



Se dispone de datos sobre desastres aéreos. Los datos están almacenados en el archivo `mayday.csv` del que se muestra un extracto con algunos desastres conocidos a continuación:

```
Date;Time;Location;Operator;Flight;Route;Type;Survivors;Fatalities;Ground;Operation
27/03/1977;17:07;Tenerife, Canary Islands;Pan American World Airways / KLM;1736/4805;Los Angeles -
Nueva York - Las Palmas / Amsterdam - Las Palmas;Boeing B-747-121 / Boeing B-747-
206B;61;583;0;Airport
25/07/2000;16:44;Gonesse, France;Air France;4590;Paris - New York City;Aerospatiale BAe
Concorde;0;109;4;Taking-off
```

Cada línea del archivo CSV incluye la siguiente información sobre un desastre aéreo: la **fecha**, la **hora**, la **localización**, los **operadores** implicados, los **códigos de los vuelos** implicados, las **rutas de los vuelos** implicados, los **modelos de aeronaves** que realizaban las rutas, el **número de supervivientes**, el **número de fallecidos**, el **número de fallecidos en tierra** y en qué **momento operativo del vuelo** ocurrió el desastre ("Airport", "Taking-off", "Climbing", "En route", "Approach", "Landing"). En el caso de que hubiese varios vuelos implicados, los datos de cada uno de los campos están separados por **" / "**.

Por ejemplo, en el CSV se puede observar:

- La primera línea es una línea de cabecera.
- En la segunda línea, se indica que el **27/03/1977**, a las **17:07**, en el aeropuerto de **Tenerife**, los vuelos operados por **PANAM** y **KLM**, con códigos **1736** y **4805**, rutas **Los Ángeles-Nueva York-Las Palmas** y **Amsterdam-Las Palmas**, operados con los modelos **Boeing B747-121** y **Boeing B747-206B**, respectivamente, colisionaron en tierra ("**Airport**") dejando **61 supervivientes**, **583 fallecidos** y **0 fallecidos en tierra**.
- La tercera línea indica que el **25/07/2000**, a las **16:44**, en el pueblo de **Gonesse**, el vuelo operado por **Air France**, con código **4590**, que cubría la ruta **París-Nueva York** y era operado por un **Concorde**, se estrelló durante la operación de despegue ("**Taking-off**"), dejando **0 supervivientes**, **109 fallecidos** más otros **4 fallecidos en tierra**.

Nótese que no siempre está disponible la hora del desastre. Si hay varios vuelos implicados, el carácter separador ("**/**") siempre aparece para separar los campos.

PROYECTO QUE SE LE REQUIERE IMPLEMENTAR

Se le facilita un proyecto Python con las siguientes carpetas:

- **data**: en la que se ubica el fichero `mayday.csv`
- **src**: que está vacía y en la que debe implementar los módulos Python `mayday.py` y `mayday_test.py`

Para realizar los ejercicios que se solicitan **utilice obligatoriamente** las definiciones de los *NamedTuple* que se indican a continuación:



```
from typing import NamedTuple
from datetime import datetime, date, time

Vuelo = NamedTuple("Vuelo",
    [("operador", str), # Compañía aérea que operaba el vuelo (opcional)
    ("codigo", str),   # Código de vuelo (opcional)
    ("ruta", str),     # Ruta del vuelo (opcional)
    ("modelo", str)]) # Modelo de avión que operaba el vuelo (opcional)

Desastre = NamedTuple("Desastre",
    [("fecha", date), # Fecha del desastre aéreo
    ("hora", time | None), # Hora del desastre (opcional)
    ("localizacion", str), # Localización del desastre
    ("supervivientes", int), # Supervivientes
    ("fallecidos", int), # Fallecidos
    ("fallecidos_en_tierra", int), # Fallecidos en tierra (no eran pasajeros del vuelo)
    ("operacion", str), # Momento operativo del vuelo cuando ocurrió el desastre
    ("vuelos", list[Vuelo])]) # Vuelos implicados en el desastre
```

Se pide implementar las funciones de los ejercicios que figuran a continuación en el módulo **mayday.py** y sus correspondientes **métodos de test con los parámetros adecuados** en el módulo **mayday_test.py**. Tenga en cuenta que se pueden definir funciones auxiliares cuando se considere necesario. Las puntuaciones están indicadas para cada ejercicio, y los tests solo se puntuarán si las funciones de test tienen los parámetros adecuados, se pueden ejecutar y el resultado coincide con el esperado.

EJERCICIO 1 - (1,75 puntos + 0,25 test)

lee_desastres: dada una cadena de texto con la ruta de un fichero CSV, devuelve una lista de tuplas de tipo Desastre con la información contenida en el fichero.

```
def lee_desastres(ruta:str)->list[Desastre]:
```

Para realizar la función anterior, ayúdese de las siguientes funciones auxiliares:

- **parsea_fecha**: dada una cadena con fecha en formato "%d/%m/%Y", devolver la fecha correspondiente.

```
def parsea_fecha(fecha:str) -> date:
```

- **parsea_hora**: dada una cadena con hora en formato "%H:%M", devolver la hora correspondiente.

```
def parsea_hora(hora:str) -> time:
```



- **parsea_vuelos:** dadas las cadenas operadores, códigos, rutas y modelos que tienen información agrupada de los vuelos implicados en el desastre aéreo, debe devolver una lista de tuplas de tipo Vuelo individuales. Por ejemplo, si nos dan las cadenas:

```
operadores = "Military Soviet Air Force / Aeroflot"
codigos     = "Unavailable / 831"
rutas       = "Kaunas - Ryazan / Moscow - Gomel"
modelos     = "Antonov An-12 / Ilyshin IL-14"
```

La función debería devolver la lista:

```
[
    Vuelo(operador='Military Soviet Air Force', codigo='Unavailable', ruta='Kaunas - Ryazan',
          modelo='Antonov An-12'),
    Vuelo(operador='Aeroflot', codigo='831', ruta='Moscow - Gomel', modelo='Ilyshin IL-14')
]
```

Y si nos dan las cadenas:

```
operadores = "Air Vietnam (South Vietnam) / U.S. Air Force"
codigos     = "Unavailable/Unavailable"
rutas       = "Pleiku - Da Nang / Unavailable"
modelos     = "Douglas DC-4 / USAF F-4E"
```

La lista de vuelos debe ser:

```
[
    Vuelo(operador='Air Vietnam (South Vietnam)', codigo='Unavailable', ruta='Pleiku - Da Nang',
          modelo='Douglas DC-4'),
    Vuelo(operador='U.S. Air Force', codigo='Unavailable', ruta='Unavailable', modelo='USAF F-4E')
]
```

Si no aparece el carácter separador ("/") se trata de un desastre aéreo que solo involucra a un vuelo, entonces la lista contiene una sola tupla Vuelo, como puede verse en los resultados esperados.

```
def parsea_vuelos(operadores:str,\
                  codigos:str,\
                  rutas:str,\
                  modelos:str) -> list[Vuelo]:
```

Resultados esperados:

```
Test leer_desastres =====
Número de desastres leídos: 456
Los dos primeros son:
    Desastre(fecha=datetime.date(1950, 3, 12), hora=datetime.time(14, 50), localizacion='Llandow Airport, Cardiff, Wales', supervivientes=3, fallecidos=80, fallecidos_en_tierra=0, operacion='Landing',
vuelos=[Vuelo(operador='Fairflight Ltd.', codigo='Unavailable', ruta='Llandow - Dublin', modelo='Avro 689 Tudor 5')])
    Desastre(fecha=datetime.date(1952, 3, 26), hora=None, localizacion='Tula Airport', supervivientes=0,
fallecidos=70, fallecidos_en_tierra=0, operacion='Landing', vuelos=[Vuelo(operador='Aeroflot',
codigo='Unavailable', ruta='Odessa - Tula - Moscow', modelo='Lisunov Li-2'), Vuelo(operador='Military
transport', codigo='Unavailable', ruta='Tula - Berlin, Germany', modelo='Ilyushin IL-12')])
Los dos últimos son:
    Desastre(fecha=datetime.date(2009, 5, 20), hora=datetime.time(6, 30), localizacion='Near Madiun, Indonesia', supervivientes=14, fallecidos=98, fallecidos_en_tierra=2, operacion='Approach',
vuelos=[Vuelo(operador='Military - Indonesian Air Force', codigo='Unavailable', ruta='Jakarta - Maduin',
modelo='Lockheed C-130 Hercules')])
    Desastre(fecha=datetime.date(2009, 6, 1), hora=datetime.time(0, 15), localizacion='AtlantiOcean, 570 miles northeast of Natal, Brazil', supervivientes=0, fallecidos=228, fallecidos_en_tierra=0, operacion='En route', vuelos=[Vuelo(operador='Air France', codigo='447', ruta='Rio de Janeiro - Paris', modelo='Airbus A330-203')])
```

**EJERCICIO 2 - (1,25 puntos + 0,25 test)**

desastres_con_fallecidos_en_tierra: dada una lista *desastres* de tuplas de tipo *Desastre* y un número que por defecto tomará el valor *None*, debe devolver una lista ordenada de tuplas (*localización, fecha y hora del desastre, fallecidos en tierra*) de aquellos desastres que resultaron con fallecidos en tierra, que no eran pasajeros de las aeronaves afectadas. Esta lista se presentará en orden descendente y el resultado se limitará a los peores “n” desastres, es decir, aquéllos en los que hubo más fallecidos en tierra.

```
def desastres_con_fallecidos_en_tierra(desastres:list[Desastre],n:int|None=None)\n    ->list[tuple[str,date,time,int]]:
```

Resultados esperados:

```
Test desastres_con_fallecidos_en_tierra =====
Los 5 desastres con más fallecidos en tierra son:
('World Trade Center, New York City, NY', datetime.date(2001, 9, 11), datetime.time(8, 45),
2750)
('The Pentagon, Arlington County, Virginia', datetime.date(2001, 9, 11), datetime.time(9, 39),
125)
('Kano, Nigeria', datetime.date(2002, 5, 4), datetime.time(13, 30), 75)
('Maracaibo, Zulia, Venezuela', datetime.date(1969, 3, 16), datetime.time(12, 0), 70)
('Near Havana, Cuba', datetime.date(1989, 9, 3), datetime.time(19, 0), 45)
```

EJERCICIO 3 - (1,75 puntos + 0,25 el test)

decada_mas_colisiones: dada una lista *desastres* de tuplas de tipo *Desastre*, debe devolver una tupla (*década, número de desastres con colisiones*) correspondiente a la década donde se produjeron mayor número de desastres que implicaron a 2 o más vuelos.

```
def decada_mas_colisiones(desastres:list[Desastre]) -> tuple[int,int]:
```

Por década se entiende los 10 años consecutivos que comienza por un año que es múltiplo de 10. Por ejemplo, 1970 representa a la década correspondiente a los años entre 1970 y 1979, ambos inclusivos.

Resultados esperados:

```
Test decada_mas_colisiones =====
La década de 1970 fue la peor, con 8 desastres con colisiones de aeronaves.
```

EJERCICIO 4 - (2,00 puntos + 0,25 el test)

mayor_periodo_sin_desastres: dada una lista de tuplas *desastres* de tipo *Desastre*, devuelve una tupla que incluye la fecha inicial y final, así como la duración, que corresponde al mayor período sin desastres, entre las fechas de inicio y fin de desastres registrados. El resultado puede estar filtrado por la operación que realizaba la aeronave cuando ocurrió el desastre. Puede tomar el valor por defecto *None*, en cuyo caso se obtendrá el mayor período sin registrar desastres aéreos para cualquier operación.

```
def mayor_periodo_sin_desastres(desastres:list[Desastre],\n    operacion:str|None=None)-> tuple[date, date, int]:
```

Nótese que solamente se tienen en cuenta las fechas de los desastres, ya que no siempre se tiene información de la hora en la que ocurrieron. Para calcular los días transcurridos, la propiedad *days* devuelve los días de diferencia entre dos fechas *f1* y *f2*: *(f1-f2).days*.



Resultados esperados:

```
Test mayor_periodo_sin_desastres =====
El mayor periodo sin desastres comienza el 1953-06-18, termina el 1956-06-20 y dura 1098 días.
El mayor periodo sin desastres durante la operación Taking-off comienza el 1953-06-18, termina el
1960-09-19 y dura 2650 días.
El mayor periodo sin desastres durante la operación Landing comienza el 1952-03-26, termina el
1959-02-03 y dura 2505 días.
```

EJERCICIO 5 - (2,00 puntos + 0,25 el test)

estadisticas_por_operacion: dada una lista *desastres* de tuplas de tipo **Desastre** y un límite mínimo de la tasa de supervivencia, que por defecto tomará el valor **None**, debe devolver un diccionario cuya clave será una operación y el valor asociado una tupla que incluya el número de desastres asociado a dicha operación, la media de supervivientes y la media de fallecidos.

Si se proporciona el límite de la tasa de supervivencia, el resultado solo incluirá aquellos desastres con una tasa de supervivencia menor o igual a dicho límite.

La tasa de supervivencia es la ratio entre fallecidos y pasajeros totales de la aeronave que ha sufrido el desastre, que es la suma de supervivientes y fallecidos, **SIN** incluir a los fallecidos en tierra.

```
def estadisticas_por_operacion(desastres: list[Desastre],\
                               limite_tasa_supervivencia:float|None=None) \
    ->dict[str,tuple[int,float,float]]:
```

Resultados esperados:

```
Test estadisticas_por_operacion =====
Las estadísticas por operación son:
('Landing', (94, 62.97872340425532, 68.85106382978724))
('Taking-off', (120, 40.06666666666667, 94.25833333333334))
('En route', (126, 46.15873015873016, 101))
('Approach', (90, 11.977777777777778, 106.95555555555555))
('Climbing', (10, 42.9, 123.3))
('Airport', (16, 151, 48.125))
Las estadísticas por operación con tasa de supervivencia menor a 0.15 son:
('Landing', (44, 3.7045454545454546, 98.5))
('Taking-off', (80, 1.7375, 122.0125))
('En route', (91, 0.4725274725274725, 129.45054945054946))
('Approach', (75, 1.72, 117.69333333333333))
('Climbing', (6, 0, 196.83333333333334))
('Airport', (1, 61, 583))
```