Universidad de Margarita

Alma mater del Caribe

Decanato de Ingeniería de Sistemas

Cátedra: Estructura de Datos

Sección M-01



**Gestión de Información para la Venta de Boletos en una Aerolínea.**

Profesor:

Parra, Michel.

Integrantes:

Guevara Lunar, Jose Salvador.

CI: 16825525.

Velásquez, Jhonattan.

CI: 23590348.

El Valle del Espíritu Santo, miércoles 02 de octubre de 2013.

**Descripción general:** Es un programa desarrollado para la gestión y venta de boletos para una aerolínea cualquiera que crea y mantiene un registro de:

* **Clientes:** Se crea un archivo llamado **<cliente.txt>** en el cual se registran los siguientes datos:
  + **Nombre:** Nombre de la persona que compro el boleto aéreo.
  + **Apellido:** Apellido de dicha persona.
  + **Cédula:** Número de Cédula de Identidad de la persona que hace la compra, ese número se utiliza como campo clave para la búsqueda de datos dentro del archivo.
  + **Teléfono:** Número de teléfono de la persona.
  + **Dirección:** Especifica el sector o área donde reside la persona.
* **Vuelo:** Crea un archivo llamado **<vuelo.txt>** donde se registra y consulta la información de los vuelos a medida que se van creando. Este archivo contiene:
  + **Número de Vuelo:** Identifica cada vuelo con un número único consecutivo.
  + **Fecha de Vuelo:** Indica el día, el mes y el año de partida del vuelo.
  + **Hora de Vuelo:** Indica la hora y minuto de partida del vuelo.
  + **Ciudad Origen:** Indica, mediante un código de siglas estandarizado por la aviación civil venezolana, la ciudad desde donde partirá el vuelo.
  + **Ciudad Destino:** Al igual que la anterior, esta indica la ciudad a la que se dirige el vuelo.
  + **Tarifa Base del Vuelo:** Indica el precio fijado para cada vuelo.
* **Boletos:** Crea un archivo correspondiente a cada vuelo, con el nombre **<boletos#.txt>** donde <#> es el número de vuelo para el cual se generará el boleto. Este archivo contiene:
  + **Número de Boleto:** Identifica cada boleto con un número único consecutivo.
  + **Número de Vuelo:** Indica a que vuelo pertenece cada boleto. Aun cuando el archivo ya está identificado con el número de vuelo, ese se coloca en la información guardada de cada boleto para facilitar la referencia al momento de abstraer los datos del mismo.
  + **Cédula:** Indica el número de Cédula de Identidad de la persona que adquiere el boleto, al momento de imprimir el boleto por pantalla se hace una búsqueda en el archivo **<cliente.txt>** para abstraer los datos que coincidan con dicha cédula y completar los campo de información del cliente en la muestra por pantalla.
  + **Columna:** indica la línea vertical en la matriz de asientos, este valor es un número menor al mostrado gráficamente ya que la matriz inicia el conteo desde la posición 0.
  + **Fila:** Indica con un número (de 0 a 5), la posición de la letra del asiento (de |A| a |F|).
  + **Clase:** Indica la clase en la que se encuentra el asiento elegido por el cliente de entre tres (3) de ellas:
    - **C:** con el mismo valor que la tarifa base, va desde la columna número 9 hasta la 16.
    - **B:** con un recargo de 15% sobre la tarifa base, va desde la columna 4 hasta la 8.
    - **A:** con un recargo de 25% sobre la tarifa base, de la columna 1 a la 3.
  + **Tarifa:** Indica el valor del boleto según la clase que se haya seleccionado.

**Manual de utilización y referencia:**

**Descripción de menús**

**Menú Principal:** Es el menú de entrada al sistema, es la primera referencia visual de las capacidades y características del sistema. Está compuesto por:

1. **Registro y Gestión de Vuelos.**
   1. **Crear Vuelo:** 
      * Se crea un archivo de nombre **<vuelo.txt>** o se verifica su existencia, si el archivo ya existe se lee para buscar el número para crear el próximo vuelo y se imprime en pantalla de color rojo.
      * Se crea o verifica la existencia del archivo **<boletos#.txt>** donde **<#>** es el número de vuelo a crear.
      * Solicita la fecha de salida del vuelo, debe estar en formato: dd/mm/aaaa.
      * Solicita la hora de salida del vuelo, debe estar en formato 24h: hh:mm.
      * Pide que se seleccione la ciudad desde donde partirá el vuelo de una lista mostrada en pantalla.
      * Pide seleccionar la ciudad de destino de la misma forma.
      * Solicita el ingresar el monto para establecer la tarifa base del vuelo.
      * Regresa al menú Registro y Gestión de Vuelos.
   2. **Buscar Vuelo:**
      * Solicita que se ingrese en número del vuelo que se desea consultar.

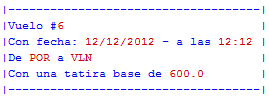


Figura 1.1

* + - Mostrará la información del vuelo
      * Numero de vuelo.
      * Fecha y hora del vuelo.
      * Ciudad Origen y Ciudad Destino.
      * Tarifa base.
    - En caso de introducir un número de vuelo que no exista en la lista del archivo **<vuelo.txt>** retornará un mensaje indicando que el vuelo no se ha encontrado y regresará al menú.

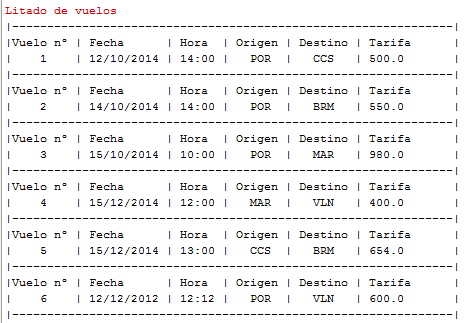


Figura 1.2

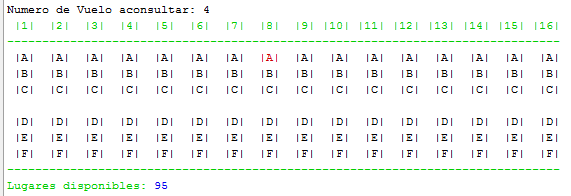
* 1. **Ver Lista de Vuelos:**
     + Se genera una lista con la información de todos los vuelos creados.
  2. **Ver Disponibilidad en Vuelo:**
     + Numero de vuelo a consultar: Solicita el número de vuelo del cual se desea conocer su disponibilidad de asientos.
     + Se genera los siguiente:
     + Los números indican las columnas y las letras indican cada asiento en la columna.
     + Los asientos de color negro representan lugares disponibles y los de color rojo están ocupados.
     + Los lugares disponibles se muestran en la última línea de color azul.

Figura 1.3

* 1. **Regresar al menú anterior:** regresa al usuario al menú principal

1. **Registro Gestión de Clientes:**
   1. **Registrar Cliente Nuevo:**

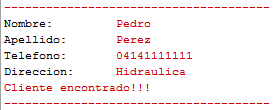


Figura 1.1

* + - Solicita la Cédula de Identidad de la persona que se va a incluir en el registro.
    - Si la cedula ya existe en el registro, el programa imprimirá en pantalla los datos adjuntos a dicha cédula.
    - Si la cédula no existe en el registro se procede a solicitar los datos necesarios para incluir al cliente en el registro
  + **Nombre.**
  + **Apellido.**

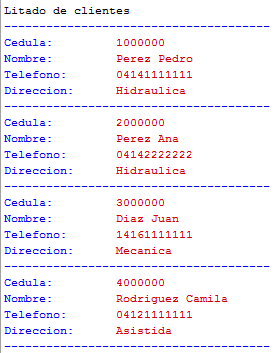


Figura 2.2

* + **Teléfono.**
  + **Dirección.**

* 1. **Buscar Cliente Registrado:**
     + Solicita el número de cédula del cliente que se desea consultar.
     + Si no se encuentra una coincidencia en el archivo **<cliente.txt>** se imprime un mensaje indicando que no se encontró el cliente buscado en el registro.
     + Si se encuentra una coincidencia se genera una impresión en pantalla con los datos del cliente.
  2. **Ver Lista de Clientes:**
     + Se genera un listado con los datos de los clientes que se encuentran en el registro.
  3. **Regresar al menú anterior:** Regresa al usuario al menú principal.

1. **Gestión y Venta de Boletos.**

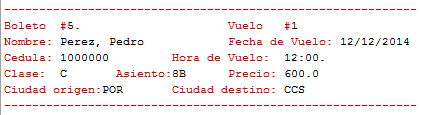


Figura 3.1

* 1. **Venta de Boletos:**
     + Solicita el número de vuelo en el que se venderá el boleto.
     + Muestra el número de vuelo y el número de boleto a emitir.
     + Solicita un número de cédula para consultar la existencia del cliente en los registros.
     + Si el cliente es encontrado se cargan y muestran sus datos en pantalla del mimo modo que en la **Figura 2.1**.
     + En caso q el cliente no se encuentre en el archivo se procede a solicitar los datos del mismo para hacer la inclusión.

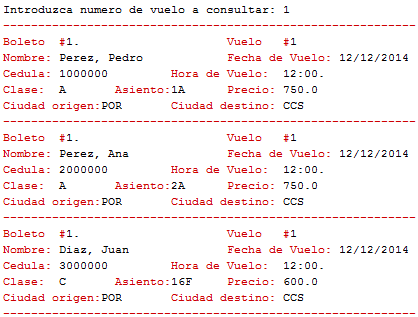


Figura 3.2

* + - Luego de alguno de los 2 casos anteriores, se imprime en pantalla la distribución del vuelo con su respectiva disponibilidad, como se muestra en la **Figura 1.3**.
    - Se solicita ingresar el número de línea en la que se desea comprar el boleto tomando en cuenta que de la línea 1 a la 3 son los asientos de la clase **A**, de la 4 a la 8 pertenecen a la clase **B** y de la 9 a la 16 a la clase **C**.
    - Inmediatamente después de esto se imprime en pantalla el boleto correspondiente.
  1. **Ver Lista de Boletos:**
     + Solicita el número de un vuelo para crear una lista de todos los boletos vendidos para dicho vuelo.
  2. **Regresar al menú anterior:** Regresa al usuario al menú principal.

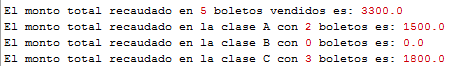


Figura 4.1

1. **Generación Reportes:**
   1. **Ventas Totales:** Refleja el monto recaudado por la venta de boletos en tres (3) fases:
      * Monto Total: Imprime el total recaudado por la venta de boletos en todos los vuelos.
      * Monto Total por Clase: Imprime el monto total recaudado por la venta de boletos en todos los vuelos pero separados en cada una de las tres (3) clases que posee cada vuelo.

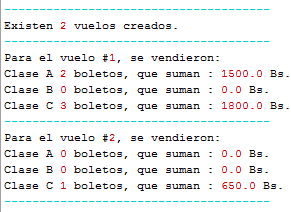


Figura 4.2

* 1. **Vetas por Clases:**
     + Refleja la cantidad de vuelos vendidos.
     + Imprime un recuadro con la información de venta para cada vuelo, desglosado por clase.
     + Muestra la cantidad de boletos vendidos en cada clase y el monto que representa.
  2. **Regresar al menú anterior:** Regresa al usuario al menú principal.

1. **Salir:** Termina la ejecución del programa con un mensaje: **<Adiós!!!>**

**Código fuente:**

|  |
| --- |
| NewAirLineSeller.java |

/\*Venta de boletos aéreos para JSalvaG Airlines con una flota de 10 aviones Boeing 373, con vuelos a las principales ciudades venezolanas:

\* Caracas

\* Porlamar

\* Valencia

\* Maracaibo

\* Barcelona

\* Barquisimeto

\* Cada aeronave cuenta con 3 clases: VIP, Ejecutiva y turística.

\*/

package newairlineseller;

import java.io.IOException;

/\*\*

\* **@author** JSalvaG

\*/

public class **NewAirLineSeller** {

public static void ***main***(String[] args) throws IOException {

Menu menu = new Menu();

menu.menuPrincipal();

}

}

|  |
| --- |
| Menu.java |

package newairlineseller;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

public class **Menu** {

BufferedReader **br** = new BufferedReader (new InputStreamReader(System.***in***));

Cliente **c** = new Cliente();

Manejador **m** = new Manejador();

Vuelos **v** = new Vuelos();

Boleto **b** = new Boleto();

Reportes **r** = new Reportes();

void **menuPrincipal**() throws IOException{

**m**.crear("vuelo");

**m**.crear("cliente");

System.***out***.println("\033[35m-------------------------------------------------------------------------------");

System.***out***.println("\033[35m----------------------------- BienVenido --------------------------------------");

System.***out***.println("\033[35m----------------Sistema de venta de Boletos para JSalvaG Airlines--------------");

System.***out***.println("\033[35m-------------------------------------------------------------------------------");

System.***out***.println("\033[35m-----------------------------Selecciones su accion-----------------------------");

System.***out***.println("\033[35m-------------------------------------------------------------------------------");

boolean salir = false;

do{

System.***out***.println("\n1.-Gestion de Vuelos.");

System.***out***.println("2.-Gestion de clientes.");

System.***out***.println("3.-Gestion de boletos.");

System.***out***.println("4.-Reportes.");

System.***out***.print("5.-Salir.\nSeleccion: ");

switch (**br**.readLine()) {

case "1": //GEstion de vuelos...

subMenu2();

break;

case "2":

subMenu1();

break;

case "3":

subMenu3();

break;

case "4":

subMenu4();

break;

case "5":

salir = true;

System.***out***.println("Adios!!!");

break;

default:

System.***out***.println("Opcion invalida.");

break;

}

}while(!salir);

}

void **subMenu1**() throws IOException{

boolean salir = false;

do{

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

System.***out***.println("- Registro y gestion de clientes -");

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

System.***out***.println("1.-Registrar Cliente Nuevo");

System.***out***.println("2.-Buscar Cliente Registrado");

System.***out***.println("3.-Ver lista de Clientes");

System.***out***.println("4.-Regresar al menu anterior");

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

System.***out***.print("Opcion: ");

String op = **br**.readLine();

switch(op){

case "1":

**c**.crearCliente();

break;

case "2":

**c**.buscarCliente();

break;

case "3":

**m**.leerArchivoCliente();

break;

case "4":

salir = true;

break;

default:

System.***out***.println("Opcion invalida");

salir = false;

}

}while(!salir);

}

void **subMenu2**() throws IOException{//Submenu de gestion de vuelos

boolean salir = false;

do{

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

System.***out***.println("- Registro y gestion de vuelos -");

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

System.***out***.println("1.-Crear Vuelo");

System.***out***.println("2.-Buscar Vuelo");

System.***out***.println("3.-Ver lista de Vuelos");

System.***out***.println("4.-Ver Disponibilidad en Vuelo");

System.***out***.println("5.-Regresar al menu anterior");

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

System.***out***.print("Opcion: ");

String op = **br**.readLine();

switch(op){

case "1"://crear vuelo.

**v**.crearVuelo();

break;

case "2"://ver vuelos creados.

**v**.verVuelos();

break;

case "3":

**m**.leerArchivoVuelo();

break;

case "4":

System.***out***.print("Numero de Vuelo aconsultar: ");

String nV = **br**.readLine();

**v**.disponibilidad(nV);

break;

case "5":

salir = true;

break;

default:

System.***out***.println("Opcion invalida");

salir = false;

}

}while(!salir);

}

void **subMenu3**() throws IOException{

boolean salir = false;

do{

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

System.***out***.println("- Gestion y venta de boletos -");

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

System.***out***.println("1.-Venta de Boletos");

System.***out***.println("2.-Ver lista de Boletos");

System.***out***.println("3.-Regresar al menu anterior");

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

System.***out***.print("Opcion: ");

String op = **br**.readLine();

switch(op){

case "1":

**b**.crearBoleto();

break;

case "2":

**b**.imprimirBoleto();

break;

case "3":

salir = true;

break;

default:

System.***out***.println("Opcion invalida");

salir = false;

}

}while(!salir);

}

void **subMenu4**() throws IOException{

boolean salir = false;

do{

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

System.***out***.println("- Generacion de reportes -");

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

System.***out***.println("1.-Ventas totales.");

System.***out***.println("2.-Vuelos vendidos.");

System.***out***.println("3.-Regresar al menu anterior");

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

System.***out***.print("Opcion: ");

String op = **br**.readLine();

switch(op){

case "1":

**r**.imprimirTotal();

break;

case "2":

**r**.vueloVendidos();

break;

case "3":

salir = true;

break;

default:

System.***out***.println("Opcion invalida");

salir = false;

}

}while(!salir);

}

}

|  |
| --- |
| Manejador.java |

/\*Clase creada para la interaccion con los archivos de texto\*/

package newairlineseller;

import java.io.\*;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class **Manejador** {

PrintWriter **pf**;

FileReader **fr**;

BufferedReader **br** = new BufferedReader(new InputStreamReader(System.***in***));

void **crear**(String nArchivo){

/\*crea el archivo vacio o verifica su existencia,

\* necesario para almacenar los datos que generan los distintos metodos

\* nArchivo es el parametro con el nombre que tendra el ficrero de texto

\*/

try{

**pf** = new PrintWriter(new FileWriter(nArchivo+".txt",true));

**pf**.close();

System.***out***.println("Archivo "+nArchivo+".txt se genero o existe...");

}catch (Exception e){

System.***out***.println("Error en creacion de archivo.\nError: "+e);

}

}

void **guardar**(String nArchivo, String nDatos){

/\*crea o verifica la existencia del archivo para agregar los datos

\* generados en los diferentes metodos que lo invocan

\*/

try{

**pf** = new PrintWriter(new FileWriter(nArchivo+".txt",true));

**pf**.println(nDatos);

**pf**.close();

System.***out***.println("Archivo "+nArchivo+".txt se genero o existe...");

}catch (Exception e){

System.***out***.println("Error de guardado en archivo.\nError: "+e);

}

}

String[] **buscar**(String nArchivo, String ID){

/\*busca por numero de cedula una linea especifica dentro de un archivo de texto

\*/

String[] datos;

try {

**fr** = new FileReader(nArchivo+".txt");

BufferedReader brf = new BufferedReader(**fr**);

String linea;

while((linea=brf.readLine())!=null){

datos = linea.split("-");

if(datos[0].equalsIgnoreCase(ID)){

return datos;

}

}

} catch (Exception e) {

}

return datos = null;

}

double[] **buscarMontos**(int nVuelo) throws IOException{

/\*busca los montos en un archivo de texto\*/

String[] datos;

double[] suma = {0,0,0,0,0,0,0,0};

//suma contiene{0-monto total, 1-monto clase A, 2-monto clase B, 0-monto clase C,

//4-cantidad total de boletos, 5-cant boletos A, 6-cant boletos B, 7-cant boletos C}

**fr** = new FileReader("boletos"+nVuelo+".txt");

BufferedReader brf = new BufferedReader(**fr**);

String linea;

while((linea=brf.readLine())!=null){

datos = linea.split("-");

if(datos!=null){

suma[4]++;

double aux = Double.parseDouble(datos[6]);

suma[0] += aux;

if(datos[5].equals("A")){

suma[1] += aux;

suma[5]++;

}

if(datos[5].equals("B")){

suma[2] += aux;

suma[6]++;

}

if(datos[5].equals("C")){

suma[3] += aux;

suma[7]++;

}

}

}

**fr**.close();

return suma;

}

int **buscar**(String nArchivo) throws IOException{

/\*busca en un archivo el valor de la primera posicion en cada linea para

\* saber el siguiente numero para el el vuelo\*/

int mayor = 0;

String[] datos;

**fr** = new FileReader(nArchivo+".txt");

BufferedReader brf = new BufferedReader(**fr**);

String linea;

while((linea = brf.readLine())!=null){

datos = linea.split("-");

if (datos.**length**==1){

return 0;

}else{

// for (int i = 0; i < datos.length; i++) {

int indice = Integer.parseInt(datos[0]);

if(indice>= mayor){

mayor = indice;

}else{

return -1;

}

// }

}

}

return mayor; //retorna el valor del ultimo vuelo

}

void **leerArchivoCliente**() throws IOException{

/\*Imprime en pantalla una lista entendible con el contenido del archivo

\* de clientes

\*/

**fr** = new FileReader("cliente.txt");

BufferedReader brf = new BufferedReader(**fr**);

String linea;

System.***out***.println("Litado de clientes");

while((linea=brf.readLine())!=null){

String[] datos = linea.split("-");

System.***out***.println("\033[34m--------------------------------------");

System.***out***.println("\033[34mCedula:\t\t\033[31m"+datos[0]);

System.***out***.println("\033[34mNombre:\t\t\033[31m"+datos[2]+" "+datos[1]);

System.***out***.println("\033[34mTelefono:\t\033[31m"+datos[3]);

System.***out***.println("\033[34mDireccion:\t\033[31m"+datos[4]);

}

System.***out***.println("\033[34m--------------------------------------");

}

void **leerArchivoVuelo**() throws IOException{

/\*Imprime en pantalla una lista entendible con el contenido del archivo

\* de clientes

\*/

**fr** = new FileReader("vuelo.txt");

BufferedReader brf = new BufferedReader(**fr**);

String linea;

System.***out***.println("\033[31mLitado de vuelos");

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

while((linea=brf.readLine())!=null){

String[] datos = linea.split("-");

System.***out***.println("|Vuelo nº | Fecha | Hora | Origen | Destino | Tarifa\t|");

System.***out***.println("| "+datos[0]+"\t | "+datos[1]+" | "+datos[2]+" | "+datos[3]+" | "+datos[4]+" | "+datos[5]+"\t|");

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

// System.out.println( "Nº de vuelo:\t"+datos[0]);

// System.out.println( "Fecha y hora:\t"+datos[2]+" "+datos[1]);

// System.out.println("Ciudad Origen:\t"+datos[3]

// +"\nCiudad Destino:\t"+datos[4]);

// System.out.println( "Tarifa Base:\t"+datos[5]);

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

// System.out.println();

}

}

boolean **validarCedula**(String ci){

//valida la cedula comparandola con un patron

Pattern pat = Pattern.compile("[0-9]{6,8}");

Matcher mat = pat.matcher(ci);

if(mat.matches()){return true;}

else{return false;}

}

boolean **validarNombApe**(String nom){

//valida el nombre y el apellido comparandolo con un patron

boolean patF = Pattern.matches("[a-zA-Z]{1,100}",nom);

return patF;

}

boolean **validarTelf**(String telf){

//valida el telefono comparandolo con un patron

boolean patF = Pattern.matches("[0-9]{11}",telf);

return patF;

}

boolean **validarDir**(String dir){

//valida la direccion comparandola con un patron

boolean patF = Pattern.matches("[a-zA-Z ]{1,200}",dir);

return patF;

}

boolean **validarTarifa**(String t){

boolean patT = Pattern.matches("\\d+\\.\*\\d\*",t);

return patT;

}

boolean **validarFila**(String fil){

//verifica que el numero de fila este entre 1 y 16

boolean patT = Pattern.matches("[1-16]",fil);

return patT;

}

boolean **validarHora**(String vHora){

//verifica que la hora ingresada tenga un formato valido.

boolean patH = Pattern.matches("[0-2]?\\d:[0-5]?\\d",vHora);

return patH;

}

boolean **validarFecha**(String vFecha){

//verifica que la fecha tenga un formato valido.

boolean patF = Pattern.matches("\\d+/\\d+/\\d{4}",vFecha);

return patF;

}

String **selecCiudad**() throws IOException{

/\*muestra una lista de ciudades disponibles para los vuelos y devuelve

\* el codigo para una de ella\*/

String ciudad = "";

boolean val=false;

do {

System.***out***.println("1 - Caracas.");

System.***out***.println("2 - Barquisimeto.");

System.***out***.println("3 - Valencia.");

System.***out***.println("4 - Barcelona.");

System.***out***.println("5 - Maracaibo.");

System.***out***.println("6 - Porlamar");

System.***out***.print("Seleccione Ciudad: ");

String aux3 = **br**.readLine();

if(Pattern.matches("[1-6]", aux3)){

switch(aux3){

case "1" : ciudad = "CCS"; val=true; break;

case "2" : ciudad = "BRM"; val=true; break;

case "3" : ciudad = "VLN"; val=true; break;

case "4" : ciudad = "BLA"; val=true; break;

case "5" : ciudad = "MAR"; val=true; break;

case "6" : ciudad = "POR"; val=true; break;

}

}else{

System.***out***.println("Seleccion invalida.");

val=false;

}

} while (!val);

return ciudad;

}

boolean **verDispon**(int fil, int col, String numVuelo){

// verifica la disponibilidad de un asiento en un vuelo especifico

String[] datos;

try {

**fr** = new FileReader("boletos"+numVuelo+".txt");

BufferedReader brf = new BufferedReader(**fr**);

String linea;

while((linea=brf.readLine())!=null){

datos = linea.split("-");

if((datos[3].equals(fil+""))&&(datos[4].equals(col+""))){

return false;

}

}

} catch (Exception e) {

System.***err***.println(e);

}

return true;

}

}

|  |
| --- |
| Cliente.java |

/\*Creacion y gestion del registro de clientes\*/

package newairlineseller;

import java.io.\*;

import java.util.logging.Level;

import java.util.logging.Logger;

public class **Cliente** {

boolean **si**, **si1**;

String **nom**, **ape**, **ci**, **dir**, **telf**, **aux**;

Manejador **man** = new Manejador();

BufferedReader **br** = new BufferedReader(new InputStreamReader(System.***in***));

void **crearCliente**(){

/\*crea un archivo con la informacion de los clientes que compran un voleto

\* se emplean ciclos reperitivos para verificar que los datos ingresados

\* son correctos o tiene un formato compatible con el sistema\*/

try {

**man**.crear("cliente");

do{

System.***out***.print("Cedula: \t");

**aux** = **br**.readLine();

if(**man**.validarCedula(**aux**)){

**ci** = **aux**;

String[] ver = **man**.buscar("cliente", **ci**);

if (ver==null){

pedirDatos();

String data = **ci**+"-"+**nom**+"-"+**ape**+"-"+**telf**+"-"+**dir**;

**man**.guardar("cliente", data);

**si** = true;

}else{

if(ver[0].equals(**ci**)){

System.***out***.println("\033[30mNombre:\t\t\033[31m"+ver[1]);

**nom** = ver[1];

System.***out***.println("\033[30mApellido:\t\033[31m"+ver[2]);

**ape** = ver[2];

System.***out***.println("\033[30mTelefono:\t\033[31m"+ver[3]);

**telf** = ver[3];

System.***out***.println("\033[30mDireccion:\t\033[31m"+ver[4]);

**dir** = ver[4];

System.***out***.println("\033[31mCliente encontrado!!!\033[30m");

**si** = true;

}else{

pedirDatos();

String data = **ci**+"-"+**nom**+"-"+**ape**+"-"+**telf**+"-"+**dir**;

**man**.guardar("cliente", data);

**si** = true;

}

}

}else{

System.***out***.println("Cedula invalida..");

**si** = false;

}

}while(!**si**);

} catch (IOException ex) {

Logger.getLogger(Cliente.class.getName()).log(Level.***SEVERE***, null, ex);

}

}

void **pedirDatos**() throws IOException{

// pide los datos basicos de informacion para el archivo cliente.txt

do{

System.***out***.print("Nombre:\t\t");

**aux** = **br**.readLine();

if(**man**.validarNombApe(**aux**)){

**nom** = **aux**;

**si1** = true;

}else{

System.***out***.println("Nombre invalido!!!");

**si1** = false;

}

}while(!**si1**);

do{

System.***out***.print("Apelido:\t");

**aux** = **br**.readLine();

if(**man**.validarNombApe(**aux**)){

**ape** = **aux**;

**si1** = true;

}else{

System.***out***.println("Apellido invalido!!!");

**si1** = false;

}

}while(!**si1**);

do{

System.***out***.print("Telefono:\t");

**aux** = **br**.readLine();

if(**man**.validarTelf(**aux**)){

**telf** = **aux**;

**si1** = true;

}else{

System.***out***.println("Telefono invalido!!!");

**si1** = false;

}

}while(!**si1**);

do{

System.***out***.print("Diereccion:\t");

**aux** = **br**.readLine();

if(**man**.validarDir(**aux**)){

**dir** = **aux**;

**si1** = true;

}else{

System.***out***.println("Direccion invalido!!!");

**si1** = false;

}

}while(!**si1**);

}

String[] **pedirDatos2**() throws IOException{

/\*solicita los datos de un cliente y devuelve un arreglo para utilizar

\* los datos en otra clase\*/

String[] datos = new String[4];

do{

System.***out***.print("Nombre:\t\t");

**aux** = **br**.readLine();

if(**man**.validarNombApe(**aux**)){

datos[0] = **aux**;

**si1** = true;

}else{

System.***out***.println("Nombre invalido!!!");

**si1** = false;

}

}while(!**si1**);

do{

System.***out***.print("Apelido:\t");

**aux** = **br**.readLine();

if(**man**.validarNombApe(**aux**)){

datos[1] = **aux**;

**si1** = true;

}else{

System.***out***.println("Apellido invalido!!!");

**si1** = false;

}

}while(!**si1**);

do{

System.***out***.print("Telefono:\t");

**aux** = **br**.readLine();

if(**man**.validarTelf(**aux**)){

datos[2] = **aux**;

**si1** = true;

}else{

System.***out***.println("Telefono invalido!!!");

**si1** = false;

}

}while(!**si1**);

do{

System.***out***.print("Diereccion:\t");

**aux** = **br**.readLine();

if(**man**.validarDir(**aux**)){

datos[3] = **aux**;

**si1** = true;

}else{

System.***out***.println("Direccion invalido!!!");

**si1** = false;

}

}while(!**si1**);

return datos;

}

void **buscarCliente**() throws IOException{

// imprime por pantalla los datos de un cliente registrado en el archivo

do{

System.***out***.print("Introduzca cedula a buscar: ");

**aux** = **br**.readLine();

if(**man**.validarCedula(**aux**)){

**ci** = **aux**;

String[] ver = **man**.buscar("cliente", **ci**);

if (ver==null){

System.***out***.println("\033[31mCliente no encontrado...\033[30m");

**si** = true;

}else{

if(ver[0].equals(**ci**)){

System.***out***.println("\033[31m--------------------------------------");

System.***out***.println("\033[30mNombre:\t\t\033[31m"+ver[1]);

**nom** = ver[1];

System.***out***.println("\033[30mApellido:\t\033[31m"+ver[2]);

**ape** = ver[2];

System.***out***.println("\033[30mTelefono:\t\033[31m"+ver[3]);

**telf** = ver[3];

System.***out***.println("\033[30mDireccion:\t\033[31m"+ver[4]);

**dir** = ver[4];

System.***out***.println("\033[31mCliente encontrado!!!\033[30m");

System.***out***.println("\033[31m--------------------------------------");

**si** = true;

}else{

pedirDatos();

String data = **ci**+"-"+**nom**+"-"+**ape**+"-"+**telf**+"-"+**dir**;

**man**.guardar("cliente", data);

**si** = true;

}

}

}else{

System.***out***.println("Cedula invalida..");

**si** = false;

}

}while(!**si**);

}

}

|  |
| --- |
| Vuelos.java |

/\*Creacion y gestion vuelos\*/

package newairlineseller;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

public class **Vuelos** {

String **fechaVuelo**, **horaVuelo**, **ciudadOrigen**, **ciudadDestino**, **aux**, **color**;

double **tarifaBase**;

boolean **si**, **dispon**;

int **cantNaves** = 10, **disp** = 96;

Manejador **man** = new Manejador();

BufferedReader **br** = new BufferedReader(new InputStreamReader(System.***in***));

void **crearVuelo**() throws IOException{//Crear vuelo

**man**.crear("vuelo");//Crear archivo vuelo.txt

int num = **man**.buscar("vuelo")+1;//busca en el numero del ultimo vuelo creado y le suma 1 para crear el siguiente vuelo.

System.***out***.println("\033[30mVuelo #\033[31m"+num+"\033[30m.");//Imprime en pantalla el numero de vuelo

do{//bucle para verificacion de datos ingresados para la cracion del vuelo

System.***out***.print("Fecha del Vuelo: ");

**aux** = **br**.readLine();

if(**man**.validarFecha(**aux**)){//evalua mediante una funcion si el dato fue ingrado en el formato correcto.

**fechaVuelo** = **aux**;

**si** = true;

//Si el dato es correcto se carga el valor de la variable auxiliar en la variable definitiva.

//la variable "si" se evalua a verdadero.

}else{

System.***out***.println("Fecha invalida.");

**si** = false;

//si el dato no tiene un formato valido, le indica al usuario que el dato no paso la verificacion

//evalua la variable "si" a falso para que el bucle se repita y permita ingresarl el dato nuevamente.

}

}while(!**si**);//El bucle se reperita mientras la variable "si" se evalue falsa.

do{//bucle de virificacion del formato de la hora.

System.***out***.print("Hora de salida: ");

**aux** = **br**.readLine();

if(**man**.validarHora(**aux**)){

**horaVuelo** = **aux**;

**si** = true;

}else{

System.***out***.println("Hora invalida.");

**si** = false;

}

}while(!**si**);

System.***out***.println("Ciudad de origen: ");

**ciudadOrigen** = **man**.selecCiudad();//mediante la fucion selecCiudad carga un codigo de ciudad en la variable correspondiente

System.***out***.println("Ciudad destino: ");

**ciudadDestino** = **man**.selecCiudad();

do{

System.***out***.print("Tarifa base: ");

**aux** = **br**.readLine();

if(**man**.validarTarifa(**aux**)){

**tarifaBase** = Double.parseDouble(**aux**);

**si** = true;

}else{

System.***out***.println("Fecha invalida.");

**si** = false;

}

}while(!**si**);

String data = num+"-"+**fechaVuelo**+"-"+**horaVuelo**+"-"+**ciudadOrigen**

+"-"+**ciudadDestino**+"-"+**tarifaBase**;

**man**.guardar("vuelo", data);

}

void **verVuelos**() throws IOException{//muestra los vuelos creados

System.***out***.print("Introduzca el numero del vuelos: ");

**aux** = **br**.readLine();

String[] datos = **man**.buscar("vuelo", **aux**);

//si la funcion buscar() devuelve un null significa que el archivo esta vacio

if(datos!=null){

System.***out***.println("\033[34m|------------------------------------|");

System.***out***.println("\033[34m|Vuelo #\033[31m"+datos[0]+"\033[34m\t\t\t |");

System.***out***.println("\033[34m|Con fecha: \033[31m"+datos[1]+"\033[34m - a las \033[31m"+datos[2]+"\033[34m |");

System.***out***.println("\033[34m|De \033[31m"+datos[3]+"\033[34m a \033[31m"+datos[4]+"\033[34m\t\t\t |");

System.***out***.println("\033[34m|Con una tatira base de \033[31m"+datos[5]+"\033[34m\t |");

System.***out***.println("\033[34m|------------------------------------|");

}else{

System.***out***.println("Vuelo no encontrado...!");

}

}

boolean **verificarVuelo**(String n){//verifica la existencia de un vuelo en le archivo vuelo.txt

String[] data = **man**.buscar("vuelo", n);

if(data!=null){

if(data[0].equals(n)){

return true;

}else{

return false;

}

}else{

return false;

}

}

String[]**infoVuelo**(String n){//envia un arreglo con la informacion de un buelo buscado

String[] data;

data = **man**.buscar("vuelo",n);

return data;

}

void **disponibilidad**(String numV){

/\*imprime en pantalla una matriz con la disposicion de los diferentes asientos

\* en un vuelo. Se incican en color negro los asientos disponibles, y en

\* color rojo los asientos ocupados\*/

int i,j,x;

for(x=1; x<=16; x++){

if(x<=9){

System.***out***.print("\033[32m |"+x+"| ");

}else{

System.***out***.print("\033[32m|"+x+"| ");

}

}

System.***out***.println();

System.***out***.println("\033[32m-------------------------------------------------------------------------------");

for (i=0;i<=5;i++){

if(i==3){

System.***out***.println();

}

for (j=0;j<=15;j++){

//funcion de ver disponibilidad

if(**man**.verDispon(j, i, numV)){

**dispon** = false;

}else{

**dispon** = true;

}

if(**dispon**){

**disp**--;

**color** = "\033[31m";

}else{

**color** = "\033[30m";

}

switch(i){

case 0 : System.***out***.print(**color**+" |A| "); break;

case 1 : System.***out***.print(**color**+" |B| "); break;

case 2 : System.***out***.print(**color**+" |C| "); break;

case 3 : System.***out***.print(**color**+" |D| "); break;

case 4 : System.***out***.print(**color**+" |E| "); break;

case 5 : System.***out***.print(**color**+" |F| "); break;

}

}

System.***out***.println();

}

System.***out***.println("\033[32m-------------------------------------------------------------------------------");

System.***out***.println("Lugares disponibles: \033[31m"+**disp**);

**disp** = 96;

}

}

|  |
| --- |
| Boleto.java |

/\*Gestion y venta de boletos\*/

package newairlineseller;

import java.io.BufferedReader;

import java.io.File;

import java.io.IOException;

import java.io.InputStreamReader;

public class **Boleto** {

String **nom**, **ape**, **ci**, **dir**, **telf**, **nVuel**, **fVuelo**, **hVuelo**, **nBoleto**,

**clase**, **ciudadOrigen**, **ciudadDestino**, **aux**, **asiento**;

boolean **si**;

double **tarifa**, **base**;

int **fil**, **col**;

Manejador **man** = new Manejador();

Vuelos **v** = new Vuelos();

Cliente **c** = new Cliente();

BufferedReader **br** = new BufferedReader(new InputStreamReader(System.***in***));

double **tarifaBase**(String nV){

//busca en el archivo vuelo la tarifa base correspondiente a un vuelo especifico

double b;

String[] vl = **man**.buscar("vuelo", nV);

b = Double.parseDouble(vl[5]);

return b;

}

void **crearBoleto**() throws IOException{

/\*inicia la recoleccion de datos para la cracion del boleto y su

\* correspondiente registro en archivo\*/

String nomArch = "boletos";

/\*se inicializa la variable para evitar que se cree un archivo con

\* nombre incorrecto\*/

**si** = false;

do{

System.***out***.print("Nuemro de vuelo a vender: ");

**aux** = **br**.readLine();

if(**v**.verificarVuelo(**aux**)){

**nVuel** = **aux**;

nomArch = "boletos"+**nVuel**;

/\*se crea un archivo con el nombre boleto y la referencia del numero de vuelo\*/

**man**.crear(nomArch);

**nBoleto** = Integer.toString(**man**.buscar(nomArch)+1);

System.***out***.println("Venta de boleto para el vuelo nº \033[31m"+**nVuel**+"\033[30m.");

System.***out***.println("\033[30mBoleto #\033[31m"+**nBoleto**+"\033[30m.");

System.***out***.print("Cedula: ");

**ci** = **br**.readLine();

String[] ver = **man**.buscar("cliente", **ci**);

if(ver!=null){

if(ver[0].equals(**ci**)){

System.***out***.println("\033[30mNombre:\t\t\033[31m"+ver[1]);

**nom** = ver[1];

System.***out***.println("\033[30mApellido:\t\033[31m"+ver[2]);

**ape** = ver[2];

System.***out***.println("\033[30mTelefono:\t\033[31m"+ver[3]);

**telf** = ver[3];

System.***out***.println("\033[30mDireccion:\t\033[31m"+ver[4]);

**dir** = ver[4];

System.***out***.println("\033[31mCliente encontrado!!!\033[30m");

**si** = true;

}else{

String[] datos = **c**.pedirDatos2();

**nom** = datos[0]; **ape** = datos[1]; **telf** = datos[2]; **dir** = datos[3];

String data = **ci**+"-"+**nom**+"-"+**ape**+"-"+**telf**+"-"+**dir**;

**man**.guardar("cliente", data);

**si** = true;

}

}else{

String[] datos = **c**.pedirDatos2();

**nom** = datos[0]; **ape** = datos[1]; **telf** = datos[2]; **dir** = datos[3];

String data = **ci**+"-"+**nom**+"-"+**ape**+"-"+**telf**+"-"+**dir**;

**man**.guardar("cliente", data);

**si** = true;

}

}else{

System.***out***.print("No se ha creado el vuelo.\n"

+ "Desea crear un nuevo vuelo? (s/n): ");

String aux1 = **br**.readLine();

if(aux1.equals("s")){

**v**.crearVuelo();

**si** = false;

}else{

**si** = true;

}

}

}while(!**si**);

**v**.disponibilidad(**nVuel**);

System.***out***.println("Asiento");

do{

System.***out***.print("Linea: ");

**aux** = **br**.readLine();

**fil** = Integer.parseInt(**aux**);

**tarifa** = tarifaBase(**nVuel**);

if((**fil**>0)&&(**fil**<17)){

if(**fil**<3){

**tarifa** += **tarifa**\*0.25; **clase** = "A";

**si** = true;

}

if((**fil**>=3)&&(**fil**<8)){

**tarifa** += **tarifa**\*0.15; **clase** = "B";

**si** = true;

}

if(**fil**>=8){

**tarifa** = tarifaBase(**nVuel**); **clase** = "C";

**si** = true;

}

}else{

System.***out***.println("Linea invalida");

**si** = false;

}

}while(!**si**);

do{

System.***out***.print("Letra: ");

**aux** = **br**.readLine();

**asiento** = **fil**+**aux**.toUpperCase();

switch(**aux**.toLowerCase()){

case "a" : **col** = 0; **si**=true; break;

case "b" : **col** = 1; **si**=true; break;

case "c" : **col** = 2; **si**=true; break;

case "d" : **col** = 3; **si**=true; break;

case "e" : **col** = 4; **si**=true; break;

case "f" : **col** = 5; **si**=true; break;

default:

System.***out***.println("Letra invalida.");

**si** = false; break;

}

}while(!**si**);

String data = **nBoleto**+"-"+**nVuel**+"-"+**ci**+"-"+(**fil**-1)+"-"+**col**+"-"+**clase**+"-"+**tarifa**;

**man**.guardar(nomArch, data);

imprimirBoleto(**nBoleto**, **nVuel**, **ci**);

}

void **imprimirBoleto**(String Bol, String Vuel, String CI){

String[] vuelo= **man**.buscar("vuelo", Vuel);

String[] cliente= **man**.buscar("cliente", CI);

System.***out***.println("\033[31m-----------------------------------------------------------");

System.***out***.println("\033[31mBoleto\t#\033[30m"+Bol+".\t\t\t\033[31mVuelo\t#\033[30m"+**nVuel**);

System.***out***.println("\033[31mNombre:\t\033[30m"+cliente[2]+", "+cliente[1]+"\t\t\033[31mFecha de Vuelo:\t\033[30m"+vuelo[1]);

System.***out***.println("\033[31mCedula:\t\033[30m"+cliente[0]+"\t\t\033[31mHora de Vuelo:\t\033[30m"+vuelo[2]+".");

System.***out***.println("\033[31mClase:\t\033[30m"+**clase**+"\t\033[31mAsiento:\033[30m"+**asiento**+"\t\033[31mPrecio:\t\033[30m"+**tarifa**);

System.***out***.println("\033[31mCiudad origen:\033[30m"+vuelo[3]+"\t\033[31mCiudad destino:\t\033[30m"+vuelo[4]);

System.***out***.println("\033[31m-----------------------------------------------------------");

}

void **imprimirBoleto**() throws IOException{

int i = 1;

System.***out***.print("Introduzca numero de vuelo a consultar: ");

String n = **br**.readLine();

String[] vuelo= **man**.buscar("vuelo", n);

while(true){

if(vuelo==null){

System.***out***.println("\033[31m-----------------------------------------------------------");

System.***out***.println("El vuelo #"+n+" no existe.");

break;

}

String[] boleto = **man**.buscar("boletos"+n, Integer.toString(i));

i++;

if(boleto!=null){

String[] cliente= **man**.buscar("cliente", boleto[2]);

String aux1 = "";

switch(boleto[4]){

case "0" : aux1 = "A"; **si**=true; break;

case "1" : aux1 = "B"; **si**=true; break;

case "2" : aux1 = "C"; **si**=true; break;

case "3" : aux1 = "D"; **si**=true; break;

case "4" : aux1 = "E"; **si**=true; break;

case "5" : aux1 = "F"; **si**=true; break;

}

String lugar = (Integer.parseInt(boleto[3])+1)+aux1+"";

System.***out***.println("\033[31m-----------------------------------------------------------");

System.***out***.println("\033[31mBoleto\t#\033[30m"+boleto[1]+".\t\t\t\033[31mVuelo\t#\033[30m"+n);

System.***out***.println("\033[31mNombre:\t\033[30m"+cliente[2]+", "+cliente[1]+"\t\t\033[31mFecha de Vuelo:\t\033[30m"+vuelo[1]);

System.***out***.println("\033[31mCedula:\t\033[30m"+cliente[0]+"\t\t\033[31mHora de Vuelo:\t\033[30m"+vuelo[2]+".");

System.***out***.println("\033[31mClase:\t\033[30m"+boleto[5]+"\t\033[31mAsiento:\033[30m"+lugar+"\t\033[31mPrecio:\t\033[30m"+boleto[6]);

System.***out***.println("\033[31mCiudad origen:\033[30m"+vuelo[3]+"\t\033[31mCiudad destino:\t\033[30m"+vuelo[4]);

}else{

break;

}

}

System.***out***.println("\033[31m-----------------------------------------------------------");

}

}

|  |
| --- |
| Reportes.java |

/\*Generacion de reportes de ventas de voletos\*/

package newairlineseller;

import java.io.File;

import java.io.IOException;

public class **Reportes** {

Manejador **man** = new Manejador();

double[] **sumarMontos**() throws IOException{

double[] monto = {0,0,0,0,0,0,0,0}, datos;

int i=1;

while(new File("boletos"+i+".txt").exists()){

datos = **man**.buscarMontos(i);

monto[0] += datos[0];//total general

monto[1] += datos[1];//total calse A

monto[2] += datos[2];//total calse B

monto[3] += datos[3];//total calse C

monto[4] += datos[4];//total boletos

monto[5] += datos[5];//total boletos A

monto[6] += datos[6];//total boletos B

monto[7] += datos[7];//total boletos C

i++;

}

return monto;

}

void **imprimirTotal**() throws IOException{

double[] total = sumarMontos();

System.***out***.println("\033[30mEl monto total recaudado en \033[31m"+(int)total[4]

+"\033[30m boletos vendidos es: \033[31m"+total[0]+"\033[30m");

System.***out***.println("\033[30mEl monto total recaudado en la clase A con \033[31m"

+(int)total[5]+"\033[30m boletos es: \033[31m"+total[1]+"\033[30m");

System.***out***.println("\033[30mEl monto total recaudado en la clase B con \033[31m"

+(int)total[6]+"\033[30m boletos es: \033[31m"+total[2]+"\033[30m");

System.***out***.println("\033[30mEl monto total recaudado en la clase C con \033[31m"

+(int)total[7]+"\033[30m boletos es: \033[31m"+total[3]+"\033[30m");

}

void **vueloVendidos**() throws IOException{

double[] datos;

int i = 1;

int v = **man**.buscar("vuelo");

System.***out***.println("\033[36m--------------------------------------\033[30m");

System.***out***.println("Existen \033[31m"+v+"\033[30m vuelos creados.");

while (new File("boletos"+i+".txt").exists()){

datos = **man**.buscarMontos(i);

System.***out***.println("\033[36m--------------------------------------\033[30m");

System.***out***.println("Para el vuelo #\033[31m"+i+"\033[30m, se vendieron:");

System.***out***.println("Clase A \033[31m"+(int)datos[5]+"\033[30m boletos, que suman : \033[31m"+datos[1]+"\033[30m Bs.");

System.***out***.println("Clase B \033[31m"+(int)datos[6]+"\033[30m boletos, que suman : \033[31m"+datos[2]+"\033[30m Bs.");

System.***out***.println("Clase C \033[31m"+(int)datos[7]+"\033[30m boletos, que suman : \033[31m"+datos[3]+"\033[30m Bs.");

i++;

}

System.***out***.println("\033[36m--------------------------------------\033[30m");

}

}