

## Lista de Exercícios 5

1. Crie uma classe Pessoa, que possui um nome (String), nascimento (String), rg (long) e cpf (long). Nesta mesma classe, implemente um método construtor que receba por parâmetro todos os atributos de pessoa. Também implemente um método que exiba as informações de uma pessoa.  
Crie uma classe Aluno que é uma Pessoa. O aluno deverá ter todos os atributos de uma pessoa e também um número de matrícula (int). Implemente um método construtor em aluno que receba todos os atributos do aluno. Ainda na classe aluno, implemente um método que imprima todas as informações do aluno.  
Crie uma classe AlunoTeste onde (i) deverá ser criado um objeto do tipo pessoa, com suas informações, (ii) invoque o método para imprimir as informações do objeto pessoa, (iii) deverá ser criado um objeto aluno, com suas informações e (iv) invoque o método para imprimir as informações do objeto aluno.
2. Crie a classe Imovel, que possui um endereço e um valor base.  
Crie uma classe ImovelNovo, que herda Imovel e possui um adicional no preço. Crie métodos de acesso e impressão dos seus dados.  
Crie uma classe ImovelVelho, que herda imóvel e possui um desconto no preço. Crie métodos de acesso e impressão dos seus dados.  
Crie uma classe Testaimovel. Utilize essa classe para instanciar imóveis do tipo ImovelNovo e ImovelVelho e invocar os métodos que imprimem os dados de cada imóvel.
3. Crie uma classe chamada Ingresso que possui um valor em reais e o tipo do ingresso. Declare um método construtor que receba o tipo do ingresso e o valor. Declare o método imprimirDadosIngresso(), que imprime um relatório contendo o tipo do ingresso.  
Crie uma classe IngressoNormal, que herda Ingresso e possui um método construtor recebendo o tipo e o valor de ingresso. Além disso, declare um método chamado imprimirDadosIngresso(), que invoca o método imprimirDadosIngresso() da classe pai e imprimir o preço do ingresso normal.  
Crie uma classe IngressoVip, que herda Ingresso e possui um valor adicional e localização (camarote inferior, camarote superior, por exemplo). Crie um método construtor que recebe o tipo de ingresso, valor, valorAdicional e localização. O tipo de ingresso e o valor devem ser enviados ao construtor da classe pai, os demais atributos devem ser atribuídos no próprio método construtor. Além disso, declare um método privado calcularValorIngressoVip() que irá retornar o valor do ingresso VIP, este método somará o valor do ingresso com o valor adicional. Declare um método chamado imprimirDadosIngresso(), que invocará o método imprimirDadosIngresso() da classe pai, além disso, imprimirá a localização e o valor do ingresso vip.  
Crie as classes Camarote Superior e Camarote Inferior, ambas as classes devem herdar de IngressoVip. Além disso, as duas classes devem possuir um método construtor que receba o tipo de

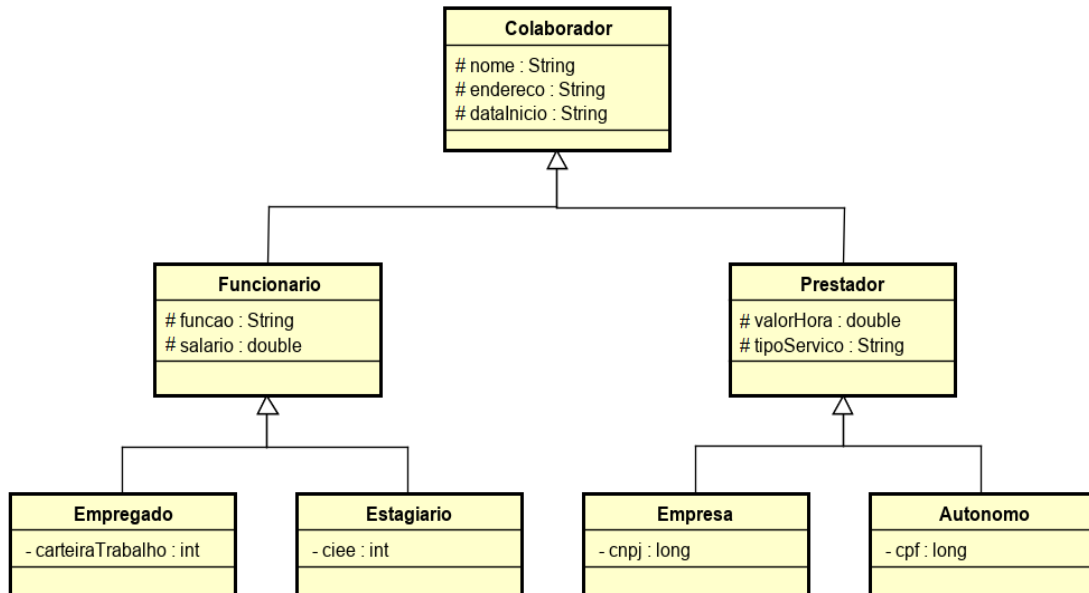
ingresso, valor do ingresso, valor adicional e localização. Ainda no método construtor, todos os argumentos devem ser passados ao método construtor da classe pai.

Crie uma classe de testes. Esta classe deve instanciar um objeto do tipo Ingresso Normal, passando o tipo de ingresso e o valor por parâmetro; Camarote Inferior e Camarote Superior, passando por parâmetro o tipo de ingresso, valor, valor adicional e localização. Para cada objeto instanciado, invoque o método imprimirDadosIngresso() para verificar se os dados de cada objeto estão corretos.

#### 4. Desenvolva:

- Construa a classe **Data** com os seguintes métodos:
  - Construtores recebendo todos os parâmetros (dia, mês e ano).
  - Construtor recebendo como parâmetros valor para mês e ano, e então chamar o construtor anterior passando o dia com o valor padrão: 1.
  - Construtor recebendo somente o valor do dia.
    - Chamar o construtor completo, passando por parâmetro o valor do dia recebido, além dos valores padrão: 01 para mês e 2019 para ano.
  - Construtor não recebendo parâmetros.
    - Chamar o construtor completo, passando os valores:
      - Dia = 01;
      - Mês = 01;
      - Ano = 2019.
  - Método **toString** que retorna uma **String** com os valores de Data, no seguinte formato:
    - dia/mês/ano.
- Construa uma classe chamada **Feriado** que deve herdar da classe **Data**.
  - Variável-membro chamada "descricao" que deve ser do tipo String;
  - Construtor que não recebe parâmetros
    - Chama o construtor-pai correspondente
    - Inicializa descricao com "Feriado"
  - Construtor recebendo uma String para a descrição
    - Chama construtor pai-correspondente;
    - Atribui valor recebido à variável-membro descricao
  - Construtor recebendo valores para dia, mês, ano e descrição
    - Chama construtor pai-correspondente
    - Atribui valor recebido à variável-membro descricao
  - Método toString que retorna uma String com os valores de Data no seguinte formato:
    - dia/mês/ano – descrição.

5. Seguindo a modelagem, desenvolva o exemplo:



- Crie todos os métodos getters, setters e construtores.
- Para cada classe, crie métodos `toString` que retornam uma variável do tipo `String`, a qual concatena todas os atributos da classe.
- Crie métodos para imprimir os atributos da classe.
- Crie uma classe de teste que irá instanciar todos os tipos de subclasses (empregado, estagiário, empresa e autonomo).