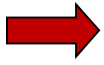


Ejemplo 1: Tratamiento de excepciones para leer 4 números enteros por teclado.

Se explican tres formas de hacer la lectura por teclado.

Opción 1: Un solo try..catch



Inconveniente: si hay un error en un dato se deben introducir todos de nuevo

```
public class Excepciones {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int x = 0, y = 0, z = 0, k = 0;
        boolean repetir;
        do {
            repetir = false;
            try {
                System.out.print("Introduce primer número entero: ");
                x = sc.nextInt();

                System.out.print("Introduce segundo número entero: ");
                y = sc.nextInt();

                System.out.print("Introduce tercer número entero: ");
                z = sc.nextInt();

                System.out.print("Introduce cuarto número entero: ");
                k = sc.nextInt();

            } catch (InputMismatchException e) {
                System.out.println("Valor no válido" + e.toString());
                sc.nextLine(); //Importante: limpiar el buffer
                repetir = true;
            }
        } while (repetir);

        System.out.println("int introducido -> " + x);
        System.out.println("int introducido -> " + y);
        System.out.println("int introducido -> " + z);
        System.out.println("int introducido -> " + k);
    }
}
```

Opción 2: Un try..catch para cada lectura.

Si hay un error solo se vuelve a pedir el número erróneo.



Inconveniente: duplicidad de código

```
public class Excepciones {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        int x = 0, y = 0, z = 0, k = 0;
        boolean repetir;

        do {
            repetir = false;
            try {
                System.out.print("Introduce primer número entero: ");
                x = sc.nextInt();
            } catch (InputMismatchException e) {
                System.out.println("Valor no válido " + e.toString());
                sc.nextLine();
                repetir = true;
            }
        } while (repetir);

        do {
            repetir = false;
            try {
                System.out.print("Introduce segundo número entero: ");
                y = sc.nextInt();
            } catch (InputMismatchException e) {
                System.out.println("Valor no válido " + e.toString());
                sc.nextLine();
                repetir = true;
            }
        } while (repetir);

        do {
            repetir = false;
            try {
                System.out.print("Introduce tercer número entero: ");
                z = sc.nextInt();
            } catch (InputMismatchException e) {
                System.out.println("Valor no válido " + e.toString());
                sc.nextLine();
                repetir = true;
            }
        } while (repetir);

        do {
            repetir = false;
            try {
                System.out.print("Introduce cuarto número entero: ");
                k = sc.nextInt();
            } catch (InputMismatchException e) {
                System.out.println("Valor no válido " + e.toString());
                sc.nextLine();
                repetir = true;
            }
        } while (repetir);

        System.out.println("int introducido -> " + x);
        System.out.println("int introducido -> " + y);
        System.out.println("int introducido -> " + z);
        System.out.println("int introducido -> " + k);
    }
}
```

Opción 3: Escribir un **método** para realizar la lectura del número entero.



Es la opción recomendada.

Dentro del método controlamos la excepción.

Evitamos la duplicidad de código.

```
public class Excepciones {
    public static void main(String[] args) {
        int x = 0, y = 0, z = 0, k = 0;

        x = leerEntero("Introduce primer número entero: ");
        y = leerEntero("Introduce segundo número entero: ");
        z = leerEntero("Introduce tercer número entero: ");
        k = leerEntero("Introduce cuarto número entero: ");

        System.out.println("int introducido -> " + x);
        System.out.println("int introducido -> " + y);
        System.out.println("int introducido -> " + z);
        System.out.println("int introducido -> " + k);
    }

    public static int leerEntero(String s) {
        Scanner sc = new Scanner(System.in);
        boolean repetir;
        int n = 0;
        do {
            repetir = false;
            try {
                System.out.print(s);
                n = sc.nextInt();
            } catch (InputMismatchException e) {
                System.out.println("Valor no válido");
                repetir = true;
            } finally {
                sc.nextLine();
            }
        } while (repetir);
        return n;
    }
}
```

El método *leerEntero* también se puede escribir de esta forma:

```
public static int leerEntero(String s) {
    Scanner sc = new Scanner(System.in);
    int n = 0;
    while(true){
        try {
            System.out.print(s);
            n = sc.nextInt();
            return n;
        } catch (InputMismatchException e) {
            System.out.println("Valor no válido");
        } finally {
            sc.nextLine();
        }
    }
}
```

Ejemplo 2: Leer dos int y dos double y mostrar los valores leídos por pantalla. Realiza un método para leer un int y otro para leer un double. Realiza en estos métodos el control de excepciones en la lectura por teclado.

```
public class Excepciones2 {
    public static void main(String[] args) {
        int x = 0, y = 0;
        double m = 0, n = 0;

        x = leerEntero("Introduce un número entero: ");
        y = leerEntero("Introduce otro número entero: ");

        m = leerDouble("Introduce un número double: ");
        n = leerDouble("Introduce otro número double: ");

        System.out.println("primer int introducido -> " + x);
        System.out.println("segundo int introducido -> " + y);
        System.out.println("primer double introducido -> " + m);
        System.out.println("segundo double introducido -> " + n);
    }

    public static int leerEntero(String s) {
        Scanner sc = new Scanner(System.in);
        int n = 0;
        while (true) {
            try {
                System.out.print(s);
                n = sc.nextInt();
                return n;
            } catch (InputMismatchException e) {
                System.out.println("Valor no válido");
            } finally {
                sc.nextLine();
            }
        }
    }

    public static double leerDouble(String s) {
        Scanner sc = new Scanner(System.in);
        double n = 0;
        while (true) {
            try {
                System.out.print(s);
                n = sc.nextDouble();
                return n;
            } catch (InputMismatchException e) {
                System.out.println("Valor no válido");
            } finally {
                sc.nextLine();
            }
        }
    }
}
```