

INFORMAÇÕES SOBRE A AVALIAÇÃO

AV2	AVALIAÇÃO - 20 PONTOS - Prática
------------	--

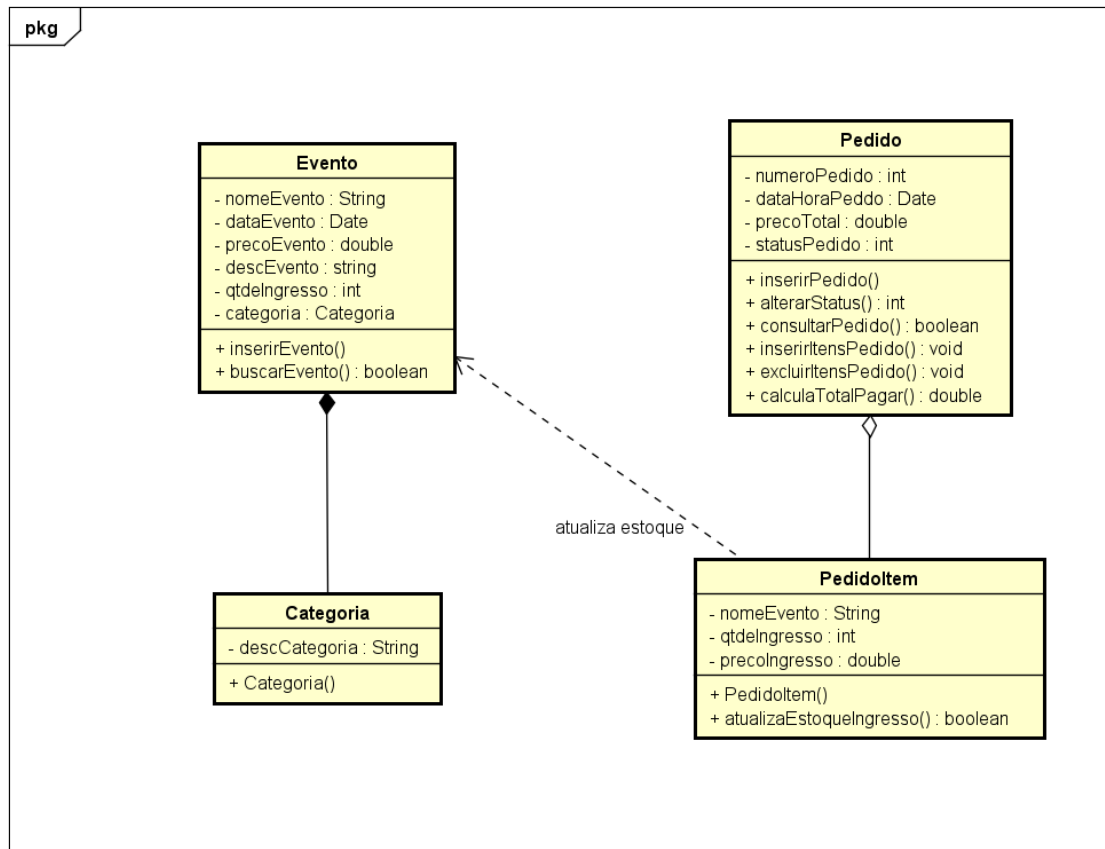
INFORMAÇÕES DOCENTE

CURSO: ADS/SI/ C. Comp.	DISCIPLINA: Programação Orienta a Objetos	TURNIO	MANHÃ	TARDE	NOITE	PERÍODO/ SALA:
					X	
PROFESSOR (A): Michelle Hanne Soares de Andrade						

INFORMAÇÕES DISCENTE

ALUNO(A):	RA:
DATA:	NOTA:
INSTRUÇÕES: <ol style="list-style-type: none"> 1) Respostas deverão ser NECESSARIAMENTE registradas à caneta azul ou preta, de corpo transparente. Respostas à lápis e com caneta "escreve e apaga" não serão corrigidas. 2) Caso seja identificada rasura ou duplo preenchimento da questão objetiva, a questão será anulada. 3) Nas questões discursivas, quando aplicável/necessário, deve ser apresentada a memória dos cálculos (ou o raciocínio desenvolvido). 4) Desenvolva a prova discursiva com letra legível. Caso a caligrafia impossibilite o entendimento do conteúdo, o mesmo não será corrigido. 5) Não será permitido o uso de celulares, relógios digitais ou de qualquer outro aparelho eletrônico durante a realização da prova, exceto o uso de calculadora portátil no caso de disciplinas que a autorizam. 6) Será eliminado do processo avaliativo, com atribuição de nota 0 (zero), o aluno que, durante a prova, utilizar meios ilícitos, como por exemplo: comunicação com outros alunos, consulta a qualquer tipo de material extraprova e uso de aparelhos eletrônicos não autorizados. Também não poderá haver empréstimo de qualquer material. Boa Prova!	

Questão 1) Analise cuidadosamente o Diagrama de Classe abaixo:



a) Implementar as classes relacionadas a um exemplo de **vendaIngressos** com os respectivos modificadores, atributos, relacionamento e métodos, **como está no Diagrama acima**.

Os métodos serão detalhados nas questões posteriores, bem como o Main.

b) Sobre a Classe Evento:

- O método **inserirEvento** será o **construtor** e deverá receber como parâmetro todos os atributos desta classe.
- Criar os métodos *getters* e *setters* para todos os atributos desta classe.
- O método **buscarEvento** deverá receber um objeto do tipo Evento e fazer a comparação com o atributo nomeEvento, se for igual deve retornar verdadeiro, caso contrário falso.

c) Sobre a Classe Categoria:

- Terá o método Construtor com o mesmo nome da classe e os métodos *getters* e *setters* dos atributos.

d) Sobre a classe **Pedido**:

- O método **inserirPedido** será o **construtor** e deverá receber como parâmetro todos os atributos desta classe.
- O método **alterarStatus** deverá receber um parâmetro do tipo `int` e alterar o atributo `statusPedido`.
- O método **consultarPedido** deverá receber um objeto do tipo `Pedido` e fazer a comparação com o atributo `numeroPedido` ser for igual retornar verdadeiro, caso contrário falso.
- O método **inserirItensPedido** deverá receber um objeto do tipo `PedidosItem` e acrescentar em um `ArrayList`. Para isso criar um `ArrayList` do tipo `PedidosItem`.
- O método **excluirItensPedido** deverá receber um objeto do tipo `PedidosItem` e excluí-lo do `ArrayList`, se o item existir.
- O método **calculaTotalPagar** deverá calcular o total do pedido, referente a todos os itens contidos no `ArrayList` de `PedidosItem`.

e) Sobre a classe **PedidosItem**:

- O método **PedidosItem** será o **construtor** e deverá receber como parâmetro todos os atributos desta classe.
- Criar os métodos *getters* e *setters* para todos os atributos desta classe.
- O método **atualizaEstoqueIngresso** deverá atualizar a quantidade de ingressos do evento.

Questão 2) Criar um `Main()` para manipular as classes e assim testar se o métodos estão de acordo com o esperado.

ArrayList de Evento: inserir 5 eventos distintos distintos.

ArrayList de Pedido: inserir 2 pedidos com 3 respectivos itens cada. O status padrão é 1, a data hora do pedido é a data e hora atual, o valor total é zero, pois será calculado posteriormente.

Deverão ser realizados as seguintes validações:

- Ao inserir um **PedidosItem** este deverá **existir como Evento**.
- A `qtdeIngresso` deverá ser atualizada após a **inserção do Item**.
- O **status** do pedido deverá ser **atualizado para 2**, após a **inserção dos itens e atualização do estoque**.
- Para **cada pedido** deverá ser calculado o **valor total a pagar**.

Não existe uma resposta única, cada aluno deverá criar o seu `Main()`.