

NOME: Giovana Kerche Bonás **RA:** 216832

NOME: Kaio Ken-Ichi Carvalho Takuma **RA:** 219510

OBSERVAÇÕES GERAIS

1. Para unificar a forma de nomenclatura das variáveis vou Usar como padrão snake case . todos os termos em maiúsculas separado por '_ '.
2. Contudo Usaremos o snake case a princípio somente para variáveis, como forma de melhor didática do código.
3. Padrão de classes:
 - 3.1. Constantes Static
 - 3.2. Atributos
 - 3.3. Construtor
 - 3.4. Métodos acessores
 - 3.5. Funções auxiliar
4. Modelo seguido para implementação do projeto se encontra logo abaixo .

1 Estudo de Caso: Cinema

1.1 Descrição do Problema

Nosso objetivo foi desenvolver uma aplicação para um cinema que controla a emissão de ingressos. Permitindo ao cliente realizar a compra através do cartão ou do dinheiro para dois filmes, sendo para cada um deles duas sessões do cinema reservadas.

Os caixas automáticos operam em dois modos distintos: supervisor e cliente. Sendo que no modo cliente, o cliente consegue efetuar a compra, a classe que é mais responsável pela compra é o controladorIngresso.

1.2 Análise e Projeto do Sistema de Cinema

No projeto do sistema definimos as seguintes classes:

Sessao

Classe primogênita do sistema, responsável por armazenar dados das sessões, verificando as sessões disponíveis, sessões ocupadas, número de sessões como também resetar a sessão. Além disso cuida da estruturação da sala, sendo que para fator testes usamos um vetor de tamanho 3 para os assentos de cada sala.

Nesse sentido dentre esse sentido utilizamos alguns métodos que atendem as necessidades da classe Sessão.

O método Imprime Modelo Sessão(), nesse método devemos efetuar um ciclo for que percorra todos os assentos do cinema e imprima o esquema atual do

cinema, mostrando os assentos ocupados representados por meio do número “1” e os não ocupados representados por meio do número “0”.

O método `liberarSessao(int pwd)`, esse método recebe como parâmetro a senha do administrador , logo em seguida ele percorre por meio de um ciclo for as salas das respectivas sessões liberando as.

O método `adicionaLugar(int pwd, int poscoluna)`,esse método verifica se a senha digitada é compatível com a do administrador caso sim ela adiciona o assento desejado do usuário na sala da respectiva sessão.

O método `validarSenha(int pwd)`,esse método simplesmente faz uma comparação, verificando se a senha digitada é compatível com a do administrador , retornando “1” para sim e “0” para não.



Ingresso

Classe responsável ,por controlar os itens do objeto sessão , além disso a classe ingresso apresenta os dados referentes a venda do ingresso como os dados do usuários e as opções de escolha(filme,sala e sessões).



ControladorIngresso

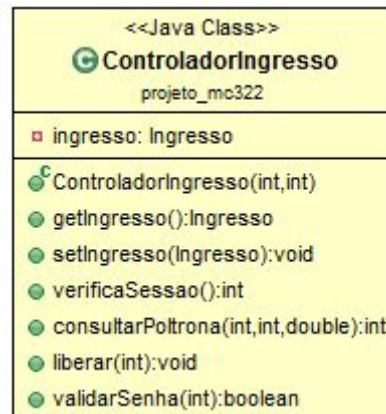
Um objeto da classe ControladorIngresso encapsula a política do sistema auxiliar de vendas de um cinema, em relação a todos os dados referentes ao objeto ingresso. Um objeto controlador recebe as requisições referente a escolha do administrador e coordena as requisições das seguintes operações.

Nesse sentido utilizamos alguns métodos que atendem as necessidades da classe ControladorIngresso.

O método `verificaSessao()`, percorre um ciclo verificando se para aquela sessão a sala apresenta o número de assentos disponíveis maior ou igual a um, caso sim ela retorna "1" e informa o atendente por meio de uma mensagem.

O método `consultarPoltrona(int pwd, int posLinha, double sessao)`, é mais específico, por meio da senha do administrador ele concede permissão, logo em seguida ele recebe os parâmetros da linha e sessão específica e verifica se a posição requisitada é igual a "1" se sim retorna que o assento está ocupado.

O método `liberar(int pwd)`, esse método utiliza a função `liberarSessao(int pwd)` do objeto sessão para liberar as sessões.



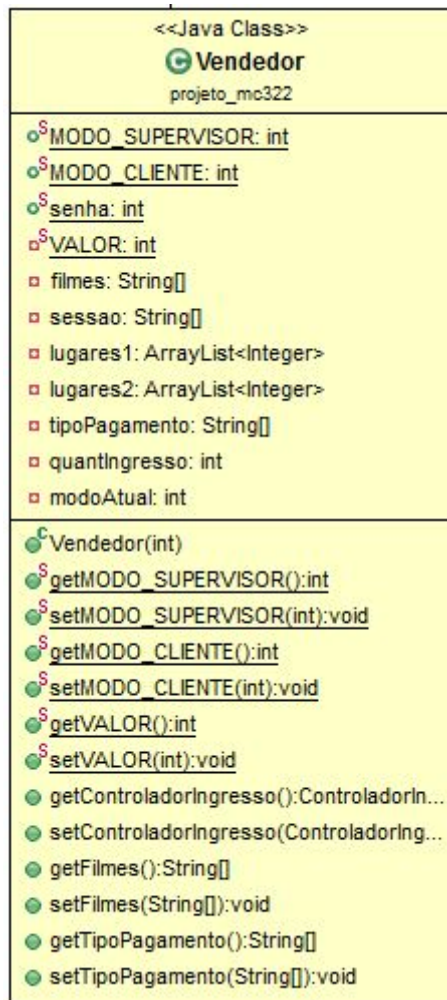
Vendedor

A classe `Vendedor`, classe responsável por interagir com o usuário de forma apropriada e receber os dados do mesmo, para que assim haja um banco de dados que possa ser passado para `CadastroCompras`.

Nesse sentido utilizamos alguns métodos que atendem as necessidades da classe `Vendedor`.

O método `iniciarOperacao()` é responsável por meio de um conjunto de perguntas e respostas, apresentar as propostas do cinema e receber os dados referente a instâncias do usuário, como a identificação, número de ingressos, sessão escolhida, tipo de ingresso e poltrona requisitada.

O método `alternar Modo(int senhaSupervisor)` simplesmente verifica se a senha do administrador está correta e lhe dar permissão de acesso.



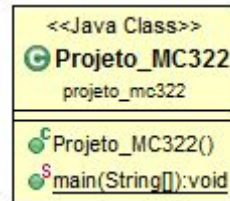
CadastroCompras

Classe responsável por inserir os dados dos clientes no objeto Ingresso como se fosse uma espécie de banco de dados.



Projeto Mc322

Classe estática responsável por operar os sistemas através do objeto CadastroCompras.



5. Funcionamento do programa:

Tela 1 adição de uma pessoa:

```

----- Bem Vindo Ao Cinema Kinoplex -----

Tudo bem? Serei Seu / Sua Atendente , qual seu nome? Giovana
Quantos ingressos deseja comprar? 2
Seja Bem vindo Giovana escolha uma das opcoes de filmes:
1 --> The Joker
2 --> Avengers
Opção de filme: 1
Voce escolheu o filme The Joker
Para o filme The Joker temos um total de 2 Sessoes
1 --> 14:00
2 --> 20:00
Digite a sessao de sua escolha: 1
Qual tipo de ingresso voce necessita? (meia ou inteira)
meia
Voce escolheu o tipo de ingresso meia
1 2 3
Qual poltrona deseja sentar? 1
  
```

Tela adição outra pessoa:

```

----- Com mais gente é mais divertido -----

Como é o nome do seu (sua) acompanhante? Mariana
Qual tipo de ingresso voce necessita? (meia ou inteira)
inteira
Poltronas:
2 3 Qual poltrona deseja sentar?
2
Formas de pagamento
1 --> Credito
2 --> Debito
3 --> Dinheiro
Qual a forma de pagamento?
3
Dados do boleto:
Valor: 21
Código de barras do boleto:
733919363
  
```

Tela mostra compra:


```

-----Compra confirmada-----
Agradecemos a confiança em nosso cinema
Dados da sua compra:
Ingresso 1
Filme escolhido: The Joker
Nome: Giovana
Sessão escolhida: 1
Poltrona escolhida: 01
Tipo de ingresso:meia
Valor: 21

```

Tela compra acima do valor de lugares disponíveis:

```

----- Bem Vindo Ao Cinema Kinoplex -----

Tudo bem? Serei Seu / Sua Atendente , qual seu nome? Kaio
Quantos ingressos deseja comprar? 8
Seja Bem vindo Kaio escolha uma das opcoes de filmes:
1 --> The Joker
2 --> Avengers
Opção de filme: 1
Voce escolheu o filme The Joker
Para o filme The Joker temos um total de 2 Sessoes
1 --> 14:00
2 --> 20:00
Digite a sessao de sua escolha: 1
Número de lugares disponíveis insuficientes para sua compra
----- Reiniciar compra -----
----- Bem Vindo Ao Cinema Kinoplex -----

Tudo bem? Serei Seu / Sua Atendente , qual seu nome?

```

6. Bibliotecas importadas :

```

import java.io.*;
import java.util.ArrayList;
import java.util.Scanner;
import java.util.Random; //Gera números aleatórios;

```

7. Resultados obtidos:

Por meio do do esquema do item 4 , implementamos um programa em java, cujo objetivo é a realização de compras de ingresso de um cinema , na qual por meio deu uma série de aplicações baseadas em POO, usamos : Encapsulamento , agregação (para simplificar a saída final e deixar o código mais interativo com o usuário de forma a se assemelhar a um padrão de projeto),classe,instâncias,mensagem,operações de interface pública,referência ao objeto,estado,métodos e método ao construtor , fomos capazes de modelar um sistema de compra e venda entre o usuário e um vendedor de cinema.

UML GERAL

