

EXCEPCIONES

Servir para utilizar situaciones anómalas y ocurren durante la ejecución. Las excepciones, en Java, también son objetos, y hay que crearlas en caso de necesitarlas.

• Hay que definir 1 clase por cada excepción que se desea tratar.

• Dicha clase será hija de **Exception**.

Para indicar que un método puede producir una excepción:

- Mediante throws (se pone después de los argumentos del método, y tras él, el nombre de la excepción) y throws (se pone en el momento en el que debe saltar la excepción, seguida va el nombre de la excepción como objeto).

- Mediante try-catch: Esto permite que el código siga ejecutándose aunque se produzca una excepción.

```
try {
    ...
    ← Código que podría generar una excepción.
}
catch (ExcepciónASaltar e) {
    ...
    ← Código que se ejecuta si se produce la excepción.
}
```

• Si tenemos un try-catch con varias operaciones dentro, si una falla, todas las que voya detrás no se ejecutan, pero aquellos que esté después del try catch sí se continúan ejecutando.

Ejemplo:

```
try {
    factorial(1)
    factorial(-5) → Esto provocará excepción, por lo que factorial(3)
    factorial(3)    no se ejecutará nunca, pero factorial(1) sí se
    catch (ExcepciónFactorialNegativo e) {
        e.printStackTrace();
    }
}
```

→ Lo que vaya tras el try catch sí se ejecuta aunque falle excepción.