

ENUNCIADO

Una tienda de venta de muebles desea implementar una aplicación donde registrar los encargos que recibe. Dicha tienda trabaja con dos tipos de muebles, aquellos ya prefabricados y los que se fabrican por encargo. La aplicación no va a llevar un registro de los muebles de la tienda, simplemente va a ir registrando los pedidos que se realizan. Un pedido sólo puede incluir muebles del mismo tipo, o prefabricados o por encargo, de forma que si un cliente desea adquirir muebles de distinto tipo deberá realizar dos pedidos. Un pedido deberá registrar la siguiente información: número de pedido que estará formado por la letra P seguido de 4 dígitos que irá asignando el sistema a medida que se vayan añadiendo nuevos pedidos, de forma que el primer pedido será el P0001, el siguiente P0002 y así sucesivamente, fecha del pedido que será la fecha actual, una descripción del pedido que será de tipo String, en dicha descripción es donde se detallará que es lo que incluye el pedido y precio total del pedido. La fecha será un objeto de la clase Fecha que deberemos implementar, esta clase tendrá tres atributos con el día el mes y el año y un constructor donde recibirá estos tres parámetros y un constructor sin parámetros que inicializará los atributos con la fecha actual. Para saber la fecha actual se hará uso de la clase Calendar de la siguiente forma:

```
Calendar cal= Calendar.getInstance();  
int dia = cal.get(Calendar.DAY_OF_MONTH);  
int mes = cal.get(Calendar.MONTH);  
int year= cal.get(Calendar.YEAR);
```

Los pedidos de muebles hechos a medida guardarán además un plazo de entrega calculado en días. Los pedidos a medida implementarán una interfaz que definirá el método fechaEntrega que devolverá un objeto de tipo Fecha que será la fecha de entrega calculada a partir de la fecha del pedido y el número de días que tarda en entregarse.

Para poder mostrar la información de los pedidos todas las clases implementarán un método toString que generará un String con el número del pedido, fecha del pedido, descripción, precio y, en el caso de los pedidos por encargo, el número de días y la fecha de entrega.

El programa principal deberá gestionar una única lista de pedidos que almacenará en un array de 20 elementos, para lo que mostrará un menú con las siguientes opciones:

1. **Nuevo pedido:** registrará un nuevo pedido. Para ello le preguntará al usuario el tipo de pedido (prefabricado o por encargo), su descripción, su precio y, en el caso de los pedidos por encargo, el número de días estimado para su entrega, y se creará el pedido añadiéndolo a la lista de pedidos. En caso de que el array esté lleno y no pueda añadirse el pedido, se lo indicará al usuario antes de pedirle ningún dato.
2. **Listar pedidos por encargo:** Mostrará por pantalla todos los pedidos por encargo con la información de cada uno.
3. **Listar todos los pedidos:** Mostrará un listado con todos los pedidos.
4. **Buscar pedido:** preguntará por el código de un pedido y, si éste existe, lo mostrará por pantalla. Si no existe lo indicará mediante un mensaje.
5. **Salir:** finaliza el programa.

Se pide:

- Diseñar el diagrama de clases para este sistema, agrupando elementos (atributos y métodos) comunes evitando duplicar elementos. Para ello se podrá añadir clases intermedias o abstractas si se considera oportuno. Se deberá entregar la representación de dicho diagrama por escrito.
- Implementar las clases descritas ajustándose al esquema diseñado en el apartado anterior. Se implementarán los métodos que se consideren necesarios y se valorará el correcto uso de los modificadores.

Para aprobar el examen será imprescindible que éste no tenga errores de compilación.

Criterios de calificación			
Apartado	Criterio	Max	Puntos
Compilación	Si el programa no compila la nota máxima del examen será de 2 puntos		
Diseño del diagrama de clases	Elección adecuada de las clases añadiendo aquellas necesarias para evitar duplicidad	1,25	
Implementación del diagrama de clases	Correspondencia del diagrama con su implementación	0,25	
	Correcto uso de los modificadores	0,25	
	Uso correcto de clases abstractas e interfaces	0,5	
	Implementación de los atributos y métodos necesarios para cumplir con las especificaciones descritas	0,75	
	Implementación y uso de atributos y métodos estáticos	0,5	
	Correcto funcionamiento de los métodos de las clases siguiendo las especificaciones del enunciado.	1,5	
Programa principal	Organización del programa principal usando los métodos necesarios para tener un código claro y legible	0,5	
	Se añade pedido correctamente	1,5	
	Se muestran los pedidos por encargo	1	
	Se muestran los pedidos	0,5	
	Se busca un pedido	1	
	Funcionamiento del menú	0,5	
Total		10	