# Excepciones III

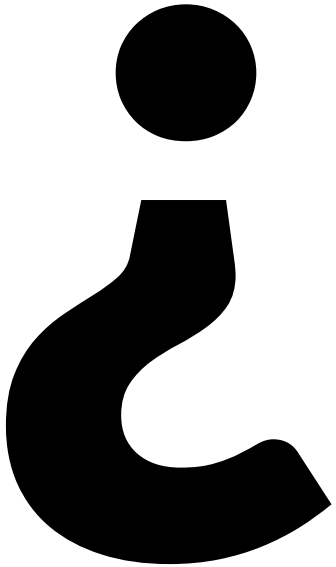
UNRN

Universidad Nacional
de **Río Negro**

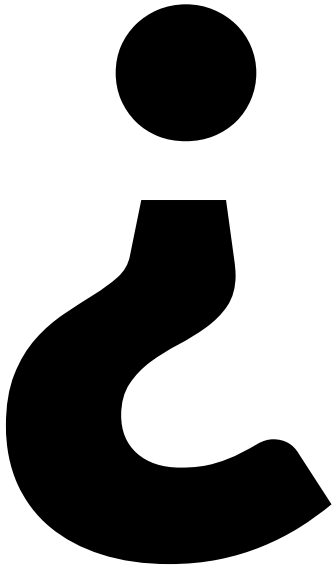
IIX

2024


r15



# Dudas del TP3



# Dudas del TP4


¿Preguntas?

# Documentación de Excepciones

# Etiqueta @throws

```
/**
 *
 * @throws ArregloExcepcion ¿cuando se lanza?
 * @throws IOException ¿Cuando se lanza?
 */
```

# Por ejemplo

```
/**
 *Pide un número entero, con un mensaje personalizado
 * y una cantidad limitada de intentos
 * @throws NoMasIntentosException cuando agotamos los intentos
*/
public static int pideInt(String mensaje, int intentos)
                        throws NoMasIntentosException {
    código que resuelve el ejercicio
}
```

Solo para Exception

# Idealmente, se documenta todo lo que sepan que aparece.


¿Preguntas?

# Creación de Excepciones

# En un archivo separado

```
public class NoMasIntentosException extends Exception{
    public NoMasIntentosException(){
        super();
    }
}
```


**El TP3 tiene un ejemplo más completo.**

# En un archivo separado

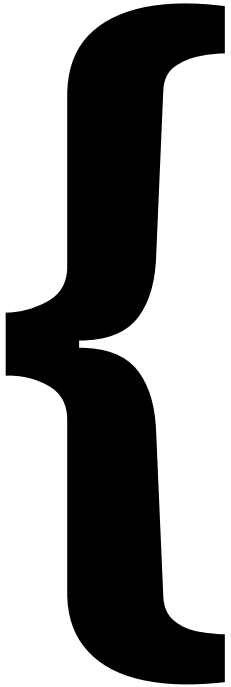
```
public class NoMasIntentosException extends RuntimeException{
    public NoMasIntentosException(){
        super();
    }
}
```


**El TP3 tiene un ejemplo más completo.**

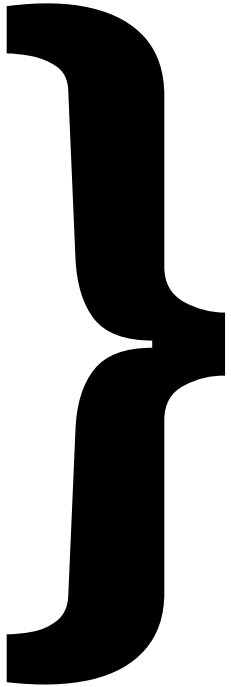
# Recuerden que *puede* ser necesario declarar su uso

```
/**
 *Pide un número entero, con un mensaje personalizado
 * y una cantidad limitada de intentos
 * @throws NoMasIntentosException cuando agotamos intentos
*/
public static int pideInt(String mensaje, int intentos)
                          throws NoMasIntentosException {
    código que resuelve el ejercicio
}
```


¿Preguntas?



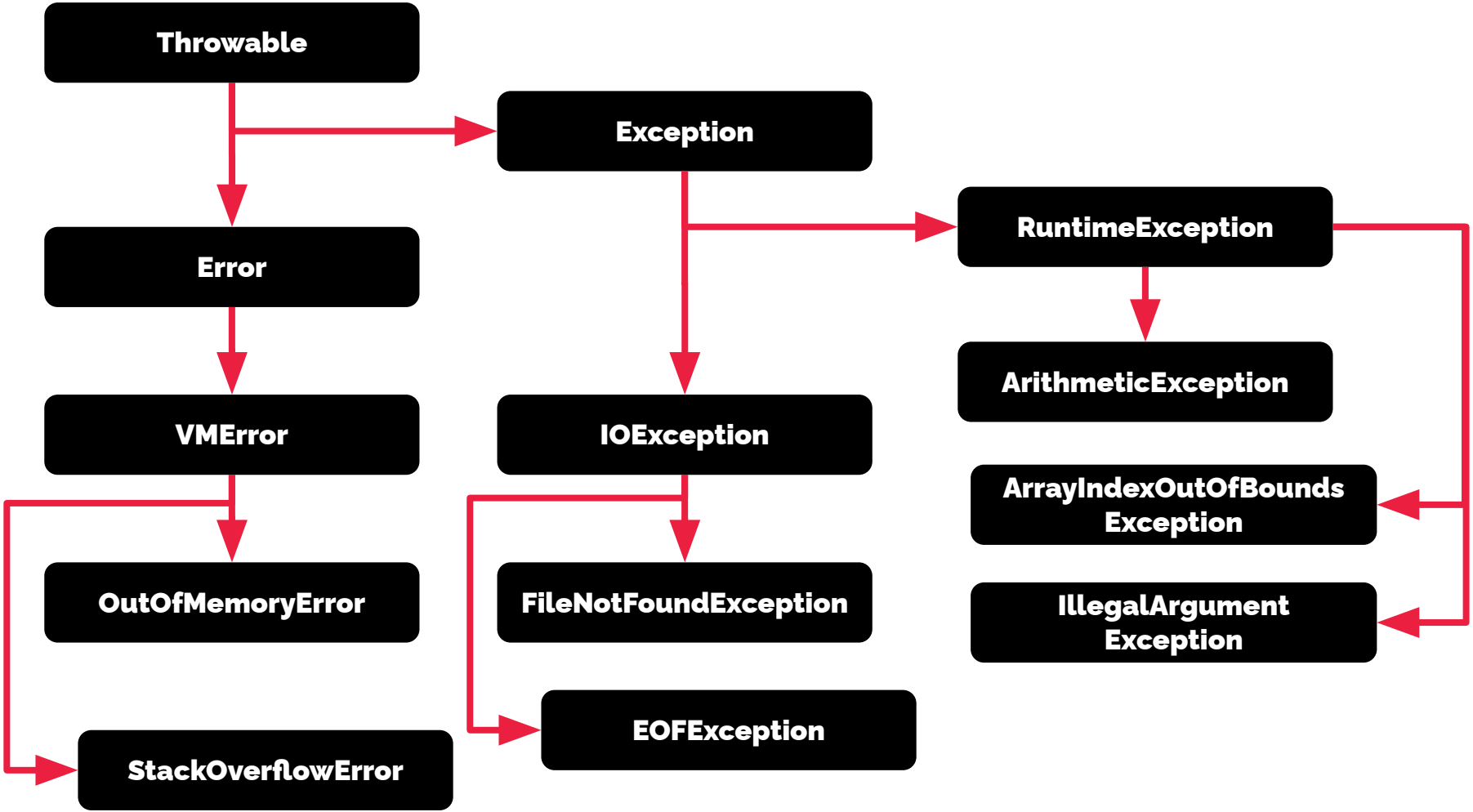
# Tipificación de excepciones

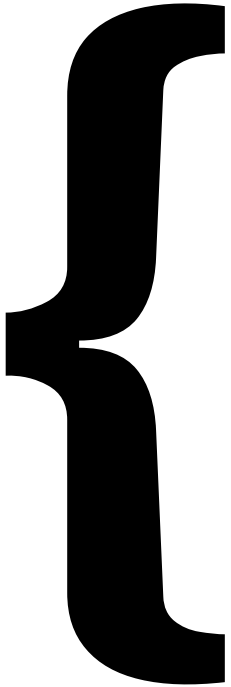


# No todas las situaciones son iguales

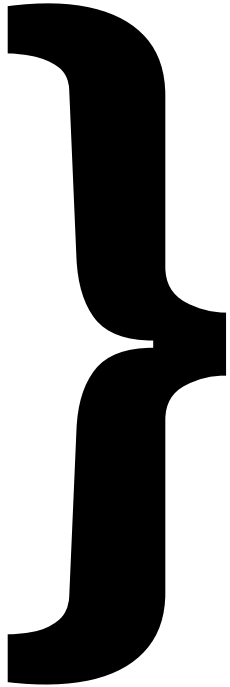
# Podemos clasificarlas para darles diferentes tratamientos

# 'clasificación' de las excepciones


Throwable
Exception
Error
RuntimeException
VMError
IOException
ArithmeticException
ArrayIndexOutOfBounds Exception
OutOfMemoryError
FileNotFoundException
IllegalArgument Exception
EOFException
StackOverflowError



# Esto es una relación "es un"

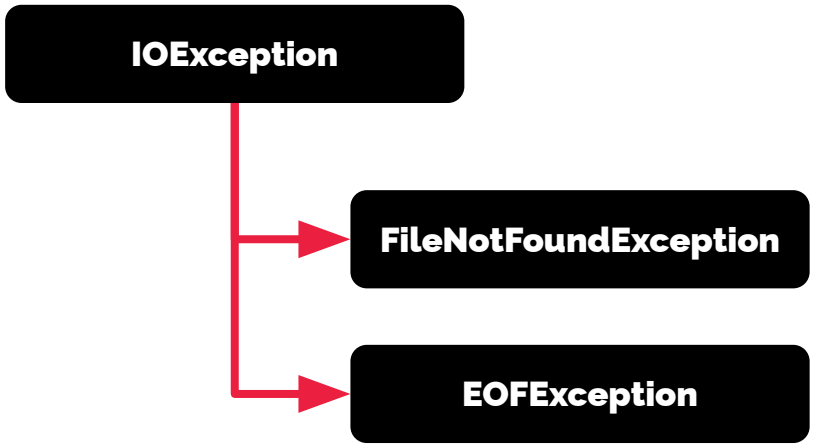


`FileNotFoundException` es una clase de `IOException`

# Al atajar, lo hacemos desde donde indicamos hacia abajo.

# Dentro de la misma familia

# Gestión especializada




Podemos tratar todo de la misma manera como IOException

Pero podemos gestionar puntualmente los tipos de excepción.

1

7

# Atajen solo lo que pueden procesar

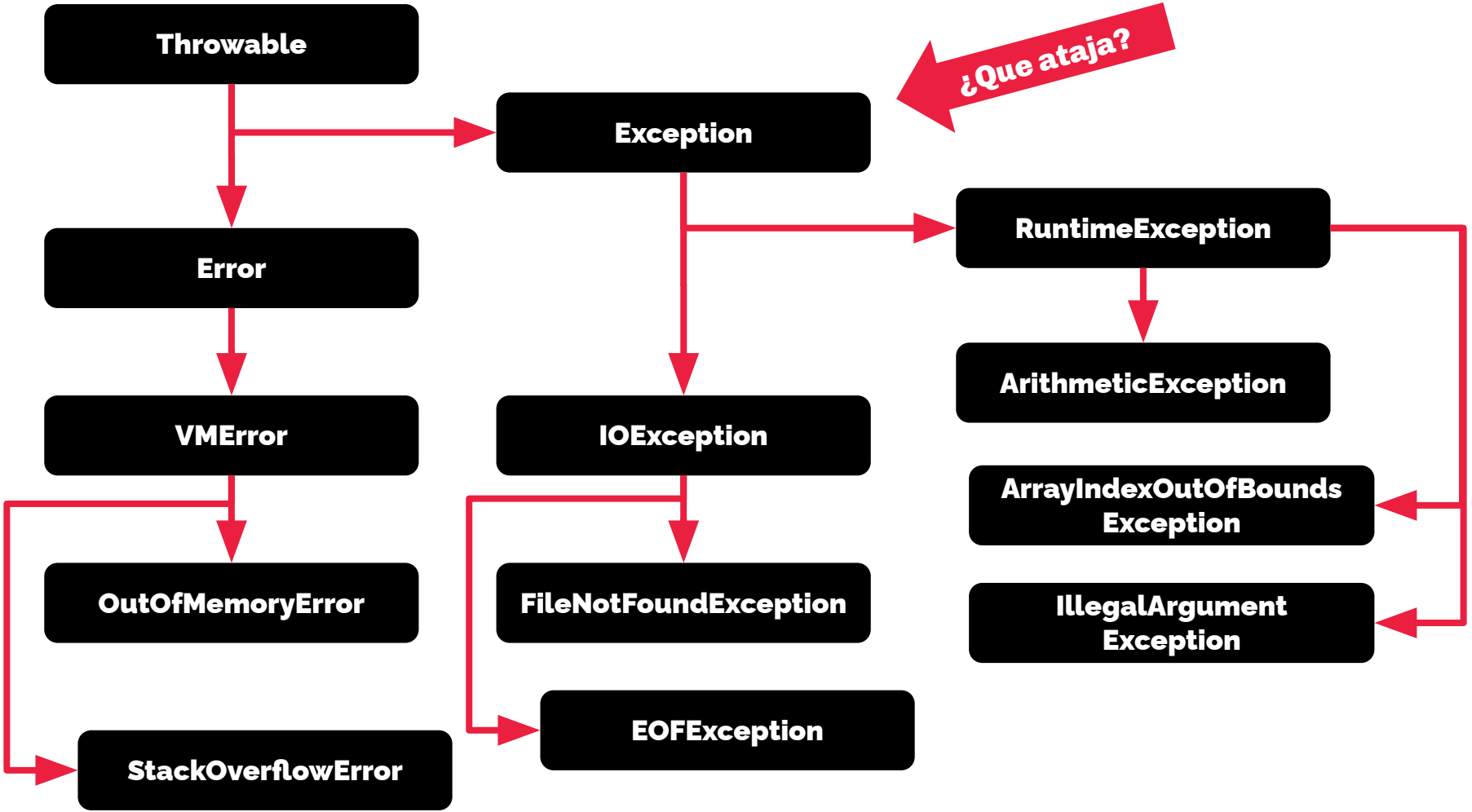
18

# Prefieran excepciones 'con tipo' o justifiquen por qué.


¿Preguntas?

# ¿Que pasa en este ejemplo?

```
try {
    código con lanzamiento de excepción
} catch (Exception exc) {
    ¿Que se ataja en este punto?
}
```


Throwable
Exception
¿Que ataja?
Error
RuntimeException
VMError
IOException
ArithmeticException
ArrayIndexOutOfBounds Exception
OutOfMemoryError
FileNotFoundException
IllegalArgument Exception
EOFException
StackOverflowError

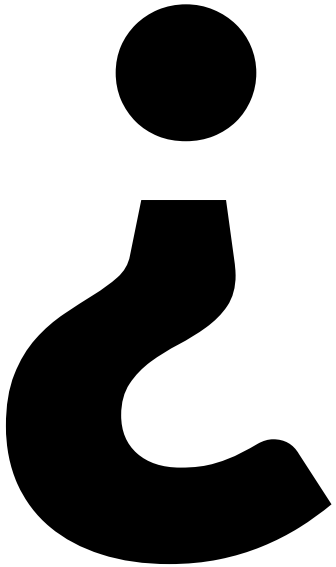
# Detalles sobre la jerarquización

```
try {
    código con lanzamiento de excepción
} catch (Exception exc) {
    ¿Que se ataja en este punto?
} catch (ArithmeticException exc) {
    ¿y acá?
}
```

# 19

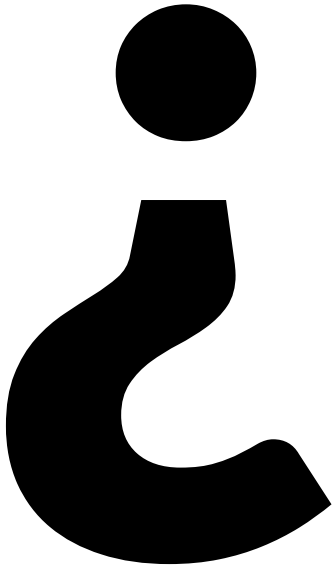
# Atajar solo para un `print` no es gestionar la excepción


¿Preguntas?



# Qué es eso de 'extends'

# El primer contacto con Orientación a Objetos



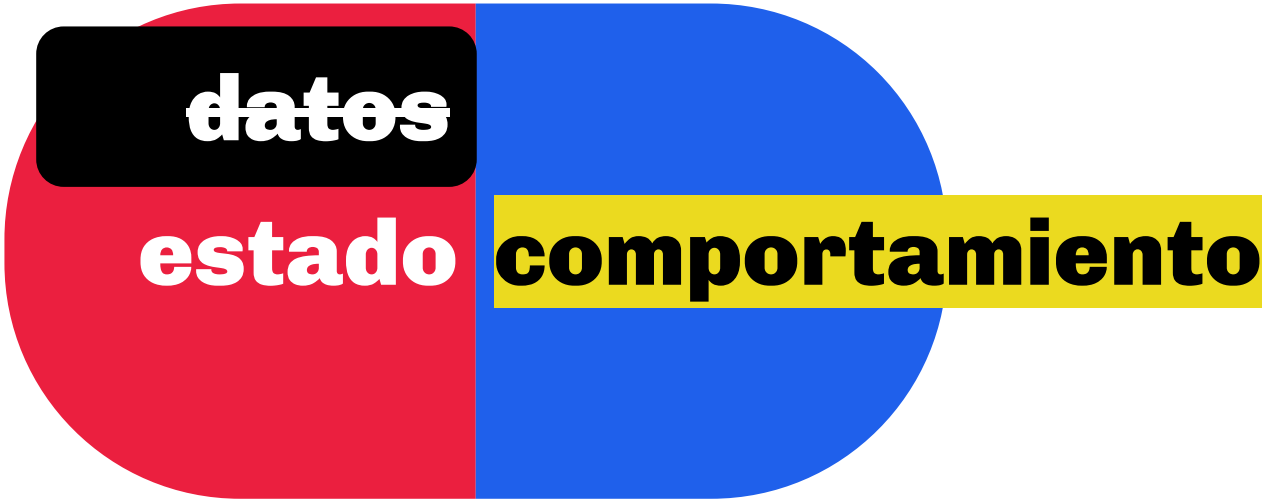
# que es un objeto


datos
Unstruct


funciones


operaciones

# un objeto es




# juntos

# ¿Y las clases?

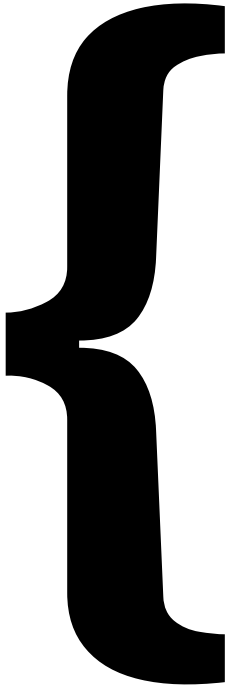
# la clase


# como el “molde” de los objetos

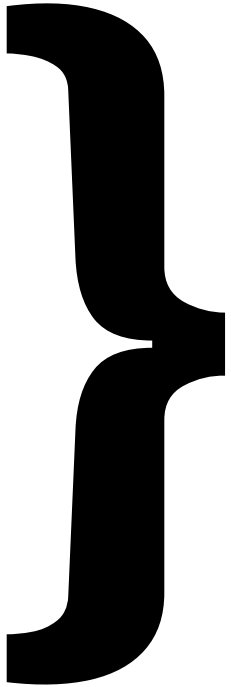
# Y un objeto es una instancia de una clase


¿Preguntas?

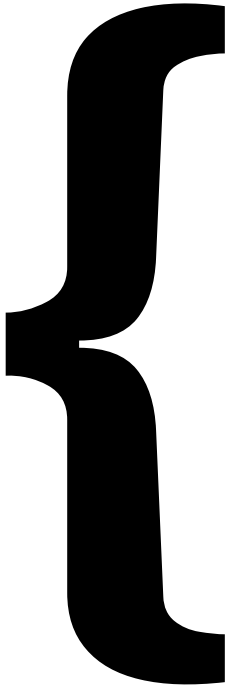
# Desde otro punto de vista...



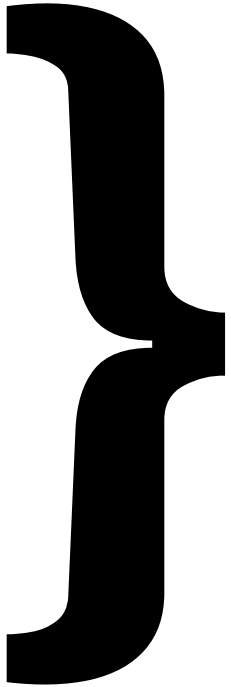
# La clase es el concepto



# Dinero
# Fecha
# Calzado



# El objeto es la cosa



# 1000 pesos
# 3 de marzo 2023
# Un par de zapatillas

# Por lo que tambien podemos decir

# 1000 pesos es un Dinero

# 3 de marzo es una Fecha

# Un par de zapatillas son un Calzado

# Y no solo uno, uno en *particular*

# Podemos decir que las clases definen tipos de cosas

# Podemos decir que las clases definen ~~tipos~~clases de cosas


¿Preguntas?

# Esto sería equivalente a un `struct`.

```
public class Hora {
    int segundo;
    int minuto;
    int hora;
}
```

```
typedef struct {
    int segundo;
    int minuto;
    int hora;
} hora_t;
```

**Puro estado, sin "comportamiento"**

# Ciclo de vida


```
Hora actual = new Hora();
actual.segundo = 10;
actual.minuto = 59;
actual.hora = 1;
```

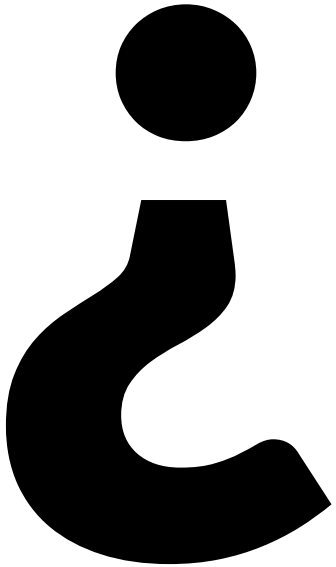
**Funcionando igual que un `struct`.**

# Con un poco de comportamiento

```
public class Hora {
    public int segundo;
    public int minuto;
    public int hora;
    public int sumarSegundos(int segundo){...}
}
```

**Para mantener el estado "consistente" con sus reglas.**

lo que iría en el javadoc



# Que debiera de hacer

```
public int sumarSegundos(int segundo)
```

# Ciclo de vida

```
Hora actual = new Hora();
actual.segundo = 410;
actual.minuto = 159;
actual.hora = 1;
```


**¿Es correcta la información en `actual`?**

# Algo más orientado a objetos, limita el acceso

```
public class Hora {
    private int segundo;
    private int minuto;
    private int hora;
... manipulaciones
... accesores
}
```

**Pero ahora necesita sí o sí comportamiento**

# El comportamiento son "métodos"

¿Y static?

```
class Tiempo{
    public Tiempo(int hora, int minutos, int segundos);
    public Tiempo(int segundos);
    public Tiempo sumar(int segundos);
    public Tiempo sumar(Tiempo otro);
    public Tiempo restar(int segundos);
    public Tiempo restar(Tiempo otro);
    public int comparar(Tiempo otro);
    public String toString();
}
```

# En C (y en Python)

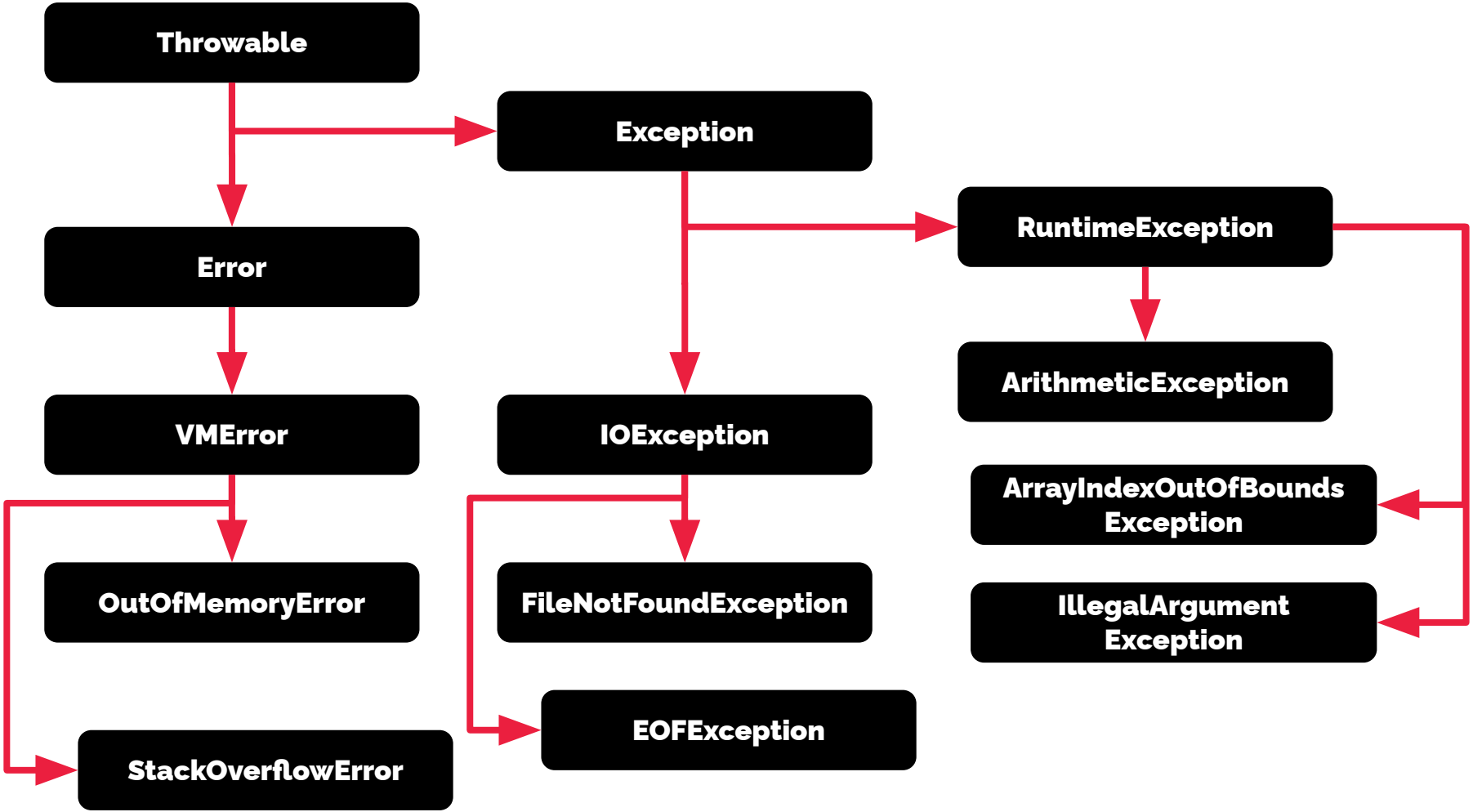
```
int comparar(tiempo_t inicio, tiempo_t fin)
```

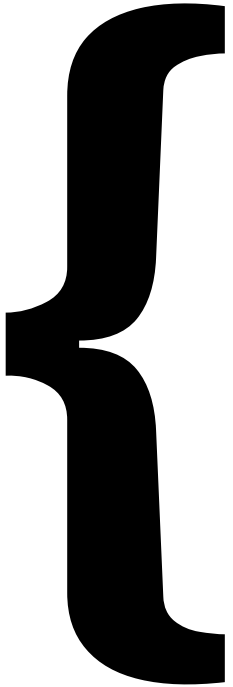
# Otro concepto importante

# Esta es la magia de la Orientación a objetos

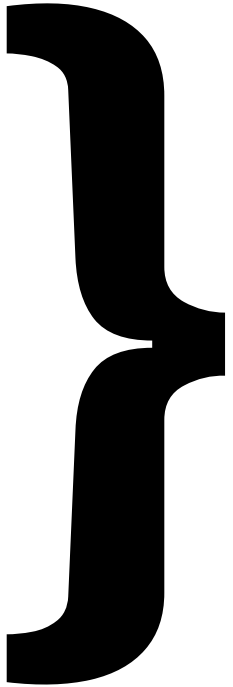
```
public class Duracion extends Tiempo{
    private int dias;
... manipulaciones para 'dias'
... accesores para 'dias'
... Y ya está todo lo de Tiempo
}
```

**Una "Duración" es un "Tiempo"**


Throwable
Exception
Error
RuntimeException
VMError
IOException
ArithmeticException
ArrayIndexOutOfBounds Exception
OutOfMemoryError
FileNotFoundException
IllegalArgument Exception
EOFException
StackOverflowError

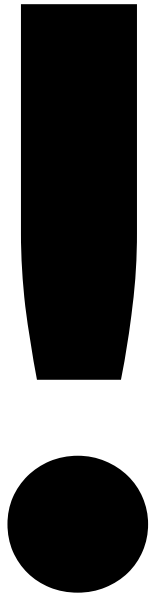


# Esto como primer contacto



# Qué en la próxima clase le vamos a dar 'nombres' a estos conceptos

# Recién empezamos



unrn.edu.ar

UNRN Universidad Nacional de Río Negro

| unrionegro