

## PECULIARIDAD DE Scanner

### ¿Qué ocurre si detrás de un `sc.nextInt()` va un `sc.nextLine()`?

Resulta que `sc.nextInt()`; deja basura en el buffer de entrada de caracteres de teclado:

2	3	\n			
---	---	----	--	--	--

Por ejemplo, si el usuario teclea 23, el método `nextInt()` se “lleva” el “2” y “3” pero deja “\n” (intro)

Si después hacemos un `nextLine()` para leer un “texto” resulta que ese método se encuentra en el buffer de teclado la siguiente información:

\n					
----	--	--	--	--	--

De modo que el método entiende que el usuario a pulsado INTRO directamente. Deja el buffer limpio pero no hemos podido leer la información que queríamos.

### SOLUCIÓN

Hacer dos `nextLine()` SEGUIDOS. El primero limpia el buffer y el segundo hace la lectura de la información.

```
Sc.nextLine(); // Limpia la entrada de caracteres
```

```
String texto = Sc.nextLine();
```