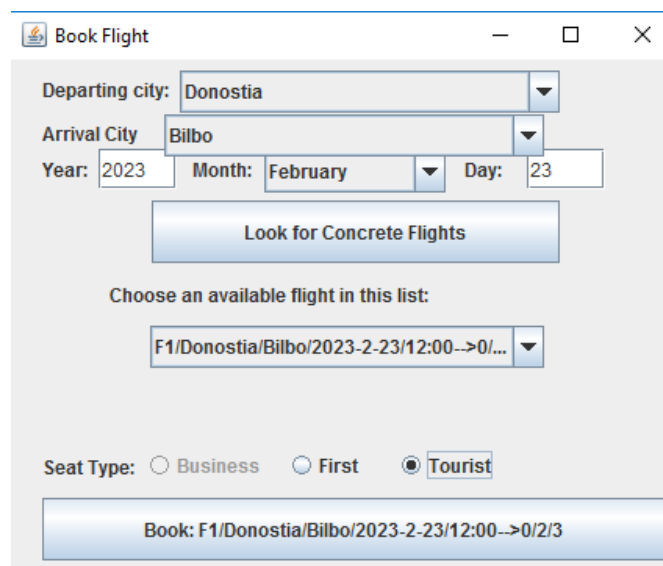


URGENTE: No hay persistencia en “RESERVAR VUELOS”

El siguiente “marrón” que tenéis que solucionar en Sinking Soft es el tema de que no hay persistencia en los objetos utilizados en el caso de uso “Reservar Vuelos”. Los vuelos concretos reservados durante una sesión NO se recuerdan en las siguientes sesiones.

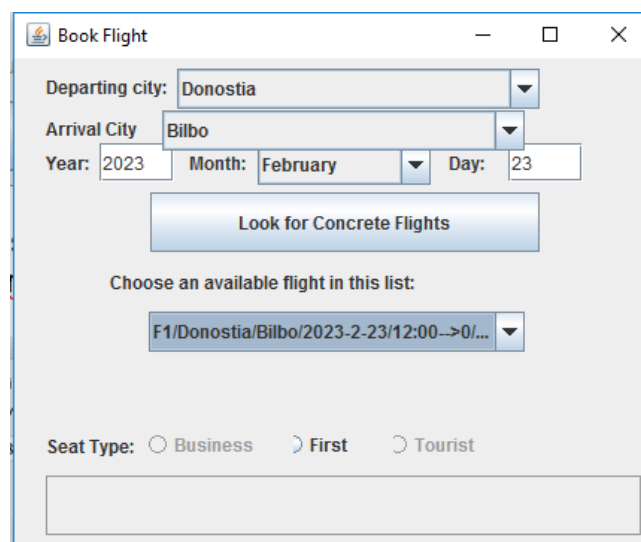
Si se ejecuta la clase de reserva de vuelos que habéis realizado después de arreglar el marrón sobre Swing, y se reservan los 3 vuelos de clase turista del vuelo de Donostia a Bilbao del 23/February/2023 a las 12:00, tras introducir los datos y pulsar sucesivamente los botones “Look for Concrete Flights” y “Book this concrete flight” (teniendo seleccionado dicho vuelo en la lista desplegable)



The screenshot shows a window titled "Book Flight" with the following elements:

- Departing city: Donostia (dropdown)
- Arrival City: Bilbo (dropdown)
- Year: 2023, Month: February, Day: 23
- Look for Concrete Flights button
- Choose an available flight in this list:
- F1/Donostia/Bilbo/2023-2-23/12:00-->0/... (dropdown)
- Seat Type: ☐ Business, ☐ First, ☒ Tourist
- Book: F1/Donostia/Bilbo/2023-2-23/12:00-->0/2/3 button

veremos que no será posible reservar un cuarto vuelo concreto, porque el botón de reserva ya no estará activado.



The screenshot shows the same "Book Flight" window, but with the following changes:

- The "Book: F1/Donostia/Bilbo/2023-2-23/12:00-->0/2/3" button is now disabled (grayed out).
- The "Seat Type" section now has ☐ Business, ☒ First, and ☐ Tourist.
- There is an empty rectangular box at the bottom of the window.



Sin embargo, si abrimos una nueva sesión (cerrando el JFrame FlightBooking) y volvemos a ejecutar la aplicación, veremos que nuevamente podemos volver a reservar el mismo vuelo.

Lo que ha sucedido es que el objeto de la clase ConcreteFlight desaparece de la memoria principal cuando se termina la aplicación, y cuando se vuelve a ejecutar la misma aplicación se vuelve a crear uno nuevo.

```
f1.addConcreteFlight("CF1-1",newDate(2023,1,23),0,2,3,"12:00");
```

Si se hubiera querido que “recordara” los valores anteriores, entonces habría que haber guardado en memoria secundaria dicho objeto (en un fichero o en una base de datos, por ejemplo) tras la primera ejecución y en la segunda ejecución habría que haberlo recuperado de memoria secundaria para seguir trabajando con el mismo objeto, esto es, habría que trabajar con OBJETOS PERSISTENTES en memoria.

Lo que se pide es lo siguiente:

- 1) Añadir un **atributo booleano** a la clase con la lógica del negocio FlightManager, de tal manera que si su valor es “**true**” se cree una nueva base de datos objectDB (borrando la anterior) y si es “**false**” abra la base de datos ya existente.

```
public static boolean initializeDB=true; // o false para trabajar con la BD anterior
```

- 2) Que al crear la nueva base de datos, los objetos de vuelos y vuelos concretos que hay en FlightManager se almacenen en la BD.
- 3) Que después de reservar un vuelo concreto, se actualice el número de billetes disponibles de ese vuelo concreto.

Nota importante: la reserva de vuelos concretos (actualización de objetos de la clase ConcreteFlight) debería solicitarse a la lógica del negocio y realizarse en la misma (no en la presentación FlightBooking)