAÇÃO

Nesta atividade, **equipes de dois a três estudantes** deverão **construir uma aplicação** em que sejam empregados os conceitos de **persistência de objetos**, com utilização de **interface gráfica** orientada a evento, **herança**, **polimorfismo** e **tratamento de exceção**.

Para orientar o desenvolvimento, adaptaremos o último exercício disponibilizado na **Unidade 8 (Exercício 5 – Persistindo objetos com interface gráfica)**, que está resolvido e disponível no nosso material (**arquivo PetSore.ZIP**).

A seguir, estão representadas as classes da aplicação em um **diagrama de classes**, que deverá ser usado como **guia/orientação** para o trabalho a ser entregue.



## QUESITOS A SER AVALIADOS

1. **Modelagem de classes com herança**, realizada com o auxílio de ferramenta (PowerPoint ou Word, por exemplo).
  - a) Cada equipe deve definir sua **organização própria** de **classes com herança**, visando a aproveitar os recursos do **polimorfismo**.
  - b) Com um domínio de aplicação selecionado/definido, a equipe deve **adaptar o exemplo dado (PetStore) para o domínio de aplicação escolhido**.
  - c) **Sugestões de classes possíveis estão disponibilizadas no fim deste documento**.



2. Polimorfismo entre classes.

- d) Sobrescrita (*@override*) de métodos.
- e) Referências polimórficas.
- f) Controle de **repetição** usado em coleção de objetos (**ArrayList**).

- 3. Gravação e recuperação de objetos em arquivo.
- 4. Utilização da estrutura **switch** para controle de **menu de opções**.
- 5. Utilização de **tratamento de exceções** para manipulação de arquivos.

## ORIENTAÇÕES GERAIS

Com base no exemplo (arquivo PetStore.ZIP), as equipes deverão utilizar os recursos a seguir, como apresentado ou de forma equivalente, desde que todos os componentes da equipe demonstrem conhecimento desses recursos.

**Lista** (no exemplo da PetStore)

- Usar estrutura de **ArrayList** para manter em memória, **semelhante** à utilizada no exemplo com a coleção de objetos mamíferos de **PetStore**:

```
ArrayList<Mamifero> mamiferos = new ArrayList<Mamifero>();
```

**Menu** (no exemplo, para manipulação das operações da PetStore)

- Usar estrutura de controle **switch**

```
...
switch (opc1) {
    case 1:// Entrar dados
        ...
        break;
    case 2: // Exibir dados
        ...
        break;
    case 9:
        JOptionPane.showMessageDialog
        (null,"Fim PET Store");
        break;
}
...
```

**Sobrescrita** (*@override*)

- Usar no método **toString()**, aproveitando o método da classe-mãe, por meio da chamada **super**.
- Usar em métodos **construtores** (encadeamento de construtores).

**Persistência de objetos:** observem os detalhes de implementação usados no exemplo.

- Objetos a serem persistidos (armazenados em arquivo) devem **implementar** a interface. **java.io.Serializable**



- Gravação (void salvaProdutos (ArrayList<Mamifero> mamiferos):

```
outputStream = new ObjectOutputStream  
                (new FileOutputStream(nomeArquivo));
```

- Recuperação (ArrayList<Mamifero> recuperaProdutos ()):

```
inputStream = new ObjectInputStream  
                (new FileInputStream(nomeArquivo));
```

### Itens esperados na aplicação

1. Cada equipe deverá ter seu **conjunto próprio de objetos (domínio de aplicação)**, que deverão ser persistidos e recuperados, conforme escolha do usuário em menu de opções.
2. O conjunto de objetos deverá trabalhar a **generalização com hierarquia**, para aproveitamento das facilidades do **polimorfismo** (referência polimórfica em coleção de objetos, como **ArrayList**, para manter coleções de objetos com as mesmas relações hierárquicas).
3. A aplicação deverá utilizar **interface gráfica**, conforme exemplificado (**arquivo PetSore.ZIP**).
4. A aplicação deverá ter **tratamento de exceção** para acesso a arquivos e para tratamento de demais erros em potencial.
5. A aplicação deverá ter tratamento de **opções para o usuário via menu**.
6. Utilizar construtores adequadamente: **reutilizar o construtor da classe-pai**, sempre que possível.
7. Utilizar **getter** e **setter** para prover encapsulamento, sempre que possível.
8. Utilizar o método **toString()** para apresentar objetos ao usuário.

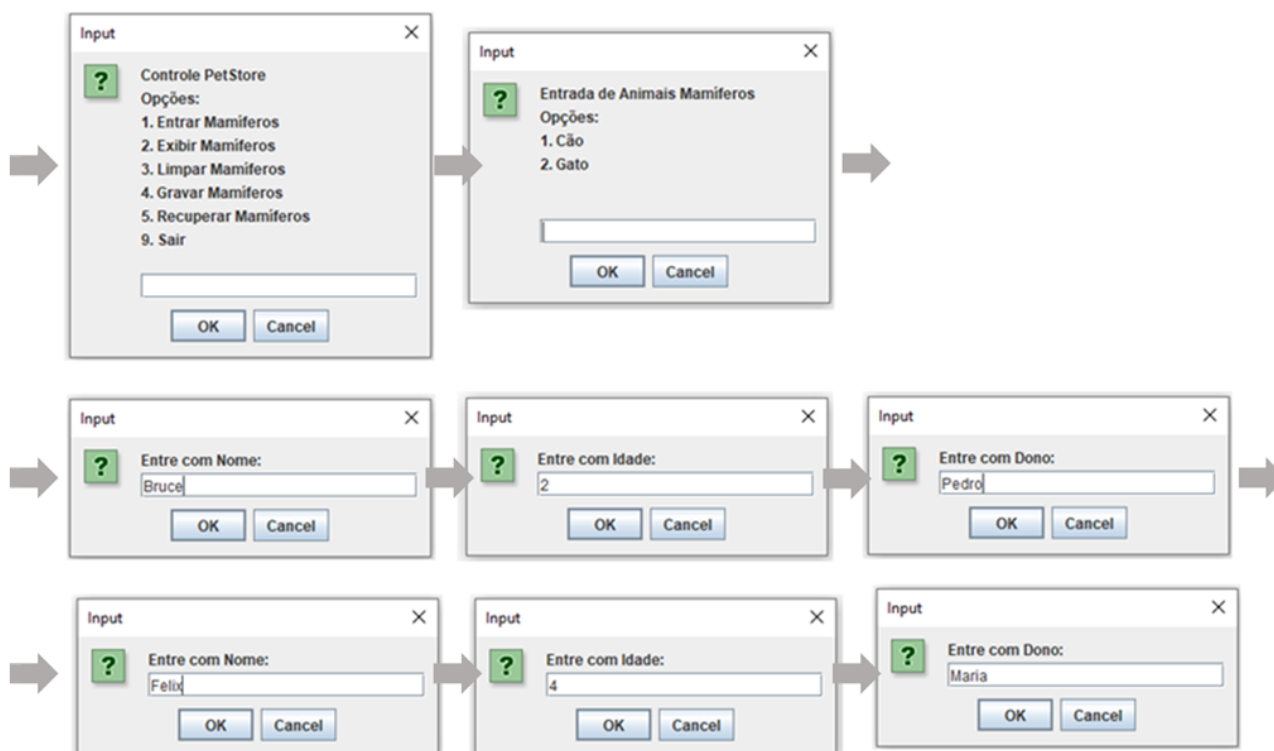
## ORIENTAÇÃO PARA IMPLEMENTAÇÃO DA INTERFACE GRÁFICA

---

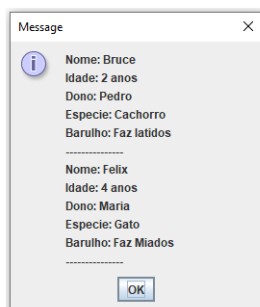
Seguir o modelo passado no exemplo da PetStore.

Sequência de apresentação das caixas de diálogo do programa – **adaptar** o domínio de aplicação escolhido, conforme demonstrado nas telas do exemplo, a seguir.

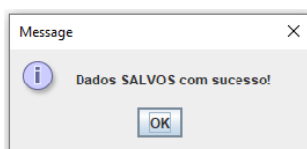
- a)** Entrar os dados de um **cão** e de um **gato**.



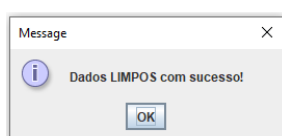
b) Exibir a lista de mamíferos.



c) Gravar os objetos criados em arquivo.

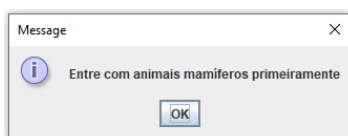


d) Limpar da memória (ArrayList) os objetos recém-criados.

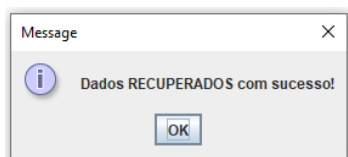




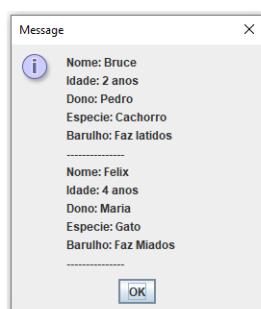
- e) Exibir novamente a lista de mamíferos.



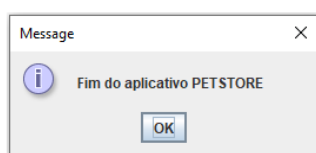
- f) Recuperar os objetos persistidos.



- g) Exibir novamente a lista de mamíferos.



- h) Sair do programa.



## SUGESTÕES DE DOMÍNIO DE APLICAÇÃO

Seguem sugestões para alteração do domínio de aplicação, para que as equipes alterem o exemplo dado. Caso prefiram, poderão criar e utilizar seu próprio conjunto de objetos, desde que validado com o professor.

**Importante:** garanta que as subclasses tenham **mais um ou dois atributos**, além dos recebidos da herança.

### Supermercado

- Produto (código, nome, fornecedor, categoria: limpeza, bebida ou laticínio)
  - Limpeza (tipo de limpeza: sabão, desinfetante, detergente...)
  - Bebida (tipo de bebida: refrigerante, vinho, suco...)
  - Laticínio (tipo de laticínio: queijos, leites, iogurtes...)



### Biblioteca

- Livro (nome, ISBN, autor, resenha)
  - Livros infantis (brinde: boneco, lápis de colorir, giz de cera...)
  - Livros de culinária (tipo de culinária: regional, internacional, doces, carnes...)
  - Guias de viagem (local: nome de país, nome de região em um país, parques...)

### Clínica veterinária

- Animal (nome, espécie, dono)
  - Cavalo (alimentação: ração, aveia, alfafa...; cuidados próprios: corrida, trote, cascos...)
  - Cachorro (alimentação: ração, carnes...; cuidados próprios: tosa, banho...)
  - Pássaro (alimentação: alpiste, frutas...; cuidados próprios: abrigo, estímulos...)

### Exames clínicos

- Exame (convênio, nome do médico, nome do paciente)
  - Sangue (preparação própria: jejum 6h, jejum 8h, remédios e bebidas proibidos...)
  - Raio X (preparação própria: jejum, laxantes...)
  - Ecografia (preparação própria: jejum, bexiga cheia...)

### Grupo escolar

- Estudante (matrícula, nome, responsável)
  - Pré-escolar (cuidados: alergias, remédios...)
  - Ensino fundamental (atividades de contraturno: natação, judô, ginástica...)
  - Ensino médio (oficinas profissionalizantes: computação, marcenaria, contabilidade...)

### Frota motorizada

- Veículo (marca, modelo, ano, quilometragem, placa)
  - Automóvel (motorização: motor 1.0, motor, motor 1.6...)
  - Caminhão (carga máxima: 16 toneladas, 23 toneladas...)
  - Ônibus (assentos: 20, 40...)

### Delegação

- Atleta (nome, número)
  - Saltador (altura)
  - Corredor (velocidade)
  - Nadador (estilo)

### Loja de eletrônicos

- Equipamento (nome, marca, modelo, tamanho da tela)
  - *Smartphone* (quantidade de *chips* de operadora, tamanho da tela)





## ATIVIDADE SOMATIVA

---

- *Smartwatch* (tipo de pulseira: couro, metal...)
- *Notebook* (processador, memória RAM, disco, tamanho da tela)



PROFESSOR-AUTOR

Cristina Verçosa Perez Barrios de Souza  
Vinícius Godoy de Mendonça



**PUCPR**  
GRUPO MARISTA