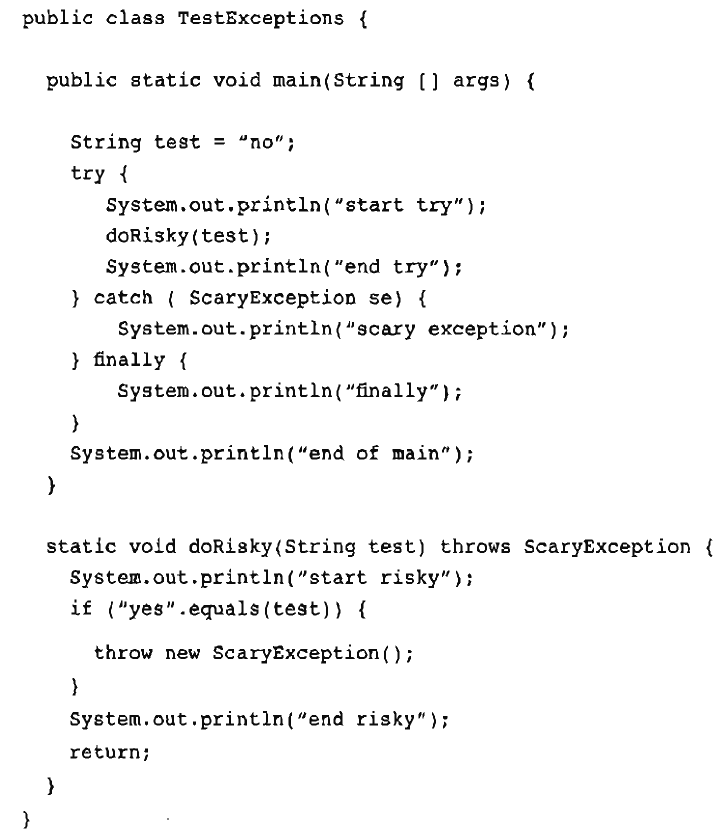
**Práctica “Tratamiento de Errores”**

**Temas:**

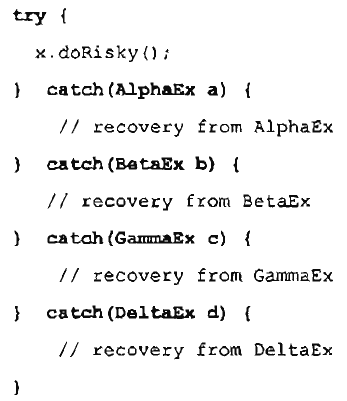
* Tratamiento de errores
* excepciones
* try/catch, throw, finally

Para realizar esta práctica cree un proyecto llamado *practica6* en su espacio de trabajo

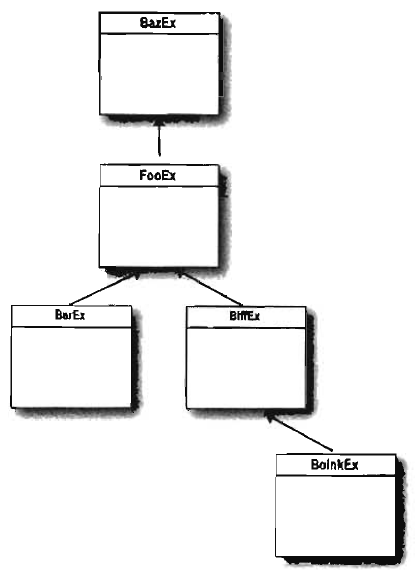
1. Analice el código a continuación, y responda luego a los incisos**.**



1. ¿Cuál debiera ser la salida del programa? (asumir que **ScaryException** extiende **Exception**)
2. Si se cambia la 3ra. Línea del programa a String test = “yes”, ¿cómo afecta esto la salida del programa?
3. Asumiendo que el **bloque try/catch** del ejemplo (abajo) está bien codificado, dibuje los 2 diagramas de clases (jerarquías) que reflejan de forma adecuada las distinta clases de **Exception**. El objetivo es que el ejemplo compile.



1. Dado el siguiente diagrama de clases, escribir 2 porciones de código con **estructuras try/catch** que reflejen adecuadamente el diagrama. El objetivo es que el código compile. Asumir que TODAS estas excepciones deben ser lanzadas por el método con el **bloque try**.



1. Dada la clase **ListOfNumbers**, agregarle un método **readList**. Este método debe leer un valor *int* de un archivo, imprimir cada valor, e ir agregando cada valor al final de un vector. En este proceso, se pide capturar todos los errores que se consideren apropiados. Asumir que se debe proveer un archivo de texto con números para hacer funcionar el programa.

import java.io.\*;

import java.util.List;

import java.util.ArrayList;

public class ListOfNumbers {

private List<Integer> list;

private static final int SIZE = 10;

public ListOfNumbers () {

list = new ArrayList<Integer>(SIZE);

for (int i = 0; i < SIZE; i++)

list.add(new Integer(i));

}

public void writeList() {

PrintWriter out = null;

try {

System.out.println("Entering try statement");

out = new PrintWriter(new FileWriter("OutFile.txt"));

for (int i = 0; i < SIZE; i++)

out.println("Value at: " + i + " = " + list.get(i));

} catch (IndexOutOfBoundsException e) {

System.err.println("Caught IndexOutOfBoundsException: " +

e.getMessage());

} catch (IOException e) {

System.err.println("Caught IOException: " + e.getMessage());

} finally {

if (out != null) {

System.out.println("Closing PrintWriter");

out.close();

} else {

System.out.println("PrintWriter not open");

}

}

}

}

1. Modificar el siguiente método **cat** para que el programa compile.

public static void cat(File file) {

RandomAccessFile input = null;

String line = null;

try {

input = new RandomAccessFile(file, "r");

while ((line = input.readLine()) != null) {

System.out.println(line);

}

return;

} finally {

if (input != null) {

input.close();

}

}

}