



Disponemos de datos sobre los vuelos que salen de distintos aeropuertos. Se dispone en un fichero denominado “vuelos.csv” de la siguiente información para cada vuelo:

- destino: de tipo string.
- precio: de tipo real
- número plazas: de tipo entero.
- número pasajeros: de tipo entero
- código: de tipo string.
- fecha salida: de tipo date.
- duración (en minutos): de tipo entero.
- hora salida: de tipo time.
- velocidad media (km/h): de tipo real.
- escalas: de tipo string que representa una lista de ciudades.
- económico: de tipo string que representa un tipo bool

Un ejemplo, de una línea del fichero es la siguiente:

Madrid ; 52,60;300;250;IBE-123; 05/07/2021 ;40;17:05:04;759.84; La Coruña-Bilbao-Valencia-Mallorca; S

Para manejar los datos se usará la siguiente namedtuple

Vuelo=namedtuple("vuelo", "destino,precio,num_plazas,num_pasajeros,código,fecha,duración,hora,velocidad,escalas,económico")

1. Cree un proyecto Python **T10_Vuelo**.
2. Cree dentro del proyecto tres carpetas **data**, **doc** y **src**.
3. En **doc** copie este enunciado
4. En **data** copie el fichero “**vuelos.csv**”
5. En **src** cree:
 - a. Un módulo **vuelos.py**, e implemente las siguientes funciones:
 - **lee_vuelos**, que reciba la ruta de un fichero en formato CSV codificado en UTF-8 y devuelva una lista de tuplas de tipo **Vuelo** con todos los registros del fichero.
 - b. Cree un módulo **vuelos_test.py** e implemente en él una prueba de la función anterior mediante la siguiente organización:
 - **test_lee_vuelo**:
 - Visualice el número de registros leídos
 - Visualice los tres primeros registros leídos
 - Visualice los tres últimos registros leídos.

Añada al **vuelos.py** las siguientes funciones y a **vuelos_test.py** las instrucciones para probarlos.

1. **filtra vuelos a**: Que, recibiendo una lista de tipo Vuelo y un destino, devuelva una lista de tipo Vuelo con los vuelos al destino dado.
2. **vuelos mas velocidad que**: Que, recibiendo una lista de tipo Vuelo y un número real que representa una velocidad media, devuelva una lista de tuplas con el destino, la fecha y la velocidad de los vuelos que superan la velocidad dada.



3. **todos vuelos mas velocidad que**: Que, recibiendo una lista de tipo Vuelo y un número real que representa una velocidad media, devuelva si todos los vuelos viajan a una velocidad media superior a la dada.
4. **vuelos más velocidad**: Que, recibiendo una lista de tipo Vuelo devuelva una tupla de tipo Vuelo con mayor velocidad media.
5. **vuelos por horario**: Que, recibiendo una lista de tipo Vuelo y un mes devuelva una lista de tuplas ordenada por las horas de salida con el destino, el precio, la fecha de salida y hora de salida de los vuelos cuyo mes de la fecha de salida coincide con el mes dado.
6. **distintas escalas**: Que, recibiendo una lista de tipo Vuelo devuelva una lista ordenada alfabéticamente de las distintas escalas por las que pasan los vuelos. Se considera escala también el aeropuerto de destino.
7. **vuelos con escalas en**: Que, recibiendo una lista de tipo Vuelo y una ciudad de escala devuelva una lista de tuplas con el destino, el precio y el número de plazas libres, ordenada por precios de los vuelos que hacen escala en la ciudad dada como parámetro.
8. **número de vuelo por destino**: Que, recibiendo una lista de tipo Vuelo devuelva un diccionario que a cada destino le haga corresponder el número de vuelos que tienen ese destino.

En el test se pide que:

- a) Se visualice diccionario directamente con un print
- b) Se visualice cada pareja clave-valor una debajo de otra
- c) Pida por teclado un destino y devuelva "Hay vuelo" o "No hay vuelo" según el destino esté o no en el diccionario

9. **suma de pasajeros por fechas**: Que, recibiendo una lista de tipo Vuelo devuelva un diccionario que a cada fecha de salida le haga corresponder la suma de pasajeros que vuelan en las distintas fechas.

En el test se pide que:

- a) Se visualice diccionario directamente con un print
- b) Se visualice cada pareja clave-valor una debajo de otra
- c) Pida por teclado una fecha y devuelva el número de pasajeros o cero (0) según haya o no vuelos en esa fecha

10. **lista destinos por compañía**: Que, recibiendo una lista de tipo Vuelo devuelva un diccionario que a cada compañía (la compañía son los tres primeros caracteres del código), le haga corresponder una lista con los destinos de las respectivas compañías.

En el test se pide que se visualice cada pareja clave-valor una debajo de otra.

11. **vuelos entre fechas**: Que, recibiendo una lista de tipo Vuelo y dos fechas devuelva una lista de vuelos, ordenado por fecha, con el destino, el precio y las escalas entre dichas fechas. Si alguna de ellas, o ambas, toman el valor *None* no se tendrá en cuenta en el filtro la/s fecha/s que tomen dicho valor.