|  |  |  |
| --- | --- | --- |
| **Nombre:** Sergio Aramis Heredia Graham | | **Matrícula:** 2835840 |
| **Nombre del curso:**  Programación orientada a objetos | **Nombre del profesor:**  Wilberth Eduardo Lara Peraza | |
| **Módulo:**  1er. Módulo | **Actividad:**  Reto 4 | |
| **Fecha:** 04/Feb/2024 | | |
| **Bibliografía:** | | |

package com.mycompany.reto4;

import java.util.Scanner;

import java.util.ArrayList;

public class Reto4 {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("""

\*\*Programa asistente para registro de venta de boletos\*\*

A. Realizar venta de boleto

B. Mostrar datos de la lista de ventas

S. Finalizar programa

""");

char opcion = ' ';

ArrayList<String> listaPasajeros = new ArrayList<>(); //Nombres de todos los pasajeros

ArrayList<String> listaTicket = new ArrayList<>(); //Nombres de los pasajeros en un ticket

ArrayList<Integer> numPasajerosPorTicket = new ArrayList<>(); //Numero de pasajeros en cada ticket

do{

System.out.print("\nOpción (A, B o S): ");

opcion = scanner.next().toUpperCase().charAt(0);

switch(opcion){

case 'A':

listaTicket = metodos.pedirInformacionYMostrarTicket(listaTicket);

numPasajerosPorTicket.add(listaTicket.size()); // Guardamos la cantidad de pasajeros en el ticket n

listaPasajeros.addAll(listaTicket); //Agregamos los pasajeros en el ticket a la lista total de pasajeros

listaTicket.clear(); //Limpiamos la lista para que el metodo pueda usarla otra vez en caso de ser necesario

break;

case 'B':

metodos.mostrarListaPasajeros(listaPasajeros, numPasajerosPorTicket); // Se muestra la lista de todos los pasajeros indicando cuales son los compradores

System.out.println("Total de pasajeros: " + listaPasajeros.size());

System.out.println("Monto total de las ventas: $" + (listaPasajeros.size() \* 1250.50));

System.out.print("---------------------------------------\n");

break;

case 'S':

System.out.print("\n----------Fin del programa------------");

System.exit(0);

break;

default:

System.out.println("Opción no válida, por favor intente de nuevo.");

break;

}

}while(opcion != 'S');

}

}

package com.mycompany.reto4;

import java.util.Scanner;

import java.util.ArrayList;

public class metodos {

static Scanner scanner = new Scanner(System.in);

// Metodo para pedir la información de la compra para posteriormente mostrarle el ticket de compra al usuaria

public static ArrayList<String> pedirInformacionYMostrarTicket(ArrayList<String> listaPasajeros){

System.out.println("\n----------------------------------------");

System.out.print("Ingresar nombre del comprador: ");

listaPasajeros.add(scanner.next());

System.out.print("Cantidad de pasajeros adicionales: ");

int adicional = solicitarInt();

if(adicional > 0){

for(int i = 1; i <= adicional; i++){

System.out.print("Ingresar nombre del pasajero adicional " + i + ": ");

listaPasajeros.add(scanner.next());

}

}

System.out.println("-----------Ticket de venta-----------");

System.out.println("Cantidad de boletos: " + listaPasajeros.size());

System.out.println("Importe de venta: $" + (listaPasajeros.size() \* 1250.50));

System.out.println("Nombre del comprador: " + listaPasajeros.get(0));

System.out.println("Pasajeros adicionales: " + (listaPasajeros.size() - 1));

System.out.println("-------------------------------------");

return listaPasajeros;

}

/\* Metodo para mostrar la lista de pasajeros, lo especial de este metodo es que imprimira (comprador) cuando sea momento de

imprimir al pasajero que compro los tickets \*/

public static void mostrarListaPasajeros(ArrayList<String> listaPasajeros, ArrayList<Integer> listaPasajerosPorTicket){

System.out.println("\n-----------Lista de pasajeros----------");

int pasajerosEnlistados = 0; //Variable que llevara la cuenta de los pasajeros que ya fueron impresos

for (int i = 0; i < listaPasajerosPorTicket.size();i++){ //El bucle se repetira la cantidad de tickets vendidos

int numPasajerosEnTicket = listaPasajerosPorTicket.get(i); // Guarda la cantidad de pasajeros por ticket

// Nos aseguramos de que no se acceda a índices fuera de los límites

if (pasajerosEnlistados < listaPasajeros.size()) {

System.out.println("Pasajero " + (pasajerosEnlistados + 1) + " (comprador): " + listaPasajeros.get(pasajerosEnlistados));

pasajerosEnlistados++; // Incrementar después de enlistar al comprador

// Mostramos a los pasajeros adicionales

for (int j = 1; j < numPasajerosEnTicket && pasajerosEnlistados < listaPasajeros.size(); j++) {

System.out.println("Pasajero " + (pasajerosEnlistados + 1) + ": " + listaPasajeros.get(pasajerosEnlistados));

pasajerosEnlistados++; // Incrementar por cada pasajero adicional enlistado

}

}

}

}

//Metodo para solicitar un numero y que no termine en error el programa

public static int solicitarInt(){

while (true) {

String intValue = scanner.next();

try {

return Integer.parseInt(intValue);

}catch (NumberFormatException e){

System.out.println("\nError: No has introducido un valor INT\nEjemplo de valores validos: 4, 8, 19 \n");

System.out.print("Numero int: ");

// No es necesario hacer nada aquí, el bucle continuará automáticamente

}

}

}

}