



Declaración y lanzamiento de una excepción





Declaración y lanzamiento de una excepción

Las excepciones permiten que un método informe al código que lo llama sobre un error o condición inusual que ocurrió durante su ejecución (Universidad Carlos III de Madrid, s.f.).

Por ejemplo, supongamos que un método toma un código postal y devuelve el nombre de la ciudad a la que pertenece. En circunstancias normales, el código de correo es correcto y este método regresa al municipio. Si el código postal o su formato no existe o es incorrecto, el método se notificará iniciando una excepción. Las excepciones en Java se conforman por objetos de la clase `Exception` o una de sus subclases.

Declaración de nuevas excepciones

Java tiene varias excepciones predefinidas para situaciones específicas, como los errores `IOException`, los cuales ocurren durante las operaciones de entrada/salida, como la lectura de un archivo. Para nuestros propios programas, a veces resulta útil crear excepciones personalizadas. Para hacer esto, necesitamos declarar una nueva clase que herede de `Exception`. Lo anterior lo explica la Universidad Carlos III de Madrid (s.f.) en el siguiente ejemplo:

```
public class BadPostCodeException
extends Exception {

    public BadPostCodeException()
    {
        super();
    }

    public
    BadPostCodeException(String message)
```

```
{
    super(message);
}
```

Normalmente, no necesitamos declarar nuevas propiedades o métodos, excepto los constructores.

Cómo hacer una excepción

Si necesitamos lanzar una excepción en un método, usamos la palabra clave `throw` seguida del caso de excepción. Para explicar lo anterior, contamos con el siguiente ejemplo (Universidad Carlos III de Madrid, s.f.):

```
if (postCode.length() != 5) {
    throw new
    BadPostCodeException("Postcodes must
    have 5 digits");
}
for (int i = 0; i < postCode.length();
i++) {
    if
    (!Character.isDigit(postCode.charAt(i))
    ) {
        throw new
        BadPostCodeException("Postcodes can
        only contain digits");
    }
}
City city = lookupPostCode(postCode);
if (city == null) {
    throw new
    BadPostCodeException("Postcode not in
    database: " + postCode);
}
```

Menos un conjunto especial de excepciones, un método que puede lanzar una excepción debe indicarlo



explícitamente usando la palabra clave throws (ten en cuenta la "s" al final, que no debe confundirse con throw) seguida del nombre de clase de la excepción. Excepción que puedes lanzar:

```
public class PostCodeManager {
    (...)

    public String town(String postCode)
    throws BadPostCodeException {
        if (postCode.length() != 5) {
            throw new
            BadPostCodeException("Postcodes must
            have 5 digits");
        }
        for (int i = 0; i <
        postCode.length(); i++) {
            if
            (!Character.isDigit(postCode.charAt(i))
            ) {
```

```
                throw new
                BadPostCodeException("Postcodes
                only contain digits");
            }
        }
        City city =
        lookupPostCode(postCode);
        if (city == null) {
            throw new
            BadPostCodeException("Postcode not in
            database: " + postCode);
        }
        (...)
    }
    (...)
}
```

Si un método puede lanzar varias clases de excepción, separa todos los nombres de clase con comas después de un único throws.





Referencias bibliográficas

- Universidad Carlos III de Madrid. (s.f.). *Breve introducción a las excepciones en Java*. Recuperado de <https://bit.ly/44G6mJD>