



# Universidad Distrital Francisco José de caldas

## Tecnología en Sistematización de Datos

```
Exception in thread "main" java.lang.NullPointerException
    at control.Controlador.inicio(Controlador.java:24)
    at proyempleado.ProyEmpleado.main(ProyEmpleado.java:22)
```

```
C:\Users\SONIA\AppData\Local\NetBeans\Cache\8.2\executor-snippets\run.xml:53: Java returned: 1
```

```
BUILD FAILED (total time: 0 seconds)
```

```
BUILD FAILED (total time: 0 seconds)
```

## Manejo de Excepciones

Programación Multinivel  
Sonia Alexandra Pinzón Nuñez



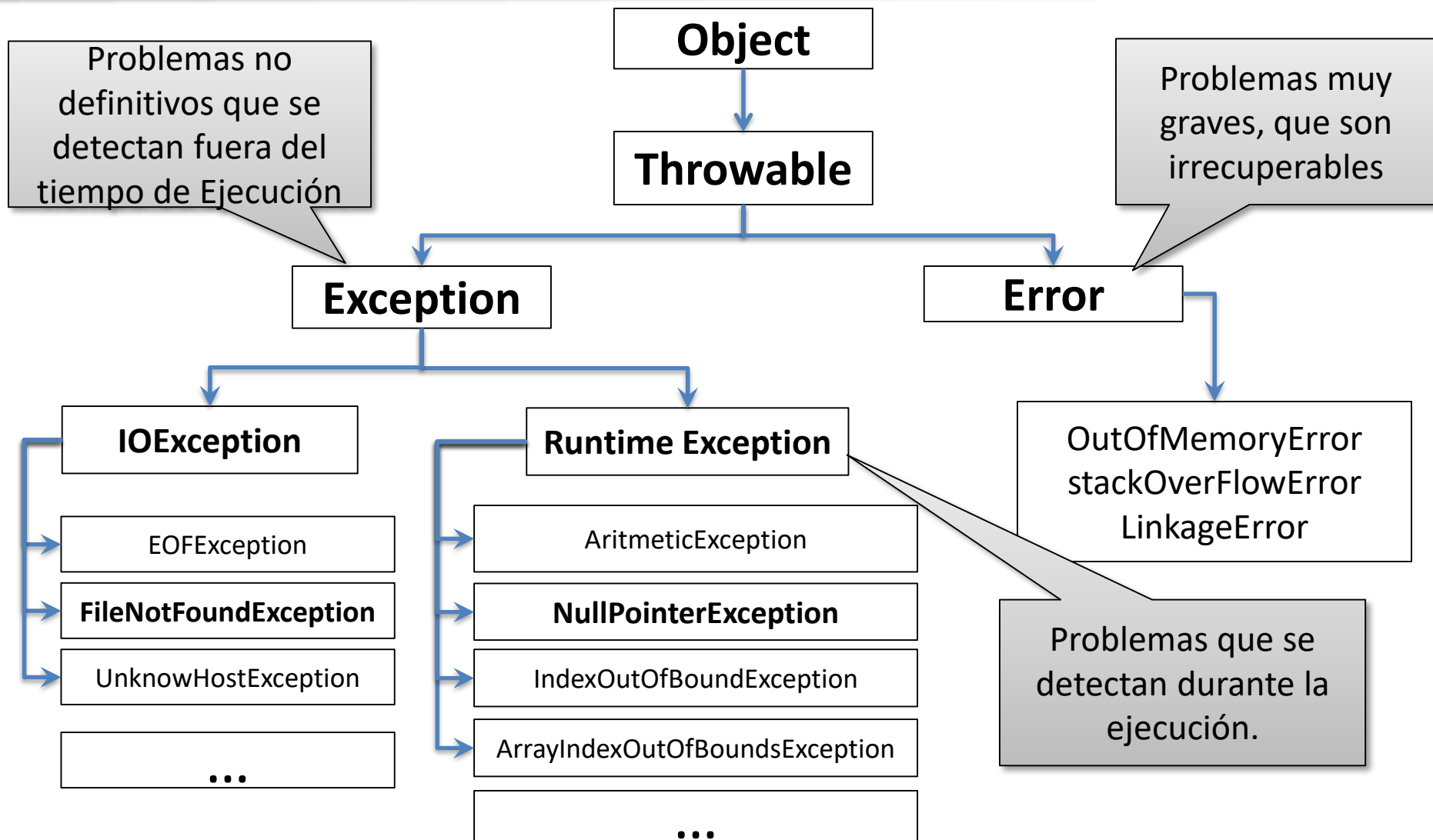
# Concepto Excepción

---

- En Java corresponde a un error de ejecución que se puede controlar.
- Son objetos de clases derivadas de la clase base Exception.



# Jerarquía de Clases de Excepción





# Captura de Excepciones

**Bloque try...catch:** Permite capturar excepciones RuntimeException

```
try{
```

```
//Instrucciones que pueden generar excepciones
```

```
} catch (NombreExcepción1 objeto){
```

```
//Instrucciones para controlar excepción 1
```

```
} catch(NombreExcepción2 objeto){
```

```
//Instrucciones para controlar excepción 1
```

```
} finally{
```

```
/*Instrucciones que se ejecutan se produzca o no alguna  
excepción*/
```

```
}
```



# Ejemplos Manejo de Excepciones

```
Exception in thread "main" java.lang.NullPointerException
    at control.Controlador.inicio(Controlador.java:24)
    at proyempleado.ProyEmpleado.main(ProyEmpleado.java:22)
C:\Users\SONIA\AppData\Local\NetBeans\Cache\8.2\executor-snippets\run.xml:53: Java returned: 1
BUILD FAILED (total time: 0 seconds)
```



# Ejemplos Manejo de Excepciones

Excepción : NullPointerException	
Descripción	Ejemplo
Indica que el objeto es nulo o no tiene apuntador de referencia. Se ha usado una referencia <i>null</i> para acceder a un campo.	<pre>import java.io.*; class Persona{     private String nombre;     private int edad;     public Persona(){     public String getNom() {return nombre;}     }      public class ejClase {     public static void main( String[] args ) {         try {    Persona per=null;                 System.out.println(" "+per.getNom() );             } catch ( NullPointerException e ){                 System.out.println( "Error : " + e.getMessage() );             }         }     } }</pre>
Cuando ocurre	
Cuando se define un objeto o una variable y este no se puede operarse por tener una referencia nula o no estar creado.	
Como se evita	
Se puede debe instanciar el objeto para quitar la referencia nula, para el ejemplo simplemente se utilizaría la siguiente Instrucción: Persona per= new Persona()	



# Ejemplos Manejo de Excepciones

Excepción : FileNotFoundException	
Descripción	Ejemplo
No se encontró el archivo o no existe	<pre>import java.io.*; class ejFile { public static void main( String[] args ) {     FileInputStream entrada = null;     int dato;     try {        entrada = new FileInputStream("valores.txt");                 while ( ( dato = entrada.read() ) != -1 )                     System.out.println( "Dato Entero" + dato );             }         catch ( FileNotFoundException e ){ // Estándar             System.out.println( "Error de FileNotFoundException: " + e.getMessage() );         }         catch ( IOException e ){ // Estándar             System.out.println( "Error de IOException: " + e.getMessage() );         }     } }</pre>
Cuando ocurre	
Al tratar de acceder a un archivo , no se puede abrir o leer.	
Como se evita	
Se debe garantizar que el archivo exista o se halla creado y que la ruta que se coloca en el FileInputStream sea la correcta.	





# Ejemplos Manejo de Excepciones

<b>Excepción : IOException</b>	
<b>Descripción</b>	<b>Ejemplo</b>
Excepcion de entrada y salida	<pre>import java.io.*; class ejFile { public static void main( String[] args ) {     FileInputStream entrada = null;     int dato;     try {         entrada = new FileInputStream("valores.txt");         while ( ( dato = entrada.read() ) != -1 )             System.out.println( "Dato Entero" + dato );     }     catch ( FileNotFoundException e ){ // Estándar         System.out.println( "Error de FileNotFoundException: " + e.getMessage() );     }     catch ( IOException e ){ // Estándar         System.out.println( "Error de IOException: " + e.getMessage() );     } }</pre>
<b>Cuando ocurre</b>	
Se produce un error en alguna operación de entrada o salida del sistema, en este caso se utiliza para verificar la lectura de los datos del archivo, al utilizar el método read.	
<b>Como se evita</b>	
Se debe garantizar que el archivo exista o se halla creado y que la ruta que se coloca en el FileInputStream sea la correcta.	





# Ejemplos Manejo de Excepciones

Excepción : NumberFormatException	
Descripción	Ejemplo
El formato de la cadena no es tipo numérico requerido	<pre>import java.io.*; import java.util.*;  public class Excepciones {     public static void main( String[] args ) {          int numeros[] = { 2,4,6,8,10 },num;         Scanner dato=new Scanner(System.in);          try {             num=Integer.parseInt(dato.next());             for(int i=num;i&lt;=numeros.length;i++)                 System.out.println(" "+numeros[i] );          }         catch( NumberFormatException e ) {             System.out.println( " El numero ingresado no es entero");             System.out.println( "Error de NumberFormatException"+                 e.toString() );         }         catch( ArrayIndexOutOfBoundsException e ) {             System.out.println( "Error de             ArrayIndexOutOfBoundsException"+ e.toString() );         }     } }</pre>
Cuando ocurre	
Cuando se hace una conversión de una cadena a un tipo de dato y el contenido de la cadena no corresponde a ese tipo de dato.	
Como se evita	
Es difícil evitarlo cuando se realizan operaciones de captura de datos, ya que el usuario se puede equivocar al escribir el numero, por lo tanto se utiliza el try – catch para atrapar la excepción y generar el mensaje correspondiente	



# Ejemplos Manejo de Excepciones

Excepción : ArrayIndexOutOfBoundsException	
Descripción	Ejemplo
El índice del arreglo esta fuera de los limites	<pre>import java.io.*; import java.util.*;  public class Excepciones {     public static void main( String[] args ) {          int numeros[] = { 2,4,6,8,10 },num;         Scanner dato=new Scanner(System.in);          try {             num=Integer.parseInt(dato.next());             for(int i=num;i&lt;=numeros.length;i++)                 System.out.println(" "+numeros[i] );          }         catch( NumberFormatException e ) {             System.out.println( "Error de NumberFormatException"+ e.toString() );         }         catch( ArrayIndexOutOfBoundsException e ) {             System.out.println( "Error de ArrayIndexOutOfBoundsException"+ e.toString() );         }     } }</pre>
Cuando ocurre	
Cuando se quiere acceder a una posición invalida de un arreglo	
Como se evita	
Especificar los limites de las posiciones del arreglo o validar la Excepción por medio de try-catch	



# Bibliografía

---

- Documentación clase Programación Multinivel Sonia Pinzón. Disponible en Drive
- Documentación Java Oracle Clase Exception. Disponible en:  
[https://docs.oracle.com/javase/7/docs/api/java x/swing/package-summary.html](https://docs.oracle.com/javase/7/docs/api/java%20x/swing/package-summary.html).
- Deitel y Deitel. Programación Java. Editorial Mc Graw Hill.