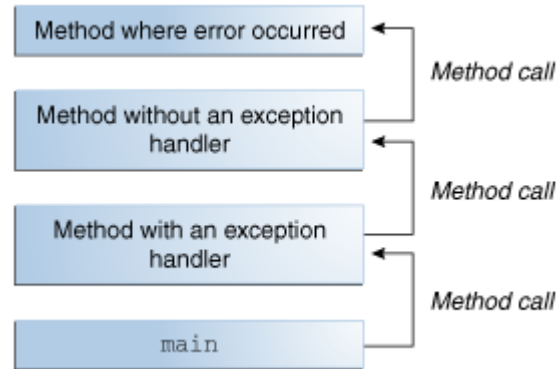


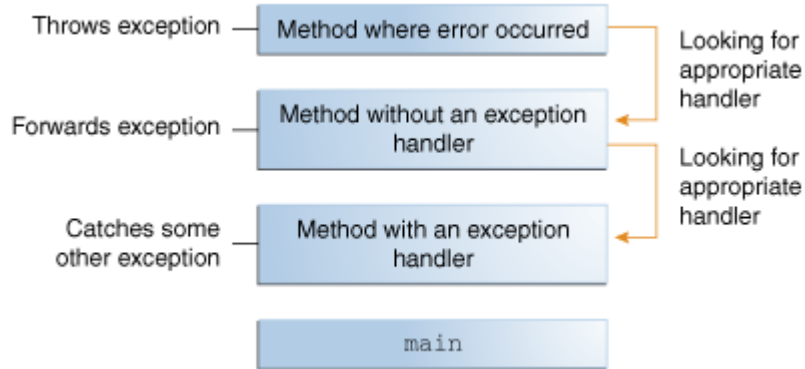
# EXCEPCIÓN

- ▶ *Situación excepcional.*
- ▶ Altera la ejecución normal del programa.
- ▶ El método donde sucede crea un objeto, llamado *objeto de excepción*, y se lo pasa a alguien que pueda tratarlo.



# EXCEPCIÓN

- Si existe quien pueda manejarlo, la recoge.



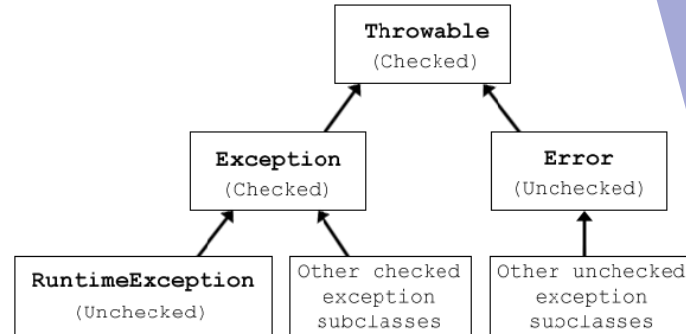
- Si no existe, se encarga la JVM.

## USO DE EXCEPCIONES

- ▶ Permiten separar el código de tratamiento de errores del código normal.
- ▶ Evitan que haya errores inadvertidos.
- ▶ Permiten la propagación de los errores.
- ▶ Permiten agrupar en un lugar común el tratamiento de errores.

# TIPOS DE EXCEPCIONES

- ▶ *Checked Exceptions:*  
excepciones que son recogidas y tratadas por programas bien escritos.
- ▶ *Error:* son externos a la aplicación, y no nos podemos anticipar a ellos.
- ▶ *Runtime error:* situaciones internas a la aplicación, y de las que no nos podemos recuperar



# TIPOS DE EXCEPCIONES

Usaremos *Checked Exceptions* cuando:

- ▶ La excepción es la única manera de detectar el error.
- ▶ No queremos que pase inadvertido

Usaremos *Unchecked Exceptions* cuando:

- ▶ Podemos intentar mejorar el código para que no suceda dicho error
- ▶ La excepción sirve para detectar y corregir usos indebidos de la clase.
- ▶ Errores internos ante los que poco podemos hacer.