

## 1 Lectura del teclado: la clase Scanner

Como recordamos, la clase **System** incluía un objeto llamado **out** para poder escribir en la consola (mediante su método **print**). De igual forma, la clase **System** tiene otro objeto llamado **in** que representa al teclado, y que usaremos para leer la información introducida por el usuario.

Para poder leer del teclado, la API de Java incluye una clase sencilla para leer información de una fuente en general, la clase **Scanner**, que se encuentra en el paquete `java.util`. Si esa fuente es el objeto **System.in**, estaremos leyendo del teclado.

El primer paso será crear un objeto de esa clase, un objeto que leerá del teclado:

```
java.util.Scanner input = new java.util.Scanner(System.in);
```

Aunque podemos usar el nombre completo de la clase (`java.util.Scanner`), lo más habitual será importar la clase al comienzo del programa y luego usar su nombre abreviado<sup>1</sup>.

```
import java.util.Scanner;

Scanner input = new Scanner(System.in);
```

Entre los muchos métodos de los que dispone la clase **Scanner** (hay un enlace a la documentación en MOOVI), se encuentra `nextLine`, que devuelve como una cadena todos los caracteres introducidos por el usuario hasta el fin de línea (es decir, hasta el carácter de retorno de carro o tecla ENTER, que en Linux es `'\n'`).

```
String linea = input.nextLine();
```

El método `nextLine` se bloqueará a la espera de que el usuario teclee la línea y pulse luego ENTER. Tras ello, la cadena tecleada se asignará a la variable `linea`. El carácter `'\n'` es el delimitador que marca el final de la lectura, es leído y retirado de la entrada, pero no es devuelto como último carácter de la cadena leída.

El programa completo sería, pues:

```
import java.util.Scanner;

public class LeeLinea {

    public static void main (String[] args) {
        String linea;
        Scanner input;

        input = new Scanner(System.in);
        System.out.print("Introduce una línea: ");
        linea = input.nextLine();
        System.out.println("Has tecleado: '"+linea+"'");
    }
}
```

**IMPORTANTE.** Fíjate en el estilo de programación, que facilita la comprensión del código:

- Las variables tienen nombres significativos.
- Las variables están declaradas al comienzo del método.
- El código está adecuadamente sangrado.

---

<sup>1</sup> Siempre y cuando no existan colisiones con otra clase llamada igual (**Scanner**) que pertenezca a otro paquete.