

APELLIDOS:

NOMBRE:

DNI:

PARTE PRÁCTICA

Ejercicio 2 (6 Puntos)

Una gran empresa de eventos quiere mejorar el sistema que tiene para gestionar las reservas de los diferentes espacios para eventos que posee en una gran ciudad. Los usuarios de la aplicación pueden consultar los espacios que hay disponibles para eventos. El usuario simplemente indicará qué tipo de evento quiere celebrar, para cuántas personas y en qué fecha, la aplicación mostrará la lista de los espacios disponibles en los que se pueda celebrar el evento. **Sólo se pueden alquilar por días, una reserva diferente para cada día distinto.**

Cada espacio estará representado por un identificador que será el nombre asignado al espacio (por ejemplo, “Sala Caracol”, “La Quinta del Jarama”, etc.), además de la capacidad en cuanto a número de personas, el número de metros cuadrados, la dirección, y el coste que supone alquilar el espacio. El alquiler mínimo de cada espacio sea del tipo que sea, es de un día.

Los tipos de eventos que se pueden celebrar en los espacios de los que dispone la empresa son: bodas, celebraciones familiares, eventos corporativos, fiestas infantiles y fiestas de empresa.

Los diferentes espacios con los que cuenta la empresa actualmente son de diferente tipo: espacio para celebraciones infantiles, lofts (donde se pueden celebrar eventos corporativos y celebraciones familiares) y NaveChic (donde se pueden celebrar eventos corporativos, celebraciones familiares, fiestas de empresa y bodas).

Es importante tener en cuenta que la empresa puede llegar a ampliar su negocio con más espacios para eventos, o incluso con más tipos de eventos que ahora no tiene en su cartera de negocio. Es decir, hay que pensar en la extensibilidad del código a la hora de hacer el diseño para solucionar el ejercicio.

Se proporcionan diferentes clases que pueden resultar de utilidad para resolver el ejercicio: una clase que representa los diferentes tipos de eventos (`TipoEvento`), una clase que representa los diferentes “juegos” que puede tener un espacio para fiestas infantiles (`JuegoInfantil`), otra clase que representa un calendario anual (`CalendarioAnual`) y una clase de utilidades variadas (`Utils`).

Se recomienda leer el **enunciado completo** antes de empezar a responder cada apartado. De este modo se puede tener una visión global de cómo sería el programa final. **Hay que tener en cuenta que el código para responder un apartado es posible que deba extenderse o modificarse para responder a un apartado posterior.**

IMPORTANTE: Se debe hacer entrega de todo el código en un único archivo .zip. Y entregarlo a través del formulario habilitado para ello.

APELLIDOS:

NOMBRE:

DNI:

Se pide lo siguiente:

- a) [2.5 puntos] Implementar todo lo necesario para que cuando los dueños de la empresa de eventos compren y acondicionan un nuevo espacio, lo puedan incluir en el sistema. Hay que tener en cuenta que en un mismo espacio se pueden celebrar diferentes tipos de eventos, pero esta empresa tiene, para cada tipo de espacio asignado a priori los posibles eventos que se pueden celebrar, esto no varía. Por ello, cuando se da de alta un espacio en la aplicación, ya se sabe esta información y se debe tener en cuenta a la hora de almacenarlo.

Además, **es importante** saber que las búsquedas de espacios para eventos por parte de los usuarios de la aplicación deben ser eficientes en cuanto al tipo de evento (ver siguiente apartado) y que, para cada tipo de evento soportado, los espacios deben estar ordenados en orden creciente según sus metros cuadrados.

- b) [0.75 puntos] Implementar todo lo necesario para que el sistema pueda mostrar los espacios de eventos que tiene para celebrar una boda.

Un ejemplo de posible salida al hacer una búsqueda puede ser la siguiente:

```
Espacios disponibles para celebrar: BODA
NaveChic -- { metrosCuadrados=500, direccion='dir 6', nombre='Nave Chic 1', capacidad=300,
coste=0.0}
*****
HotelNaveChic -- { metrosCuadrados=750, direccion='dir 8', nombre='Nave Chic Hotel 1',
capacidad=400, coste=0.0}
*****
NaveChic -- { metrosCuadrados=800, direccion='dir 7', nombre='Nave Chic 2', capacidad=500,
coste=0.0}
*****
HotelNaveChic -- { metrosCuadrados=1050, direccion='dir 9', nombre='Nave Chic Hotel 2',
capacidad=400, coste=0.0}
*****
```

- c) [1.5 puntos] Implementar lo necesario para que el sistema pueda mostrar los espacios de eventos disponibles para un determinado tipo de evento, número de personas y fecha. La fecha se debe indicar en el formato DD/MM/AAAA, si no se proporciona una fecha en este formato, se deberá indicar de alguna forma que no es una fecha válida y se finaliza el proceso de búsqueda.

Las búsquedas tienen que ser lo más eficientes posible considerando como datos relevantes para la eficiencia en la búsqueda tanto el tipo de evento como la fecha.

APELLIDOS:

NOMBRE:

DNI:

La clase proporcionada `CalendarioAnual` puede facilitar la gestión de las fechas si estas se representan como un `String`. Esta clase se puede ampliar con todos los métodos que se quiera.

Un ejemplo de posible salida al hacer una búsqueda puede ser la siguiente:

Espacios disponibles que cumplen los criterios: FIESTA_INFANTIL para 60 personas, el 22/01/2023

Espacio Infantil -- { metrosCuadrados=50, direccion='dir 2', nombre='Esp.Infantil 2', capacidad=70, coste=90.0}

- d) [1.25 puntos] Implementar todo lo necesario para hacer la reserva de un determinado espacio, es decir, proporcionando la fecha y el nombre del espacio, marcarlo como reservado. **Se deja fuera de este ejercicio toda la gestión de asociar las reservas con un determinado cliente.**

Si se intenta reservar en una fecha en la que ese espacio ya está reservado, debe mostrarse un mensaje por pantalla indicando que ya está reservado. Si se intenta reservar un espacio cuyo nombre no existe para ningún espacio dado de alta en el sistema, entonces se informa de ello y se termina la ejecución.

Importante: el acceso al espacio concreto para tramitar la reserva debe ser eficiente.