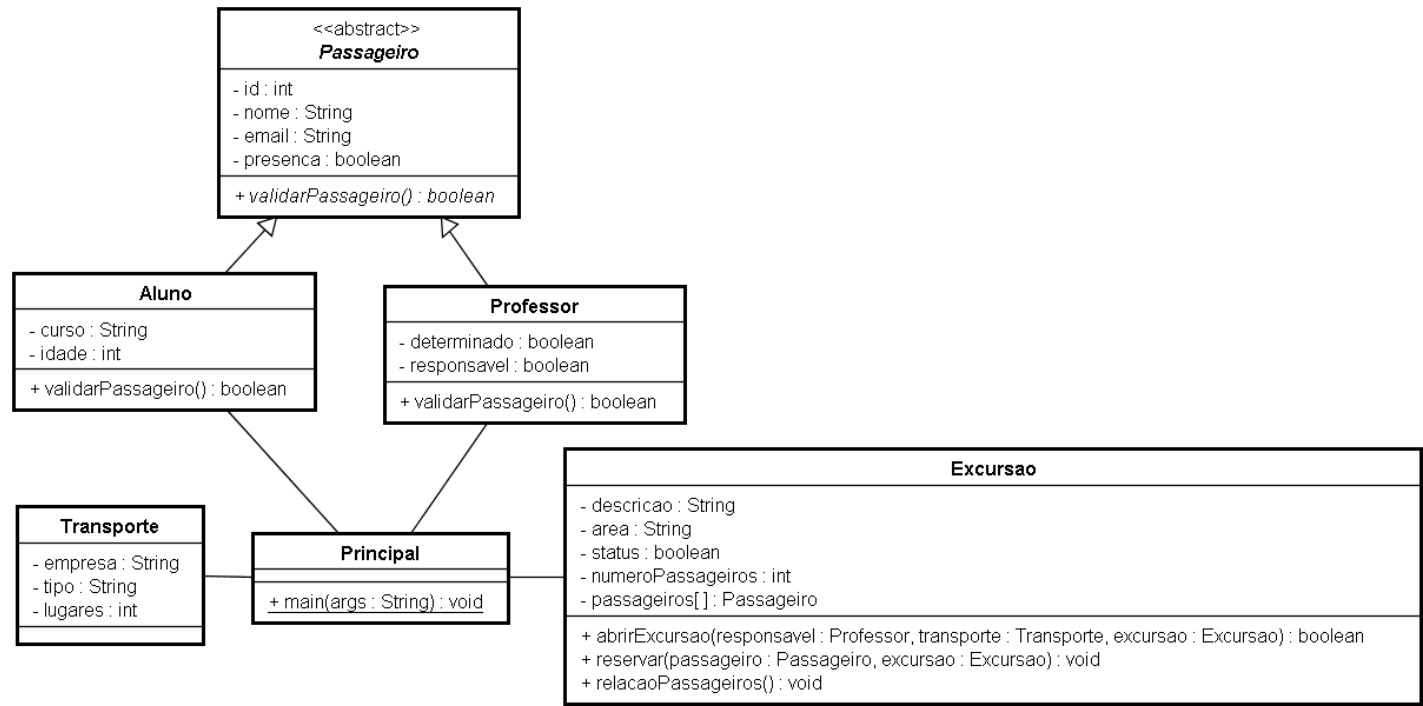


ATIVIDADE AVALIATIVA

Conteúdo: Estrutura de classe, atributo, método, construtores, getters, setters, assinaturas de métodos, passagem de parâmetros, retorno de valores, sobrescrita, classe e método abstrato, array, polimorfismo e lógica de programação orientada a objetos.

Projeto: prjTransporteEscolar



Classe abstrata: Passageiro

Classe abstrata: Passageiro	
Métodos	validarPassageiro: Método abstrato

Subclasse: Professor

Subclasse: Professor	
Métodos	validarPassageiro: Sobrescrita do método que verifica se o professor é o responsável pela excursão (se o atributo responsável é true)

Subclasse: Aluno

Subclasse: Aluno	
Métodos	validarPassageiro: Sobrescrita do método que verifica se o aluno é maior de idade (se o atributo idade é igual ou superior a 18)

Classe: Excursao	
Métodos	<p>abrirExcursao: Recebe por parâmetro um objeto do tipo Professor, um objeto do tipo Transporte e um objeto do tipo Excursao.</p> <p>Verifica se o professor é responsável pela excursão (e portando autorizado a abrir uma excursão), se for, muda o status da excursão para true e define o tamanho do vetor passageiros de acordo com a quantidade de lugares definido no transporte.</p>
	<p>reservar: Recebe por parâmetro um objeto do tipo Passageiro e um objeto do tipo Excursao.</p> <p>Faz a validação do passageiro através do método validarPassageiro, verifica se a excursão esta em aberto (através do atributo status do objeto excursão), altera o atributo presença para true e armazena o objeto passageiro no vetor passageiros.</p> <p>Se a excursão não estiver em aberto apresentar uma mensagem de “Excursão fechada”, se o passageiro não for validado apresentar mensagem de “Passageiro não autorizado”.</p>
	<p>relacaoPassageiros: Apresenta uma relação dos passageiros confirmados (percorrendo o vetor passageiros).</p>

Classe: Principal	
Método main	<ul style="list-style-type: none"> • Instancia 1 objeto do tipo Excursao chamado excursão (inicializando seus atributos pelo construtor).
	<ul style="list-style-type: none"> • Instanciar 1 objeto do tipo Transporte chamado transporte (inicializando seus atributos pelo construtor).
	<ul style="list-style-type: none"> • Instanciar 3 objetos do tipo Professor chamados interessado1, interessado2 e interessado3 (inicializando seus atributos pelo construtor).
	<ul style="list-style-type: none"> • Instanciar 10 objetos do tipo aluno chamados interessado4, interessado5, interessado6, interessado7, interessado8, interessado9, interessado10, interessado11, interessado12 e interessado13 (inicializando seus atributos pelo construtor).
	<ul style="list-style-type: none"> • Apresentar um menu com as opções de excursão.
	<ul style="list-style-type: none"> • Apresentar outro menu de professores ou de alunos interessados de acordo com a opção do menu anterior.
	<ul style="list-style-type: none"> • Realiza as chamadas dos métodos para: <ul style="list-style-type: none"> ○ Abrir uma excursão ○ Reservar um lugar ○ Listar passageiros

1. Criar as classes Passageiro, Aluno, Professor e Transporte, implementando suas estruturas internas e estabelecendo a herança entre elas e suas respectivas considerações.
2. Definir a classe Passageiro e seu método validarPassageiro como abstratos e sobrescrevê-los nas classes Aluno e Professor de acordo com as especificações do enunciado.
3. Criar a classe Excursao definindo sua estrutura interna e a assinatura de seus métodos.
Observações:
 - a. Para a declaração de um atributo como vetor (sem definir sua dimensão) basta colocar colchetes no tipo, no caso: Passageiro[]
 - b. Para inicializar o vetor no construtor vazio utilize null
4. Implemente o método abrirExcursao (Sem o tratamento do vetor passageiros).
5. Implemente o método reservar (Sem o tratamento do vetor passageiros).
6. Implemente o tratamento do vetor passageiros
7. Implemente o método relacaoPassageiros