

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 **Zapatero** que implementan el interfaz. En Armario hay un ArrayList de objetos tipo Ropa y en Zapatero 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, un Zapatero 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 Zapatero, 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 un Zapatero. 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.

Hacer otra versión del programa con un **menú** para rellenar los contenedores, que evite el error de introducir un objeto no adecuado en un Almacén.