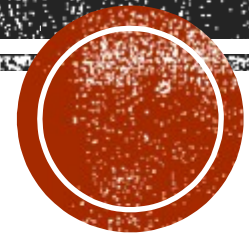


# LECTURA DE DATOS E

## JAVA



# SCANNER

- La lectura de datos en Java es realizada mediante esta clase.
- Para ello, tendremos que realizar los siguientes pasos:
  - 1) Escribir la directiva import
    - Al inicio de la clase añadimos:

```
import java.util.Scanner
```
  - 2) Creamos un objeto Scanner y le transferimos “el control del teclado” mediante System.in

```
Scanner sc = new Scanner(System.in);
```
  - 3) Al finalizar el uso del objeto, debemos cerrar “el control del teclado”, para ello añadiremos:

```
sc.close();
```



# SCANNER

➤ Ejemplo de lectura:

a) Lectura de un número entero:

```
int n;  
System.out.println("Introduzca un número entero: ");  
n = sc.nextInt();
```

b) Lectura de un número double:

```
double x;  
System.out.println("Introduzca número de tipo double: ");  
x = sc.nextDouble();
```

c) Lectura de una cadena de caracteres (**¡ojo! Hasta que pulses el enter**):

```
String s;  
System.out.println("Introduzca texto: ");  
x = sc.nextLine();
```



# SCANNER

- Problemas de lectura:

Cuando en un programa se leen por teclado datos numéricos y datos de tipo carácter o String tendremos que tener en cuenta que al introducir los datos y pulsar intro estamos también introduciendo en el buffer de entrada el intro.

Por ejemplo, dado el código:

```
System.out.print("Introduzca un número: ");  
  
n = sc.nextInt();  
System.out.print("Introduzca una cadena: ");  
  
ejemploCadena = sc.nextLine();
```



# SCANNER

Si introducimos el entero 13, se rellenará el buffer como: 13\n

Por lo tanto, tras leer el entero, el contenido del buffer será: \n

Al pedir después una cadena del buffer, leerá su contenido hasta encontrar un \n, que ya se encuentra en este.

→ Solución:

Debemos limpiar el buffer de entrada si se van a leer datos de tipo carácter a continuación de la lectura de datos numéricos.

La solución que vamos a utilizar, temporalmente, es ejecutar la instrucción:

```
sc.nextLine();
```

