

Curso Técnico em Desenvolvimento de Sistemas

Disciplina: Orientação a Objetos

Carga Horária: 120 horas aula

Período: 2º ano

Data: 01/10/2022

Revisão Classes, Atributos, Métodos e Herança

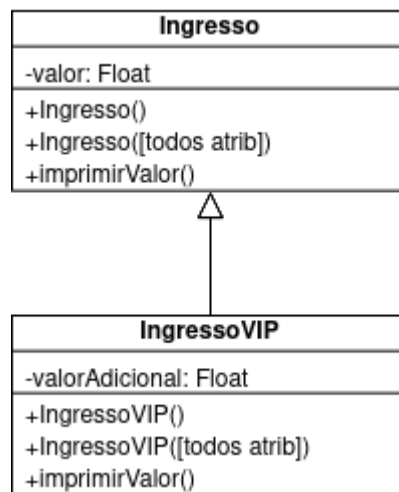
1. De acordo com o diagrama de classes abaixo, implemente em Java uma classe para representar os funcionários de uma empresa. A classe **Funcionario** deve possuir seus atributos encapsulados e acessíveis de forma pública através de métodos de GETs e SETs.

Funcionario
-primeiroNome: String
-sobrenome: String
-salarioMensal: Float
+Funcionario([todos atrib])
+imprimirSalarioAnual()

Após implementar a classe **Funcionario**, inclusive com o método para imprimir salário anual, crie outra classe de teste pra utilizá-la. Instancie dois objetos e exiba o salário anual de cada instância no formato: João da Silva ganha R\$ XXX.XX anualmente.

Na sequência, crie um novo método na classe **Funcionario** para conceder aumento, sendo: +concederAumento(percentual: Integer). Nele, atualize o salário mensal do funcionário. Por fim, faça a chamada deste novo método na classe de teste e exiba novamente o salário mensal dos dois objetos previamente criados.

2. Crie um programa em Java que implemente as classes Ingresso e IngressoVIP conforme o modelo abaixo (não esqueça do encapsulamento). Perceba que existe uma relação de herança entre as classes.



O método `imprimirValor()` deve imprimir o valor do ingresso. Para o caso da classe **IngressoVIP**, o valor do ingresso é o valor da superclasse (pai) mais o valor adicional do **IngressoVIP**.

Após, faça:

- Crie uma classe para testar a criação de objetos, imprimindo os valores de cada tipo de ingresso.
- Crie uma nova classe denominada **IngressoCamarote** (filha de **IngressoVIP**), sendo que a mesma também deve ter um atributo `valorAdicional` e um método `imprimirValor()`. O valor do **IngressoCamarote** deve considerar o valor do **IngressoVIP** mais o seu valor adicional.