

POO – Atividade de Implementação 03 – Parte 01

1. Desenvolva uma hierarquia de classes com o objetivo de representar uma `Fazenda` que possui um nome, uma certa dimensão em metros quadrados, situada em uma determinada cidade e estado. Além disso, a `Fazenda` possui inúmeros `Animais`, que são criados para serem abatidos e terem a sua carne vendida, tais como `Porcos`, `Bois`, `Galinhas` e `Patos`. Todos os animais, independentemente de sua espécie, possuem registro de idade em meses ou em dias (dependendo do tipo de animal), sexo, peso em quilos e valor do quilo de sua carne (**Observação.** Sabe-se que peças de carne de diferentes partes dos animais citados possuem diferentes preços por quilo, mas para simplificar considere a existência de um único preço por quilo). Todos os animais são capazes de se movimentar, emitir algum tipo de som e se alimentar, cada espécie com suas características próprias. Neste contexto, implemente sua hierarquia de classes, considerando os seguintes requisitos mínimos:

- Faça uso adequado do conceito de **herança** e de **polimorfismo**, implementando pelo menos uma classe abstrata e funções virtuais;
- Implemente os métodos `get` e `set` que considerar necessários para os atributos das classes;
- Faça uso de um ou mais atributos estáticos para contar o número de cada espécie na `Fazenda`;
- Deve existir um método que retorne o preço estimado de venda da carne do abate de cada um dos animais. O preço estimado deve considerar o peso do animal e o preço por quilo;
- Implemente o método `print` que deve exibir o estado do `Animal` (valor de seus atributos), além de informar o tipo de animal do objeto;
- Todos os animais devem ser capazes de emitir som. Implemente métodos que exibem em tela uma mensagem com a onomatopeia dos sons emitidos pelos animais (<https://www.dicio.com.br/sons-de-animais/>);
- Todos os animais devem ser capazes de se movimentar/deslocar. Todas as vezes que um animal se movimentar/deslocar ele perde um certo percentual de seu peso: `Bois` (0,4% de seu peso), `Porcos` (0,3% de seu peso), `Patos` (0,2% de seu peso) e `Galinhas` (0,1% de seu peso);
- Todos os animais devem ser capazes de se alimentar. Todas as vezes que um animal se alimentar ele deve ter o seu peso acrescido em uma certa quantidade em quilos. `Bois` – 5Kg, `Porcos` – 2Kg, `Patos` – 0,5Kg e `Galinhas` – 0,2Kg);
- As classes devem considerar um limite máximo de peso para cada um dos tipos de animais. `Bois` – 600Kg, `Porcos` – 120 Kg, `Patos` – 4,5Kg e `Galinhas` – 1,8Kg;
- A `Fazenda` pode conter inúmeros `Animais`, que devem ser armazenados em um atributo do tipo `vector <Animais *>`;

- Na `Fazenda`, deve ser possível inserir um novo `Animal`, assim como remover algum de seus `Animais`. Neste último caso, deve ser informado um índice do atributo do tipo `vector <Animais *>`;
- Na `Fazenda`, devem existir métodos que retornam o número de `Animais` de cada espécie. O método `print` exibe todos os atributos da `Fazenda`, o número total de animais e o número de `Animais` de cada espécie;
- Deve ser implementado um método que retorne o somatório do preço estimado de todos os `Animais` da `Fazenda`;
- Deve existir um método que retorna o preço estimado de um determinado `Animal` da `Fazenda`, para isso deve ser informado um índice do atributo do tipo `vector <Animais *>`;
- Outros métodos podem ser implementados, caso queira tornar a sua classe `Fazenda` ainda mais funcional.

OBSERVAÇÃO: Diversas características desta Atividade são fictícias, não traduzindo a realidade de uma fazenda e do abate e venda de carne de animais. Portanto, inúmeros quesitos foram simplificados.