

Un determinado aeropuerto nos ha pedido que informaticemos parte de su gestión de compañías aéreas, vuelos y pasajeros. Para ello el programa debe llevar un registro de todas las compañías y vuelos existentes así como de los pasajeros que han solicitado un vuelo concreto.

El programa deberá permitir desde menú:

- Listar las compañías aéreas y los vuelos que realizan.
- Listar los pasajeros de un determinado vuelo
- Dar bajas de vuelos
- Reubicar pasajeros; esto es, mover todos los pasajeros de un vuelo origen a otro vuelo destino.
- Cargar y grabar ficheros con la información de compañías, vuelos y pasajeros.

Deberá realizarse un programa Java con arreglo a los siguientes requisitos:

1. Información a tratar

- Compañías: Las compañías que operan en el aeropuerto. De cada compañía se guarda la siguiente información:
 - Nombre (String). Actúa como identificador
 - Destino. String Almacena los diferentes destinos, nombres de ciudades a las que llegan los vuelos de la compañía.
 - Vuelos. Lista de vuelos que realiza la compañía
- Vuelos: Los vuelos de salida existentes en el aeropuerto. De cada vuelo se guarda la siguiente información:
 - Código (String). Actúa como identificador
 - Compañía. String
 - Destino. String

- HoraSalida. String
 - HoraLlegada. String
 - Puerta. Char
 - Pasajeros. Lista de pasajeros que embarcan en el vuelo.
- Pasajeros: Los pasajeros asignados a un vuelo concreto. De cada pasajero se guarda la siguiente información:
 - Código (String). Actúa como identificador
 - Dni. String
 - Nombre. String
 - Dirección. String
 - Teléfono. String
 - Clase. String. Almacena el tipo de billete: Turista, Bussines, ...

2. Opciones de menú:

- Listar compañías y vuelos
 - Se deben mostrar los datos de todas las compañías, una línea por cada compañía. También deberá pedir por teclado el nombre de una compañía y mostrar los datos de los vuelos que realiza dicha compañía (una línea por cada vuelo).
- Listar todos los vuelos
 - Ordenados por hora de salida en orden ascendente.
 - Una línea por cada vuelo en la que aparecerán todos sus datos.
- Listar los pasajeros de un vuelo

- Se pedirá por teclado el código de vuelo y se mostrará en pantalla una línea por cada pasajero en la que aparecerán todos sus datos.
- Eliminar un vuelo
 - Se pide por teclado el código de vuelo
- Eliminar los pasajeros de un vuelo
 - Se pide por teclado el código de vuelo origen y el código de vuelo destino.
 - Los pasajeros del vuelo origen se debe asignar a otro vuelo (vuelo destino).

3. Situaciones de error a controlar:

- Se considerarán errores no recuperables los problemas genéricos de entrada/salida y los de ficheros no existentes. El programa terminará de forma ordenada y mostrará un mensaje explicativo al usuario.
- Son situaciones gestionables las que provengan de incoherencias en los datos suministrados por el usuario o cargados desde el fichero inicial. El programa no debe terminar y se debe informar al usuario. Se deben controlar como mínimo las siguientes situaciones:
 - Buscar una compañía que no existe
 - Borrado de vuelo que no existe.

4. Carga de datos desde fichero

- El programa al iniciarse cargará los datos desde fichero
- El fichero estará en formato texto, organizado en líneas (registros)
- Cada registro contiene información relativa a una compañía, un vuelo o un pasajero.

- Cada línea está compuesta por campos separados por el carácter tabulador ('\t'). Se usarán estos formatos de registro:

Registro de compañía:

1<nombre><destinos>

Registro de vuelo:

2<código><compañía><destino><horaSalida><horaLlegada><puerta>

Registro de pasajero:

3<código><dni><nombre><dirección><teléfono><clase>

- En el fichero de entrada los registros aparecen en este orden.
- El fichero puede contener líneas en blanco o mal formadas. El programa debe descartar las líneas mal formadas o en blanco pero debe cargar las siguientes
- No se debe informar al usuario de los errores pero deben quedar registrados en un fichero de log llamado log.txt.
- Son líneas mal formadas las que :
 - Son de tipo desconocido (no empiezan por 1, 2 o 3)
 - No tienen el número de campos correcto para el tipo de

5. Grabación de fichero de salida:

- Al finalizar el programa se grabarán de forma automática los datos contenidos en memoria.
 - Un fichero de texto que tendrá el mismo formato que el de entrada.

6. Requisitos de diseño:

- Debe haber una única clase que actúe de interfaz con el usuario y:
 - Será la única que podrá leer y escribir de teclado
 - No contendrá lógica de la aplicación

- Todas las situaciones de error deben controlarse utilizando excepciones. No es necesario crear un tipo de excepción para cada posible situación errónea, se pueden agrupar
- Los métodos encargados de ordenar listas deben devolver listas (no string) y las ordenaciones no alterarán las listas originales.
- La carga y grabación de ficheros debe realizarse desde una clase específica a tal fin
- Todos los incidentes producidos en la carga deben se informados en un fichero log.txt