Package com.mycompany.proyectocine

Class ProyectoCine

java.lang.Object

com.mycompany.proyectocine.ProyectoCine

public class ProyectoCine extends Object

Sistema de administración de venta de boletos Cartago's University Cinema

Author:

Rau González

Field Summary [

Fields

Modifier and Type	Field	Description
static int	creditoDisponible	* Crédito disponible para compras *
static Console	cs	* Instancia para captura de teclas presionadas *
static int[]	diasDisponibles	* Lista de días disponibles en el ménú *
static int[][]	horariosDisponibles	* Lista de horarios disponibles en el menú *
static int[][]	peliculasDisponibles	* Lista de películas disponibles en el menú *
static int[]	preciosBoletos	* Precio de los boletos *
static int[][]	registroPagos	* Registro de pagos de reservaciones *
static int[][]	registroReservaciones	* Registro de reservaciones de boletos *
static int[][]	registroTransacciones	* Registro de transacciones hechas *
static Scanner	sc	* Recolector de datos ingresados *
static String []	vectorColumnas	* Vector de nombres de columnas *
static String []	vectorDias	* Vector de nombres de días *
static String []	vectorFilas	* Vector de nombres de filas *
static String []	vectorHorarios	* Vector de nombres de horarios *
static String []	vectorPeliculas	* Vector de nombres de películas *

Constructor Summary

Constructors

Constructor **Description**

ProyectoCine() Método constructor de clase

Method Summary

All Methods Stat	ic Methods Concrete Methods	
Modifier and Type	Method	Description
static String [][]	<pre>actualizarAsientos(String [] [] matrizAsientos, int[] [] matrizRegistros, String marca)</pre>	Método para actualizar matriz de asientos marcados
static void	administrar()	Método para verificar datos administrativos
static int[][]	<pre>agregarRegistro (int[] datosReservacion, int[] [] matrizRegistro)</pre>	Método para agregar un registro a una matriz de registros
static void	<pre>cancelarReservacion (int indiceRegistro)</pre>	Método para cancelar una reservación
static void	<pre>cargarDatosPruebas()</pre>	Método de carga de datos de prueba
static void	cargarFechas()	Método para generar fechas de reservación
static int	<pre>convertirTextoNumero(String datoIngresado)</pre>	Método para convertir dato de texto a número entero
static String [][]	<pre>copiarMatriz(String [] [] matrizOriginal)</pre>	Método para copiar datos de una matriz
static int[][]	<pre>eliminarRegistro (int indiceRegistro, int[] [] matrizRegistro)</pre>	Método para eliminar un registro a una matriz de registros
static void	<pre>finalizarPago (int indiceRegistro)</pre>	Método para finalizar el proceso de pago
static String	<pre>formatearTitulo (int[] datosReservacion)</pre>	Método para dar formato al título de la aplicación

static String [][]	<pre>generarAsientos (int[] datosReservacion)</pre>	Método para generar matriz de asientos
static void	<pre>guardarReservacion (int[] datosReservacion)</pre>	Método para registrar la reservación
static void	limpiarPantalla()	Método para limpiar datos de pantalla
static void	<pre>main(String [] args)</pre>	Método principal
static boolean	<pre>mensajeConfirmacion(String mensaje)</pre>	Método para mostrar mensaje de confirmación
static void	<pre>mostrarBannerAtencion()</pre>	Método para mostrar banner de atención
static void	mostrarBannerError()	Método para mostrar banner de error
static void	<pre>mostrarBannerPrincipal()</pre>	Método para mostrar banner principal
static void	<pre>mostrarError(String mensaje)</pre>	Método para mostrar un mensaje de error
static void	<pre>mostrarMensajeEspera()</pre>	Método para mostrar mensaje de espera
static void	<pre>mostrarNotificacion(String mensaje)</pre>	Método para mostrar barra de notificaciones
static void	<pre>mostrarOpciones (int[] opcionesDisponibles, String [] vectorOpciones)</pre>	Método para mostrar opciones del menú
static void	<pre>mostrarSala(String [] [] matrizAsientos)</pre>	Método para mostrar banner de la sala
static void	<pre>mostrarTitulo(String titulo, String pagina)</pre>	Método para mostrar el título de la pantalla
static String	<pre>obtenerHorarioPelicula (int[] datosReservacion)</pre>	Método para obtener el horario de la película
static String	<pre>obtenerNombreDia (int[] datosReservacion)</pre>	Método para obtener el nombre del día de la reservación
static String	<pre>obtenerNombrePelicula (int[] datosReservacion)</pre>	Método para obtener el nombre de la película
static String []	<pre>obtenerNumerosAsientos (int[] datosReservacion)</pre>	Método para obtener número de asientos en formato fila-columna
static int	<pre>obtenerSiguientePosicion(int[] [] matrizRegistro)</pre>	Método para obtener la primera posición libre de un registro

static int[][]	<pre>ordenarRegistros(int[] [] registroOriginal)</pre>	Método para ordenar un registro eliminando filas en blanco
static void	<pre>pantallaPrincipal()</pre>	Método para mostrar la pantalla principal
static void	<pre>procesarPago(int indiceRegistro)</pre>	Método para procesar un pago
static void	<pre>procesarReservacion (int indiceRegistro)</pre>	Método para procesar una reservación
static void	<pre>registrarTransaccion (int[] datosReservacion, int tipoTransaccion)</pre>	Método para registrar una transacción
static int[][]	<pre>registrosDiaPeliculaHorario (int[] datosReservacion, int[] [] matrizRegistro)</pre>	Método para generar matriz de registros por dia-pelicula-horario
static void	reservar()	Método para seleccionar el día de la reservación
static void	<pre>seleccionarAsientos (int[] datosReservacion)</pre>	Método para seleccionar los asientos
static void	<pre>seleccionarBoletos (int[] datosReservacion)</pre>	Método para seleccionar el tipo y cantidad de boletos
static void	<pre>seleccionarHorario (int[] datosReservacion)</pre>	Método para seleccionar el horario de la película
static void	<pre>seleccionarPelicula (int[] datosReservacion)</pre>	Método para seleccionar la película
static int	<pre>validarAsientos(int fila, int columna, int cantidad)</pre>	Método de validación de números de asientos
static boolean	<pre>validarCantidadBoletos (int[] listaBoletos, int tipoBoleto, int cantidadTransaccion)</pre>	Método para validar cantidad de boletos a reservar
static int	<pre>validarDisponibilidad(int fila, int columna, int cantidad, String [][] matrizAsientos)</pre>	Método de validación de disponibilidad de asientos
static void	verificar()	Método para verificar reservaciones registradas

Methods inherited from class java.lang.Object

clone , equals , finalize , getClass , hashCode , notify , notifyAll , toString , wait , wait , wait

Field Details

CS

public static Console cs

* Instancia para captura de teclas presionadas *

SC

public static Scanner sc

* Recolector de datos ingresados *

creditoDisponible

public static int creditoDisponible

* Crédito disponible para compras *

preciosBoletos

public static int[] preciosBoletos

* Precio de los boletos *

diasDisponibles

public static int[] diasDisponibles

* Lista de días disponibles en el ménú *

peliculasDisponibles

public static int[][] peliculasDisponibles

* Lista de películas disponibles en el menú *

horariosDisponibles

```
public static int[][] horariosDisponibles
```

* Lista de horarios disponibles en el menú *

registroPagos

```
public static int[][] registroPagos
```

* Registro de pagos de reservaciones *

registroReservaciones

```
public static int[][] registroReservaciones
```

* Registro de reservaciones de boletos *

registroTransacciones

```
public static int[][] registroTransacciones
```

* Registro de transacciones hechas *

vectorDias

public static String [] vectorDias

* Vector de nombres de días *

vectorPeliculas

public static String [] vectorPeliculas

* Vector de nombres de películas *

vectorHorarios

```
public static String [] vectorHorarios
```

* Vector de nombres de horarios *

vectorFilas

public static String [] vectorFilas

* Vector de nombres de filas *

vectorColumnas

public static String [] vectorColumnas

* Vector de nombres de columnas *

Constructor Details

ProyectoCine

public ProyectoCine()

Método constructor de clase

Method Details

main

public static void main(String [] args)

Método principal

Parameters:

args - Parámetros ingresados por consola

pantallaPrincipal

public static void pantallaPrincipal()

Método para mostrar la pantalla principal

reservar

public static void reservar()

Método para seleccionar el día de la reservación

seleccionarPelicula

public static void seleccionarPelicula(int[] datosReservacion)

Método para seleccionar la película

Parameters:

datos Reservacion - Vector de datos de reservación

seleccionarHorario

public static void seleccionarHorario(int[] datosReservacion)

Método para seleccionar el horario de la película

Parameters:

datos Reservacion - Vector de datos de reservación

seleccionarBoletos

public static void seleccionarBoletos(int[] datosReservacion)

Método para seleccionar el tipo y cantidad de boletos

Parameters:

datosReservacion - Vector de datos de reservación

seleccionarAsientos

public static void seleccionarAsientos(int[] datosReservacion)

Método para seleccionar los asientos

Parameters:

datos Reservacion - Vector de datos de reservación

guardarReservacion

public static void guardarReservacion(int[] datosReservacion)

Método para registrar la reservación

Parameters:

datos Reservacion - Vector de datos de reservación

procesarPago

public static void procesarPago(int indiceRegistro)

Método para procesar un pago

Parameters:

indiceRegistro - Posición del registro a procesar

finalizarPago

public static void finalizarPago(int indiceRegistro)

Método para finalizar el proceso de pago

Parameters:

indiceRegistro - Posición del registro a procesar

verificar

public static void verificar()

Método para verificar reservaciones registradas

procesarReservacion

public static void procesarReservacion(int indiceRegistro)

Método para procesar una reservación

Parameters:

indiceRegistro - Posición del registro a procesar

cancelarReservacion

public static void cancelarReservacion(int indiceRegistro)

Método para cancelar una reservación

Parameters:

indiceRegistro - Posición del registro a procesar

administrar

public static void administrar()

Método para verificar datos administrativos

mostrarError

public static void mostrarError(String mensaje)

Método para mostrar un mensaje de error

Parameters:

mensaje - Mensaje a mostrar

mensajeConfirmacion

public static boolean mensajeConfirmacion(String mensaje)

Método para mostrar mensaje de confirmación

Parameters:

mensaje - Mensaje a mostrar

Returns:

boolean Condición de resultado

mostrarBannerPrincipal

public static void mostrarBannerPrincipal()

Método para mostrar banner principal

mostrarBannerError

public static void mostrarBannerError()

Método para mostrar banner de error

mostrarBannerAtencion

public static void mostrarBannerAtencion()

Método para mostrar banner de atención

mostrarSala

public static void mostrarSala(String [][] matrizAsientos)

Método para mostrar banner de la sala

Parameters:

matrizAsientos - Matriz de datos de asientos

registrarTransaccion

public static void registrarTransaccion(int[] datosReservacion, int tipoTransaccion)

Método para registrar una transacción

Parameters:

datosReservacion - Datos de la reservación a registrar

tipoTransaccion - Tipo de transacción a registrar

generarAsientos

```
public static String [][] generarAsientos(int[] datosReservacion)
```

Método para generar matriz de asientos

Parameters:

datos Reservacion - Vector de datos de reservación

Returns:

String Matriz de asientos con número

registrosDiaPeliculaHorario

```
public static int[][] registrosDiaPeliculaHorario(int[] datosReservacion,
                                                   int[][] matrizRegistro)
```

Método para generar matriz de registros por dia-pelicula-horario

Parameters:

datosReservación - Datos de la reservación a procesar

matrizRegistro - Registro a procesar

Returns:

int[][] Matriz de listado de registros correspondientes al día, película y horario

ordenarRegistros

public static int[][] ordenarRegistros(int[][] registroOriginal)

Método para ordenar un registro eliminando filas en blanco

Parameters:

registroOriginal - Registro original a ordenar

Returns:

int[][] Registro copia ordenado

agregarRegistro

```
public static int[][] agregarRegistro(int[] datosReservacion,
                                       int[][] matrizRegistro)
```

Método para agregar un registro a una matriz de registros

Parameters:

datosReservacion - Datos de la reservación a agregar

matrizRegistro - Matriz donde se agregarán los datos

Returns:

int[][] Matriz con los datos agregados

eliminarRegistro

```
public static int[][] eliminarRegistro(int indiceRegistro,
                                       int[][] matrizRegistro)
```

Método para eliminar un registro a una matriz de registros

Parameters:

indiceRegistro - Posición del registro a eliminar

matrizRegistro - Matriz donde se agregarán los datos

Returns:

int[][] Matriz con los datos eliminados

actualizarAsientos

```
public static String [][] actualizarAsientos(String [][] matrizAsientos,
                                             int[][] matrizRegistros,
                                             String marca)
```

Método para actualizar matriz de asientos marcados

Parameters:

matrizAsientos - Matriz de datos de asientos

matrizRegistros - Matriz de datos registrados

marca - Símbolo a aplicar al asiento

Returns:

String[][] Matriz de asientos marcados

Throws:

Error - por matriz de datos registrados en blanco

copiarMatriz

```
public static String [][] copiarMatriz(String [][] matrizOriginal)
```

Método para copiar datos de una matriz

Parameters:

matrizOriginal - Matriz de datos a copiar

Returns:

String[][] Matriz de datos copiados

obtenerNumerosAsientos

```
public static String [] obtenerNumerosAsientos(int[] datosReservacion)
```

Método para obtener número de asientos en formato fila-columna

Parameters:

datosReservación - Datos de la reservación a procesar

Returns:

Stringp[] Matriz de números de asientos

obtenerSiguientePosicion

```
public static int obtenerSiguientePosicion(int[][] matrizRegistro)
```

Método para obtener la primera posición libre de un registro

Parameters:

matrizRegistro - Registro a verificar

Returns:

int Primera posición libre del registro

mostrarTitulo

```
public static void mostrarTitulo(String titulo,
                                String pagina)
```

Método para mostrar el título de la pantalla

Parameters:

titulo - Cadena de texto del título

pagina - Cadena de texto de nombre de página

formatearTitulo

```
public static String formatearTitulo(int[] datosReservacion)
```

Método para dar formato al título de la aplicación

Parameters:

datosReservacion - Vector de datos de la reservación

Returns:

Cadena de caracteres del título

mostrarNotificacion

```
public static void mostrarNotificacion(String mensaje)
```

Método para mostrar barra de notificaciones

Parameters:

mensaje - Mensaje a mostrar

mostrarOpciones

```
public static void mostrarOpciones(int[] opcionesDisponibles,
                                   String [] vectorOpciones)
```

Método para mostrar opciones del menú

Parameters:

opcionesDisponibles - Vector de índices de opciones disponibles

vectorOpciones - Vector de texto de opciones

validarAsientos

```
public static int validarAsientos(int fila,
                                   int columna,
                                   int cantidad)
```

Método de validación de números de asientos

Parameters:

fila - Fila del asiento selecionado

columna - Columna del asiento seleccionado

cantidad - Cantidad de asientos en la transacción

Returns:

int Código de verificación de asientos

validarDisponibilidad

```
public static int validarDisponibilidad(int fila,
                                         int columna,
                                         int cantidad,
                                         String [][] matrizAsientos)
```

Método de validación de disponibilidad de asientos

Parameters:

fila - Fila del asiento selecionado

columna - Columna del asiento seleccionado

cantidad - Cantidad de asientos en la transacción

matrizAsientos - Matriz de asientos según día-película-horario

Returns:

int Código de verificación de disponibilidad

Throws:

Error - de matriz de asientos vacía

convertirTextoNumero

public static int convertirTextoNumero(String datoIngresado)

Método para convertir dato de texto a número entero

Parameters:

datoIngresado - Dato ingresado por el usuario

Returns:

int Dato en formato de número

validarCantidadBoletos

```
public static boolean validarCantidadBoletos(int[] listaBoletos,
                                              int tipoBoleto,
                                              int cantidadTransaccion)
```

Método para validar cantidad de boletos a reservar

Parameters:

listaBoletos - Lista de boletos adquiridos

tipoBoleto - Tipo de boleto de la transacción

cantidadTransaccion - Cantidad de boletos a reservar

Returns:

boolean Validación de transacción

cargarFechas

public static void cargarFechas()

Método para generar fechas de reservación

obtenerNombreDia

public static String obtenerNombreDia(int[] datosReservacion)

Método para obtener el nombre del día de la reservación

Parameters:

datosReservacion - Datos de la reservación

Returns:

String Nombre del día de la reservación

obtenerNombrePelicula

public static String obtenerNombrePelicula(int[] datosReservacion)

Método para obtener el nombre de la película

Parameters:

datosReservacion - Datos de la reservación

Returns:

String Nombre de la película

obtenerHorarioPelicula

public static String obtenerHorarioPelicula(int[] datosReservacion)

Método para obtener el horario de la película

Parameters:

datosReservacion - Datos de la reservación

Returns:

String Nombre del horario

limpiarPantalla

public static void limpiarPantalla()

Método para limpiar datos de pantalla

Throws:

Error - de interrupción

mostrarMensajeEspera

public static void mostrarMensajeEspera()

Método para mostrar mensaje de espera

cargarDatosPruebas

public static void cargarDatosPruebas()

Método de carga de datos de prueba

Copyright © 2023. All rights reserved.