EJERCICIO REPASO HERENCIA Y INTERFACES

Haremos un ejercicio con los siguientes elementos:

Paquete herencia:

Tendremos una clase llamada **Item** con los atributos nombre (String) e id (int).

Tendremos una clase llamada **Ropa** que hereda de Item, con un atributo propio llamado talla (String).

Otra clase llamada **Zapatos** que también hereda de Item sus atributos, con un atributo propio llamado número (int).

Paquete interfaz:

Interfaz **IAlmacen** con los métodos**:**

```
public boolean almacenar(Object item);
public Object recuperar(int identificador);
```

Dos clases de nombre **Armario** y **Caja** que implementan el interfaz. En Armario hay un ArrayList de objetos tipo Ropa y en Caja un ArrayList de objetos tipo Zapatos. Ambas clases tendrán un método toString para poder mostrar los datos del ArrayList de una manera apropiada. Puedes añadir algún método propio si quieres. Los ArrayList son private.

Paquete main:

En el programa principal se crea un Array de elementos de tipo IAlmacen en el que guardaremos 3 elementos: un Armario, una Caja y otro Armario.

A continuación, procederemos a rellenar el array con elementos que se piden al usuario de la manera que se describe a continuación:

En cada contenedor de tipo IAlmacen del array, ya sea Armario o Caja, se preguntará al usuario si lo que quiere guardar es Ropa o Zapatos (aun sabiendo que uno de los dos no es correcto) y se piden los datos para crear el objeto. Esto lo haremos para practicar captura de excepciones, al detectar el posible error de introducir zapatos en un armario o ropa en una caja. Se debe capturar la excepción **ClassCastException** dentro del método

almacenar() con el fin de mostrar el error correspondiente y no proceder a almacenar el objeto en ese caso.

Es responsabilidad del usuario que los identificadores sean únicos.

Se proporciona el ejecutable del programa que tienes que desarrollar para que veas el funcionamiento.

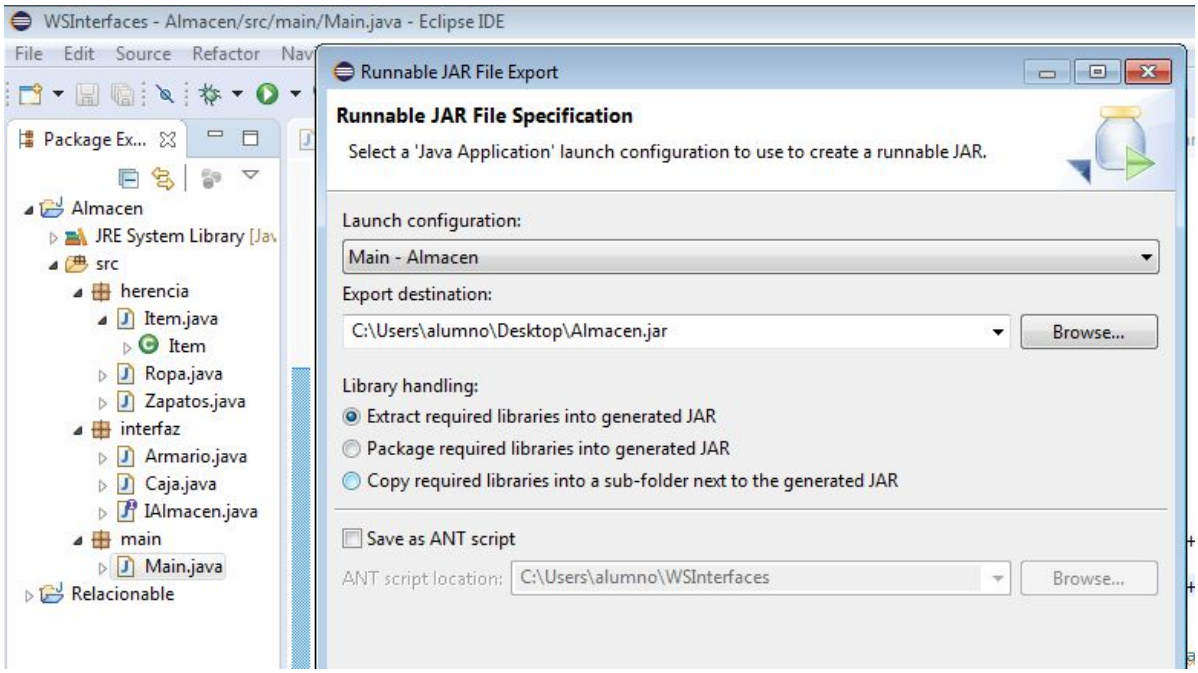
Como ves, la interfaz con el usuario se hace con la clase JOptionPane del paquete Swing de Java.

Cómo crear un ejecutable.

De paso que hacemos este ejercicio, os voy a indicar los pasos para que creéis un .exe (solo vale para Windows, claro)

Paso 1

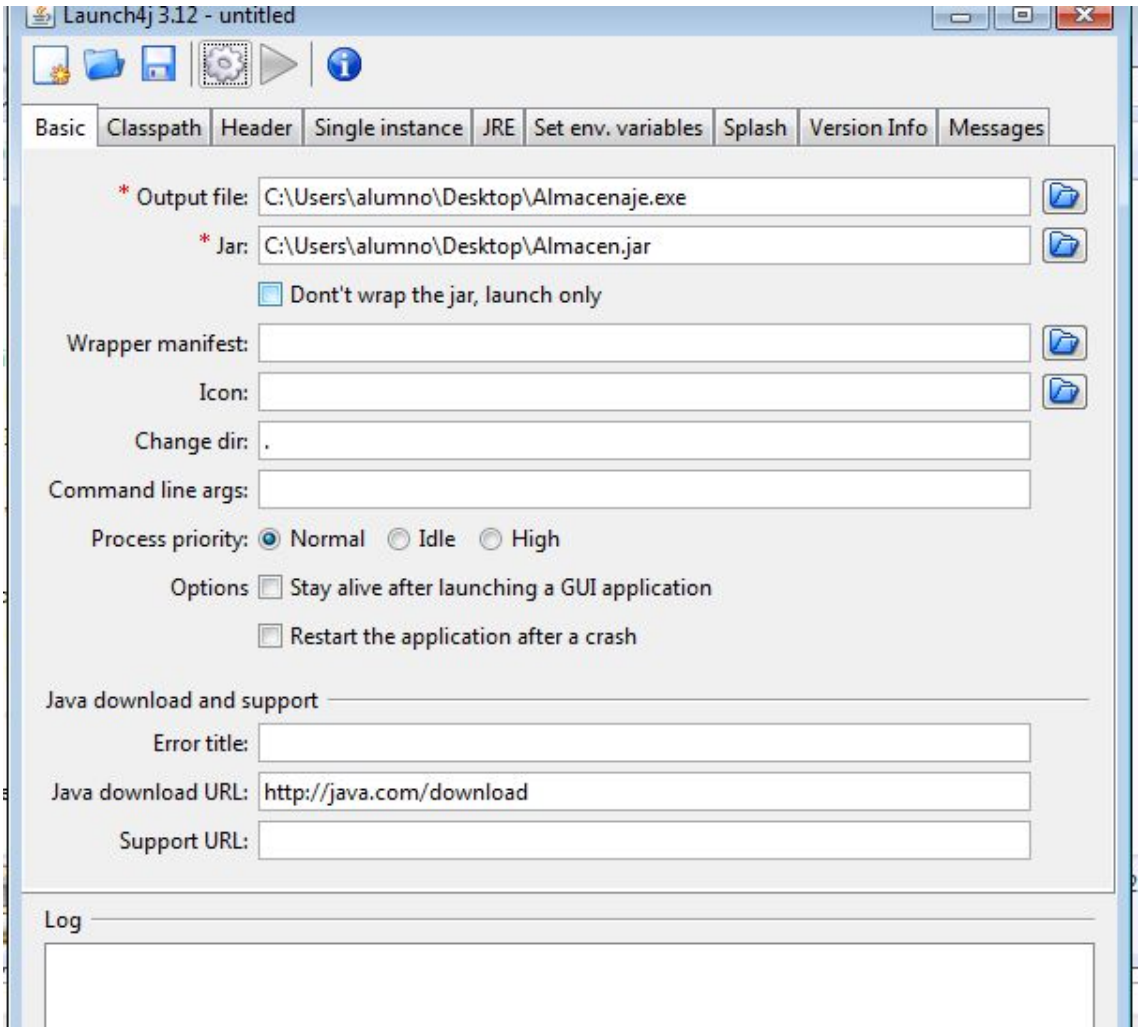
Exportamos a un jar ejecutable nuestro proyecto, indicando cuál es el main:



Paso 2

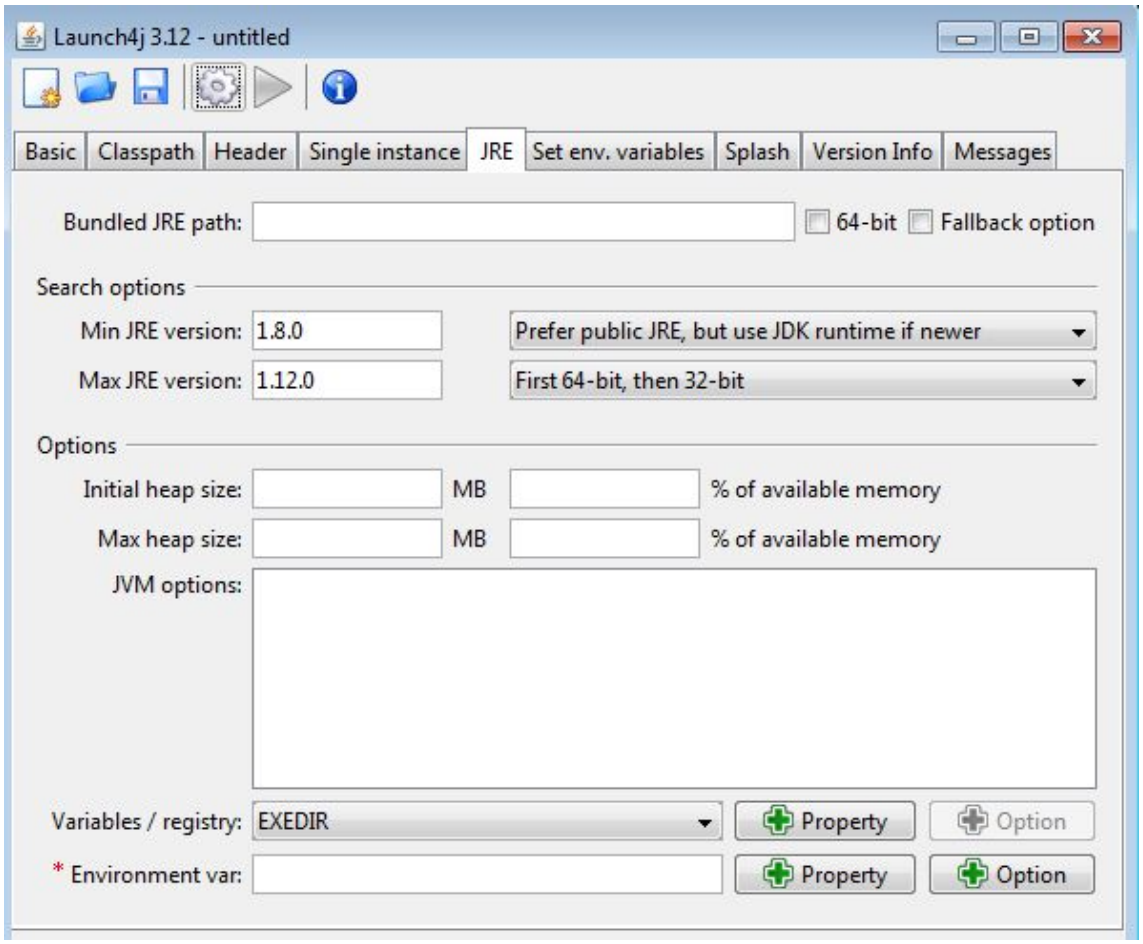
Nos instalamos la aplicación **launch4j-3.12-win32.exe** y la arrancamos**:**

Indicamos el nombre del ejecutable que queremos crear y dónde, así como dónde está el .jar ejecutable que hemos creado.



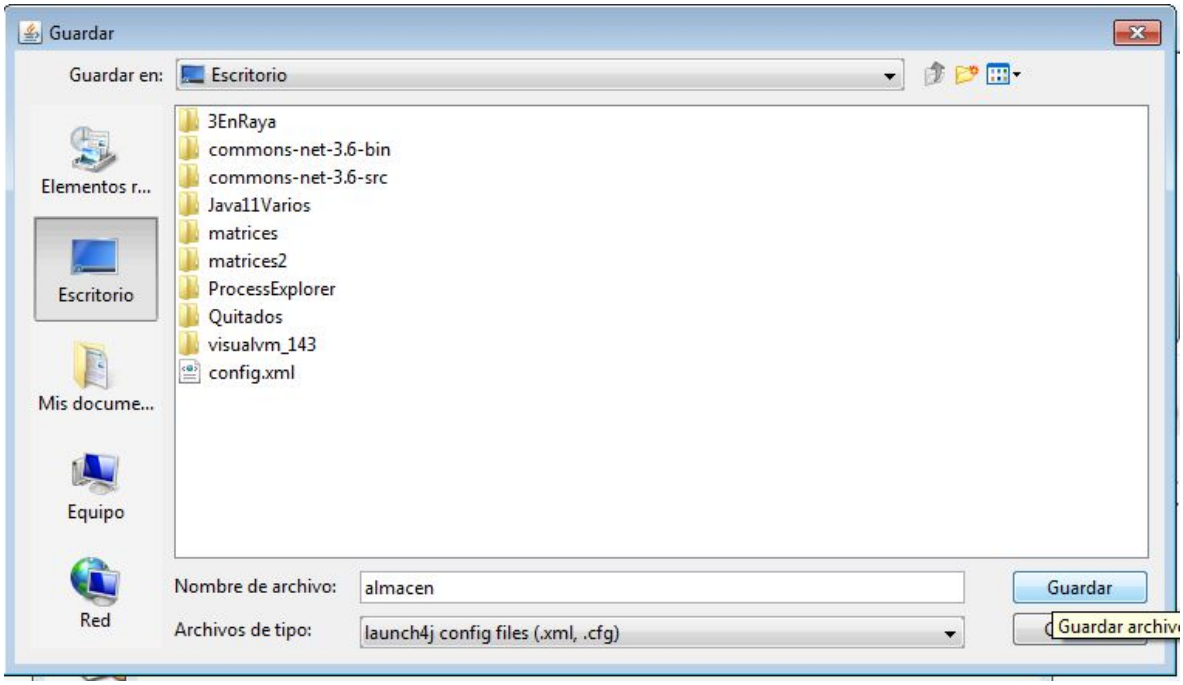
Paso 3

A continuación, indicamos con qué versiones mínima y máxima queremos que funcione nuestro programa:



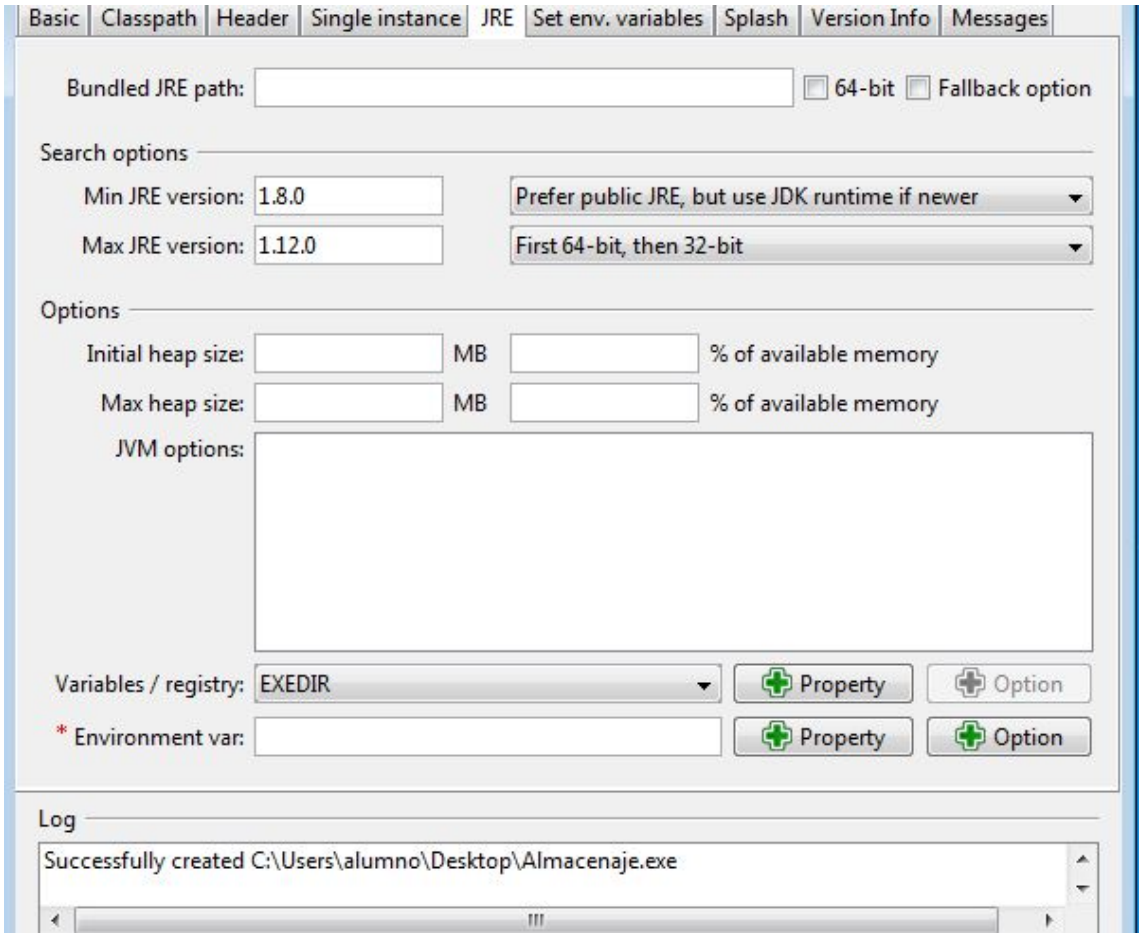
Paso 4

Damos al icono de guardar y nos pedirá un nombre para crear un archivo de configuración:



Paso 5

Pulsamos el icono de engranaje y nos crea el ejecutable:



Ya podemos distribuirlo para ejecutarlo en Windows.

Si queremos ejecutar el jar, también se podría (recordad que hemos creado un jar ejecutable, a diferencia de cuando creábamos librerías para incorporar paquetes externos). Eso lo veremos en clase cuando veamos como compilar ejecutar desde la línea de comandos.