Excepciones

El código intenta conectarse a una página web, pero la conexión a Internet ha caído.

Existe un fallo en el código e intento de acceso a un elemento de un array que no es válido.

Un método que llama a otro con un valor que el método no soporta.

Excepciones

```
01: public class Zoo {
02: public static void main(String[] args) {
03:       System.out.println(args[0]);
04:       System.out.println(args[1]);
05: } }
```

...

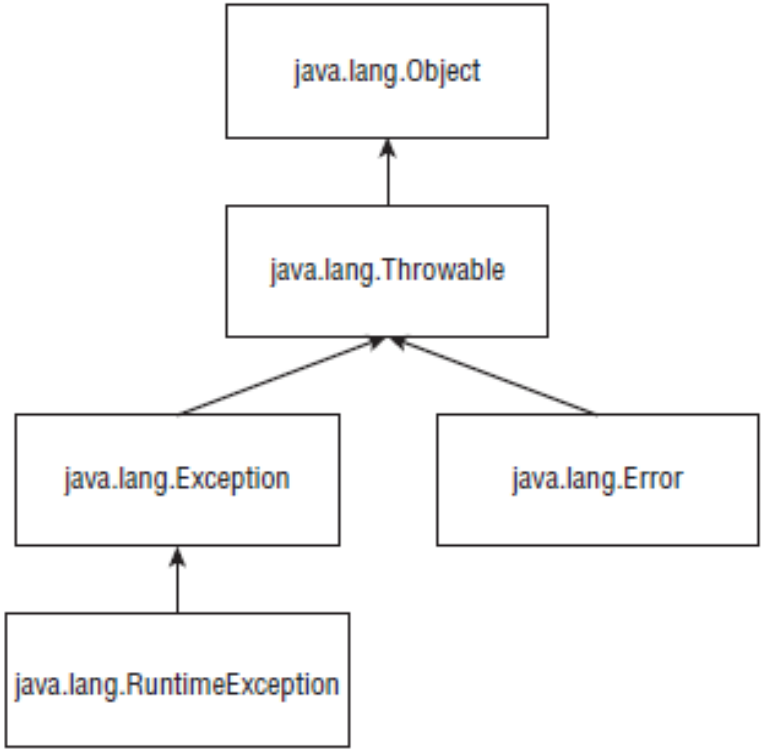
```
$ javac Zoo.java
$ java Zoo Zoo
```

ZooException in thread "main" java.lang.ArrayIndexOutOfBoundsException: 1 at mainmethod.Zoo.main(Zoo.java:7)

Excepciones

```
01: public class Zoo {
02: public static void main(String[] args) {
03:       System.out.println(args[0]);
04:       System.out.println(args[1]);
05: } }
...
public int indexOf(String[] names, String name) {
      for (int i = 0; i < names.length; i++) {
            if (names[i].equals(name)) {
                  return i;
            }
      }
      return -1;
}
...
void fall() throws Exception { throw new Exception(); }
```

Excepciones



Excepciones

```
String[] animals = new String[0];
System.out.println(animals[0]);

...

throw new Exception();
throw new Exception("Ow! I fell.");
throw new RuntimeException();
throw new RuntimeException("Ow! I fell.");
throw new Exception(e);
```

Excepciones

Tipo	Como reconocerla	¿El programa puede manejarla?	¿El programa tiene que manejarla o declararla?
Runtime exception	Subclase de RuntimeException	Si	No
Checked exception	Subclase de Exception pero no sucblase de RuntimeException	Si	Si
Error	Subclase de Error	No	No

Excepciones

```
03: void explore() {
04:         try {
05:                 fall();
06:                 System.out.println("never get here");
07:         } catch (RuntimeException e) {
08:                 getUp();
09:         }
10:         seeAnimals();
11: }
12: void fall() { throw new RuntimeException(); }
```

Excepciones

```
try                                                    // DOES NOT COMPILE
      fall();
catch (Exception e)
      System.out.println("get up");

...

try {                                                  // DOES NOT COMPILE
      fall();
}

...

try {
      fall();
} catch (Exception e) {
      System.out.println("get up");
}
```

Excepciones

```
12: void explore() {
13:         try {
14:                 seeAnimals();
15:                 fall();
16: } catch (Exception e) {
17:         getHugFromDaddy();
18: } finally {
19:         seeMoreAnimals();
20: }
21: goHome();
22: }
```

Excepciones

```
25:  try {                                                  // DOES NOT COMPILE
26:        fall();
27: } finally {
28:        System.out.println("all better");
29: } catch (Exception e) {
30:        System.out.println("get up");
31: }
32:
33: try {                                                   // DOES NOT COMPILE
34:        fall();
35: }
36:
37: try {
38:        fall();
39: } finally {
40:        System.out.println("all better");
41: }
```

Excepciones

```
String s = "";
try {
	s += "t";
} catch(Exception e) {
	s += "c"; }
finally {
	s += "f";
}
s += "a";
System.out.print(s);
```

Resultado

Excepciones

```
String s = "";
try {
	s += "t";
} catch(Exception e) {
	s += "c"; }
finally {
	s += "f";
}
s += "a";
System.out.print(s);
```

Resultado tfa

Excepciones

System.exit

Excepciones

```
public void visitPorcupine() {
      try {
            seeAnimal();
      }
      catch (AnimalsOutForAWalk e) {                  // first catch block
            System.out.print("try back later");
      }
      catch (ExhibitClosed e) {                       // second catch block
            System.out.print("not today");
      }
}
```

Excepciones

```
public void visitMonkeys() {
      try {
            seeAnimal();
      }
      catch (ExhibitClosedForLunch e) {                  // subclass exception
            System.out.print("try back later");
      }
      catch (ExhibitClosed e) {                          // superclass exception
            System.out.print("not today");
      }
}
```

Excepciones

```
public void visitMonkeys() {
	try {
		seeAnimal();
	}
	catch (ExhibitClosed e) {
		System.out.print("not today");
	}
	catch (ExhibitClosedForLunch e) {		// DOES NOT COMPILE
		System.out.print("try back later");
	}
}
```

Excepciones

```
public void visitSnakes() {
      try {
            seeAnimal();
      }
      catch (RuntimeException e) {
            System.out.print("runtime exception");
      }
      catch (ExhibitClosed e) {                          // DOES NOT COMPILE
            System.out.print("not today");
      }
      catch (Exception e) {
            System.out.print("exception");
      }
}
```

Excepciones

```
16: public static void main(String[] args) {
17:         FileReader reader = null;
18:         try {
19:                 reader = read();
20:         } catch (IOException e) {
21:                 try {
22:                         if(reader != null) reader.close();
23:                 } catch (IOException inner) {
24:                 }
25:         }
26: }
27: private static FileReader read() throws IOException {
28:         // EL CÓDIGO VA AQUÍ
29: }
```

Excepciones

```
26: try {
27:         throw new RuntimeException();
28: } catch (RuntimeException e) {
29:         throw new RuntimeException();
30: } finally {
31:         throw new Exception();
32: }
```

Excepciones

```
30: public String exceptions() {
31:         String result = "";
32:         String v = null;
33:         try {
34:               try {
35:                     result += "before";
36:                     v.length();
37:                     result += "after";
38:               } catch (NullPointerException e) {
39:                     result += "catch";
40:                     throw new RuntimeException();
41:               } finally {
42:                     result += "finally";
43:                     throw new Exception();
44:               }
45:         } catch (Exception e) {
46:               result += "done";
47:         }
48:         return result;
49: }
```

Resultado

Excepciones

```
30: public String exceptions() {
31:         String result = "";
32:         String v = null;
33:         try {
34:               try {
35:                     result += "before";
36:                     v.length();
37:                     result += "after";
38:               } catch (NullPointerException e) {
39:                     result += "catch";
40:                     throw new RuntimeException();
41:               } finally {
42:                     result += "finally";
43:                     throw new Exception();
44:               }
45:         } catch (Exception e) {
46:               result += "done";
47:         }
48:         return result;
49: }
```

Resultado before catch finally done

Excepciones

Runtime Exceptions

- ArithmeticException
- ArrayIndexOutOfBoundsException
- ClassCastException
- IllegalArgumentException
- NullPointerException
- NumberFormatException

Excepciones

Runtime Exceptions - ArithmeticException

int answer = 11 / 0;

Exception in thread "main" java.lang.ArithmeticException: / by zero

Excepciones

Runtime Exceptions - ArrayIndexOutOfBoundsException

```
int[] countsOfMoose = new int[3];
System.out.println(countsOfMoose[-1]);
```

```
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: -1
```

...

```
int total = 0;
int[] countsOfMoose = new int[3];
for (int i = 0; i <= countsOfMoose.length; i++)
      total += countsOfMoose[i];
```

```
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 3
```

Excepciones

Runtime Exceptions - ClassCastException

```
String type = "moose";
Integer number = (Integer) type;                    // DOES NOT COMPILE
```

...

```
String type = "moose";
Object obj = type;
Integer number = (Integer) obj;
```

```
Exception in thread "main" java.lang.ClassCastException: java.lang.String
cannot be cast to java.lang.Integer
```

Excepciones

Runtime Exceptions - IllegalArgumentException

```
06: public void setNumberEggs(int numberEggs) {          // setter
07:       if (numberEggs >= 0)                           // guard condition
08:            this.numberEggs = numberEggs;
09: }

...

public static void setNumberEggs(int numberEggs) {
      if (numberEggs < 0)
            throw new IllegalArgumentException("# eggs must not be negative");
      this.numberEggs = numberEggs;
}
```

Excepciones

Runtime Exceptions - NullPointerException

```
String name;
public void printLength() throws NullPointerException {
	System.out.println(name.length());
}
```

```
Exception in thread "main" java.lang.NullPointerException
```

Excepciones

Runtime Exceptions - NumberFormatException

Integer.parseInt("abc");

Exception in thread "main" java.lang.NumberFormatException: For input string: "abc"

Excepciones

Excepciones controladas (Checked Exceptions)

FileNotFoundExceptions
IOException

Excepciones

Errores

ExceptionInInitializerError
StackOverflowError
NoClassDefFoundError

Excepciones

ExceptionInInitializerError

```
static {
	int[] countsOfMoose = new int[3];
	int num = countsOfMoose[-1];
}

public static void main(String[] args) { }
```

Exception in thread "main" java.lang.ExceptionInInitializerError Caused by: java.lang.ArrayIndexOutOfBoundsException: -1

Excepciones

StackOverflowError

```
public static void doNotCodeThis(int num) {
	doNotCodeThis(1);
}
```

```
Exception in thread "main" java.lang.StackOverflowError
```

Excepciones

NoClassDefFoundError

Excepciones

```
class NoMoreCarrotsException extends Exception {}

public class Bunny {
      public static void main(String[] args) {
            eatCarrot();                                          // DOES NOT COMPILE
      }
      private static void eatCarrot() throws NoMoreCarrotsException {}
}
...
                                                      //declare exception
public static void main(String[] args) throws NoMoreCarrotsException {
      eatCarrot();
}

public static void main(String[] args) {
      try {
            eatCarrot();
      }
      catch (NoMoreCarrotsException e ) {                     // handle exception
            System.out.print("sad rabbit");
      }
}
```

Excepciones

```
public void bad() {
	try {
		eatCarrot();
	} catch (NoMoreCarrotsException e ) {		// DOES NOT COMPILE
		System.out.print("sad rabbit");
	}
}

public void good() throws NoMoreCarrotsException {
	eatCarrot();
}

private static void eatCarrot() { }
```

Excepciones

```
class CanNotHopException extends Exception { }

class Hopper {
	public void hop() { }
}

class Bunny extends Hopper {
	public void hop() throws CanNotHopException { }	// DOES NOT COMPILE
}
```

Excepciones

```
class Hopper {
	public void hop() throws CanNotHopException { }
}

class Bunny extends Hopper {
	public void hop() { }
}
```

Excepciones

```
class Hopper {
	public void hop() throws Exception { }
}

class Bunny extends Hopper {
	public void hop() throws CanNotHopException { }
}
```

Excepciones

```
class Hopper {
	public void hop() { }
}

class Bunny extends Hopper {
	public void hop() throws IllegalStateException { }
}
```

Excepciones

```
05: public static void main(String[] args) {
06:         try {
07:                 hop();
08:         } catch (Exception e) {
09:                 System.out.println(e);
10:                 System.out.println(e.getMessage());
11:                 e.printStackTrace();
12:         }
13: }
14: private static void hop() {
15:         throw new RuntimeException("cannot hop");
16: }
```

```
java.lang.RuntimeException: cannot hop
cannot hop
java.lang.RuntimeException: cannot hop at
trycatch.Handling.hop(Handling.java:15) at
trycatch.Handling.main(Handling.java:7)
```

Excepciones

```
public static void main(String[] args) {
	String textInFile = null;
	try {
		readInFile();
	} catch(IOException e) {
		// ignore exception
	}
	// imagine many lines of code here
	System.out.println(textInFile.replace("", ""));
}

private static void readInFile() throws IOException {
	throw new IOException();
}
```

Resultado NullPointerException’