Los datos llamados primitivos son los más básicos que nos presenta Java, pero hay casos, situaciones, métodos, en las que no podremos utilizar este tipo de dato. Como por ejemplo las colecciones, que no aceptan tipo de datos primitivos, pero, ¿cómo hacemos para guardar en una colección una serie de números, si no aceptan los tipos básicos?

Con las clases **Envoltorio**: Estas clases nos permiten envolver un tipo da dato básico en un objeto

**AutoBoxing** permite encapsular automáticamente un dato básico en una clase de envoltorio.

* int p=5
* Integer i=p;
* De esta forma no tenemos que crear la instancia (Java la crea aunque no se nos muestre)

**AutoUnBoxing** nos permite recuperar el tipo primitivo de un número almacenado en una clase de envoltorio sin utilizar "XXXvalue()" de esta forma:

* Integer i=new Integer(5);
* int n=i;
* Con esto no nos hace falta invocar métodos para obtener el valor del número almacenado

Dicho lo anterior muchos os preguntaréis el por qué he explicado algunos métodos si el **AutoBoxing** y el **AutoUnBoxing** nos hace lo mismo sin tener que utilizarlos. Bien, cabe mencionar, que esta cualidad fue introducida en la versión de Java 1.5 y por lo tanto en versiones anteriores a ésta habrá que utilizar los métodos