



# Java Standard Edition





## 5. Manejo de errores y excepciones

- Que es una excepción
- Especificar una excepción
- Trabajar con excepciones
- Capturar y manejar una excepción
- Bloque try
- Bloque catch
- Bloque finally
- Sentencia throw



# ¿Qué es una excepción?

En java a una situación que puede provocar un fallo en la aplicación se denominan excepción. Las excepciones en java son objetos de la clase **Exception**.





## Ejemplos:

En java los errores que podríamos encontrar en las aplicaciones podrían ser:

1. Divisiones por cero.
2. Intentar acceder a índices mayores a los declarados en un arreglo.
3. Al acceder a una base de datos.
4. En el manejo de archivos.
5. Errores definidos por el usuario.



# Try-catch

El bloque try protege el flujo de un aplicación al entrar en ejecución.

```
try{  
    //Instrucciones  
}catch(Clase identificador){  
    //Instrucciones catch  
}
```



## Ejercicios:

1. Hacer casting de un **double** a un **String**, ver la excepción generada y controlar la excepción.
2. Intentar una división por cero con tipos de dato entero, hacer la observación de la excepción generada y controlar la excepción.
3. Crear un menú de opciones y pedir al usuario elija una opción, al capturar la opción introducir una cadena de letras, ver la excepción generada y controlar la excepción.



## 6. Manejo de archivos

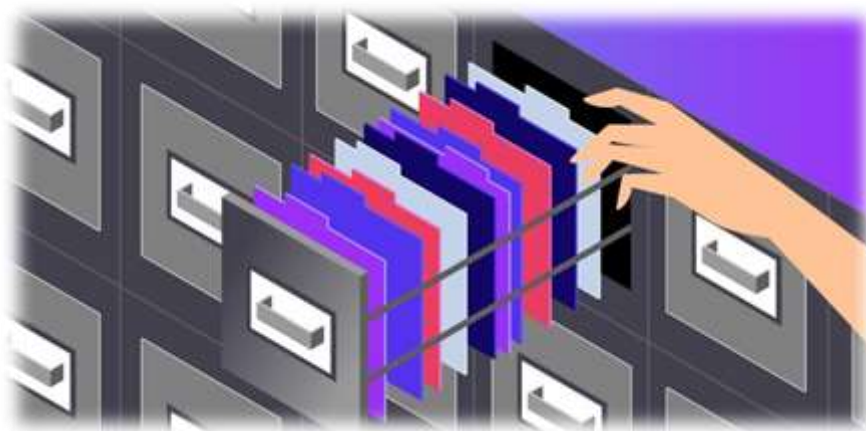


- **I/O: Leer y escribir**
- **Creación de un archivo.**
- **Analizar un directorio .**
- **Capturar y manejar una excepción**
- **Bloque try**
- **Bloque catch**
- **Bloque finally**
- **Sentencia throw**



# Clase File.

La clase File se usa para **obtener información** sobre archivos y directorios.  
Además la clase File permite crear y eliminar archivos y directorios.  
Un objeto de la clase Java File **representa** un archivo o directorio.







# Crear un archivo nuevo.

Existen un conjunto de clases que nos permiten manejar archivos en java, por ejemplo la clase **File**.

1. El hecho de crear un objeto de esta clase no creará el archivo en el disco.
2. Se puede utilizar la clase file para crear un archivo o un directorio
3. El método exists() devuelve un valor boolean que es verdadero si el archivo ya existe en el equipo y falso si no.
4. El método createNewFile() creará el archivo en el disco duro.
5. El método createNewFile() puede provocar una IOException así que debe ser manejada



# Listando archivos.

El constructor de la clase File recibe como parámetro un directorio.

El método **listFiles()** devuelve un arreglo de objetos de la clase File cada uno representa un archivo contenido en el directorio.



# Métodos de File.

1. **createNewFile()**: Crea un nuevo archivo en disco
2. **delete()**: Borra un archivo
3. **exists()**: Devuelve verdadero o falso dependiendo si el archivo existe o no
4. **isDirectory()**: Devuelve verdadero o falso dependiendo si es un directorio o no
5. **isFile()**: Devuelve verdadero o falso dependiendo si es un archivo o no
6. **list()**: Lista todos los archivos contenidos en el directorio
7. **mkdir()**: Crear un nuevo directorio
8. **renameTo()** : Re nombra un archivo