



Una pequeña empresa de mensajería quiere implementar una aplicación para almacenar sus envíos. De un **envío** debe almacenar la siguiente información: **nombre del remitente, dirección completa del remitente, código postal del remitente (formado por 5 dígitos y puede empezar por 0), nombre del destinatario, dirección completa del destinatario y código postal del destinatario (formado por 5 dígitos y puede empezar por 0)**. (No se incluye más información para simplificar el programa)

La empresa desea almacenar dos tipos de envíos distintos: **envíos ordinarios** y **envíos con seguimiento**. Los envíos con seguimiento almacenarán también un **código de seguimiento** que será creado automáticamente cuando se cree el envío y que estará formado por las letras **PK** seguidas de un número de 5 dígitos que se generará automáticamente por el programa de forma que el primer pedido tendrá el número 00001 y se irá incrementando cada vez que se cree un pedido y a continuación los 5 dígitos del código postal del destinatario. Así, por ejemplo, un código de seguimiento podría ser **PK0000128040**. Para generar el código los envíos con seguimiento implementarán una interfaz llamada **ISeguimiento** con dos métodos: **generarCodigo** que generará el código de seguimiento a partir del código postal del destinatario y **mostrarSeguimiento** que mostrará un mensaje por pantalla con el texto *“Mostrando seguimiento del envío PK0000128040”*, donde se mostrará el código de seguimiento que corresponda.

Se implementarán únicamente los constructores con parámetros y los gets y sets que sean necesarios. Se implementarán métodos **toString** que devolverá un String con la información de un envío donde aparecerá toda la información incluido el número de seguimiento en el caso de los envíos con seguimiento.

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.
- Implementar un programa **Mensajería** que contenga un método *main*. Dicho programa creará un array de envíos de 50 elementos (un único array para todos ellos) y mostrará un menú con las siguientes opciones, cuya función se describe a continuación de cada una de ellas:
 1. **Nuevo envío**: si se elige esta opción se comprobará si hay espacio para nuevos envíos en el array y, si lo hay, pedirá al usuario todos los datos del envío y posteriormente a eso se preguntará si se desea que el envío se haga con seguimiento o no. Se añadirá el envío al array si se trata de un envío con seguimiento se mostrará el texto *“Se ha generado un envío con código de seguimiento PX0000128040”* sustituyendo el código por el que corresponda. Si no hay espacio en el array se mostrará un mensaje indicándolo.
 2. **Envío entregado**: se pedirá un código de seguimiento y, si este se encuentra en el array, lo elimina dejando su posición a null. Si no se encuentra muestra un mensaje indicándolo.

3. **Listado de envíos:** mostrará un listado con la información de todos los envíos.
4. **Mostrar seguimiento:** pedirá un código de envío y si el envío se encuentra en el array y es un envío con seguimiento, llamará a su método *mostrarSeguimiento*.
5. **Salir:** el programa finalizará al seleccionar esta opción.

A no ser que se elija la opción salir después de cada acción se volverá a mostrar el menú. Cualquier decisión que se tome por no estar especificada en el enunciado y que no modifique ninguna de las especificaciones deberá ser explicada razonadamente.

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



NOMBRE: _____

DIAGRAMA DE CLASES

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 y correcta relación entre las mismas	1,5	
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	0,5	
	Implementación y uso de atributos y métodos estáticos	0,5	
	Correcto funcionamiento de los métodos de las clases siguiendo las especificaciones del enunciado.	1,25	
Programa principal	Organización del programa principal usando los métodos necesarios para tener un código claro y legible	0,5	
	Se añade un envío correctamente al array	1,25	
	Se elimina un envío correctamente del array	1,25	
	Se muestra correctamente el listado de envíos	0,5	
	Se muestra correctamente la información de seguimiento de un envío	1,25	
	Funcionamiento del menú	0,5	
Total		10	