

Programación Orientada a Objetos en Java

Curso 2006 - 2007

Tema 6 - Excepciones

Gonzalo Méndez Pozo
Dpto. de Ingeniería de Software e Inteligencia Artificial
Universidad Complutense de Madrid



Excepciones



- ¿Qué son?
 - Mecanismo de control de errores en tiempo de ejecución
 - Una forma de hacer que la aplicación continúe la ejecución si se produce un error
- ¿Qué sucede si no se puede abrir un fichero? ¿Qué pasa si un socket se cierra de manera inesperada? ¿Y si dividimos por cero?

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Excepciones

- ¿Para qué sirven?
 - A veces se usan para detectar situaciones inusuales en la ejecución
 - Si se controlan todos los posibles errores directamente, el código se puede volver ilegible
 - Con las excepciones se separa el código de un método del código que controla los errores

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



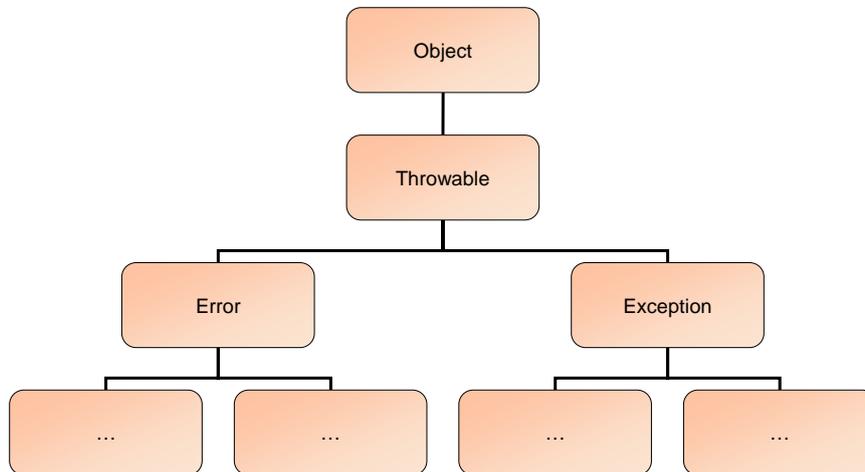
Excepciones

- Como casi todo lo demás en java, las excepciones son objetos (que se crean cuando ocurre una situación anómala)
- Se *lanzan* para que otra parte del código las *capture* y las trate

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Jerarquía de Excepciones



Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Jerarquía de Excepciones



- Las excepciones que se derivan de Error suelen estar relacionadas con la máquina virtual y no se espera que se capturen ni se traten
- Las Excepciones derivadas de Exception sí que deben ser tratadas, y en algunos casos es obligatorio hacerlo para que el programa compile

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Jerarquía de Excepciones



- No es necesario tratar las Excepciones que derivan de RuntimeException:
 - ArithmeticException
 - IndexOutOfBoundsException
 - NullPointerException
 - SecurityException
 - ...

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Tratamiento de Excepciones



- El código que puede generar la excepción debe encerrarse dentro de un bloque **try**

```
try
{
    // Código que puede generar la excepción
}
```

- A continuación, la excepción se captura con un bloque **catch**

```
catch (Exception e)
{
    // Código para tratar el error
}
```

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Tratamiento de Excepciones



- El código dentro del try puede generar más de una excepción, y se pueden capturar todas ellas

```
try
{
    //Código que puede provocar el error
}
catch(IOException ioe)
{
    //Código para tratar la IOException
}
catch(Exception e)
{
    //Código para tratar la Exception
}
```

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Tratamiento de Excepciones



- Se puede capturar una excepción utilizando un tipo de excepción más general

```
try
{
    //Código que produce una excepción de tipo IOException
}
catch (Exception e)
{
    //Código para tratar una Exception
}
```

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Tratamiento de Excepciones



- Esto último se puede hacer gracias a que unas excepciones heredan de otras
- Sólo se puede hacer con excepciones dentro de la misma jerarquía
- Sirve para tratar de manera común varios tipos de excepciones distintos

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Tratamiento de Excepciones



- **CUIDADO:** Si un bloque de código lanza varias excepciones y se usan varios catch
 - La excepción se captura en el primer catch que se ajusta a la excepción
 - Los catch deben capturar las excepciones más concretas en primer lugar, y las más generales al final
 - Si no lo hacemos así, hay bloques catch en los que no se entrará nunca

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Tratamiento de Excepciones



```
try
{
    //código que genera
    //excepciones
}
catch(IOException ioe)
{
    //catch accesible
}
catch (Exception e)
{
    //catch accesible
}

try
{
    //código que genera
    //excepciones
}
catch(Exception e)
{
    //catch accesible
}
catch (IOException ioe)
{
    //catch NO accesible
}
```

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Tratamiento de Excepciones



- A veces, cuando se produce una excepción, la aplicación queda es un estado inestable
- Al tratamiento de una excepción se le puede añadir al final un bloque finally que se ejecuta siempre, se produzcan o no excepciones
- Se puede usar para cerrar ficheros, liberar recursos, etc.

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Tratamiento de Excepciones



```
try
{
    //código que genera excepciones
}
catch(IOException ioe)
{
    //tratamiento de la excepción
}
finally
{
    //código que se ejecuta siempre
}
```

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Tratamiento de Excepciones



- Si una excepción no se captura se propaga hacia el método llamante, para que éste la trate
- Si no la trata, se sigue propagando hasta llegar al main
- Si en el main tampoco se trata, se aborta la ejecución del programa

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Lanzar Excepciones

- Parte del tratamiento de una excepción puede incluir la propagación de la misma para que también la traten los métodos llamantes
- También puede suceder que queramos que nuestro código lance una excepción en una situación determinada
- En ambos casos es necesario decir que el método puede lanzar una excepción

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Lanzar Excepciones

- Esto se hace utilizando la palabra reservada **throws** en la cabecera del método

```
public void miMetodo() throws ArithmeticException
```

- Posteriormente, en el código, se puede lanzar una excepción usando la palabra reservada **throw**

```
catch (IOException ioe)
{
    throw ioe;
}
...
throw new NullPointerException();
```

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial



Crear Nuevas Excepciones



- Se pueden crear nuevas excepciones creando una nueva clase que herede de Exception
- Se suele añadir un constructor por defecto y otro que acepta un String
- Llamar al constructor de Exception desde el constructor de la nueva clase
- Si hace falta, añadir atributos, otros constructores y otros métodos

Gonzalo Méndez - Dpto. Ingeniería de Software e Inteligencia Artificial