Programación Orientada a Objetos

Tener en cuenta…

- Ciclo 3 – Proyecto Inicial

  Sábado 01 de Octubre

- Revisión a par Proyecto inicial – Ciclo 3

  Semana del 3 al 9 de Octubre

- Laboratorio 3

  Sábado 08 de Octubre

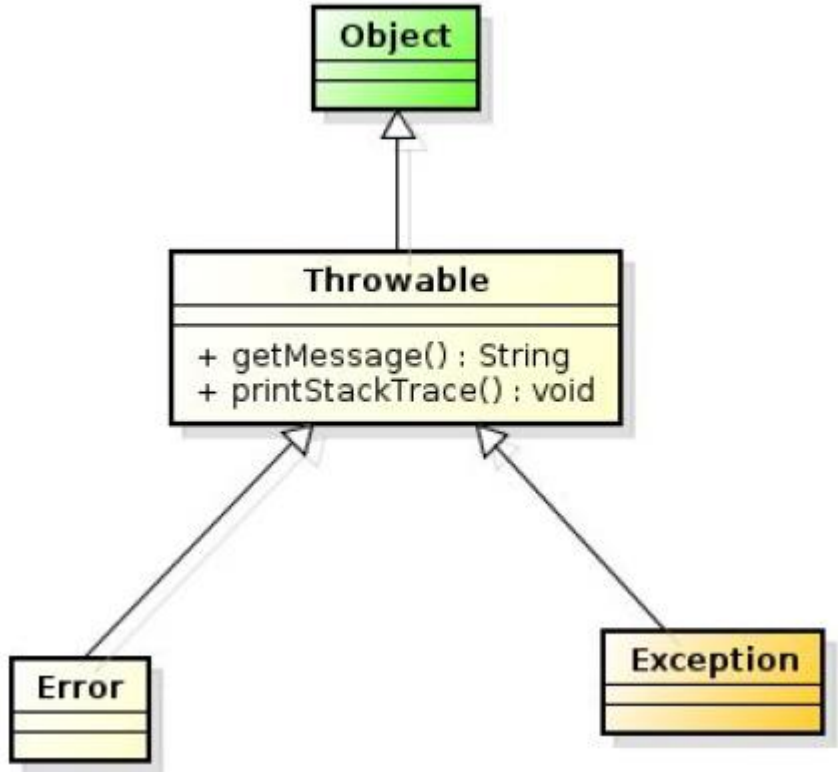
Excepciones

Class Exception

java.lang.Object
    java.lang.Throwable
        java.lang.Exception

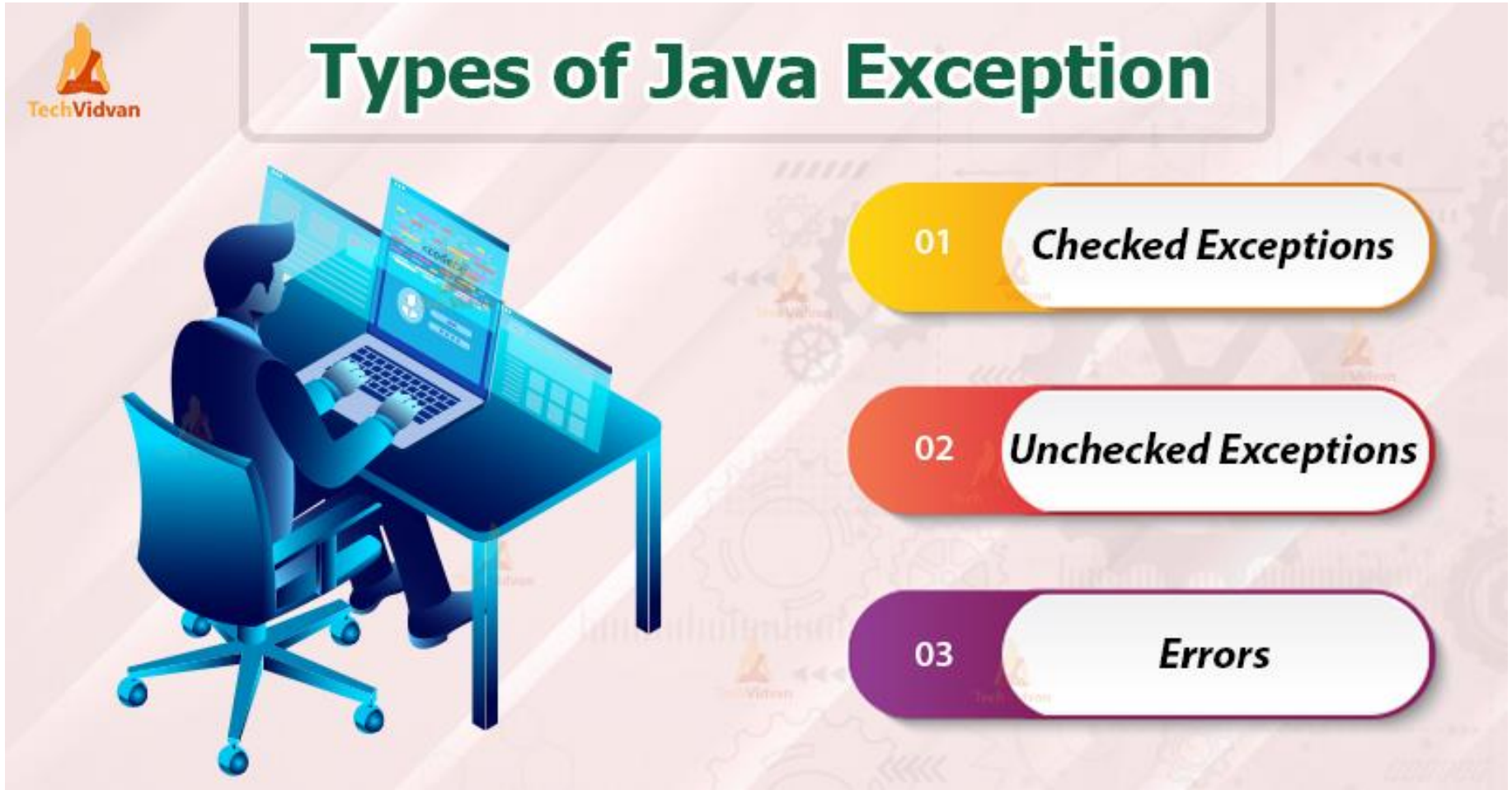
All Implemented Interfaces:

Serializable



Situación que altera al flujo normal de un programa en ejecución

Tipos de excepciones



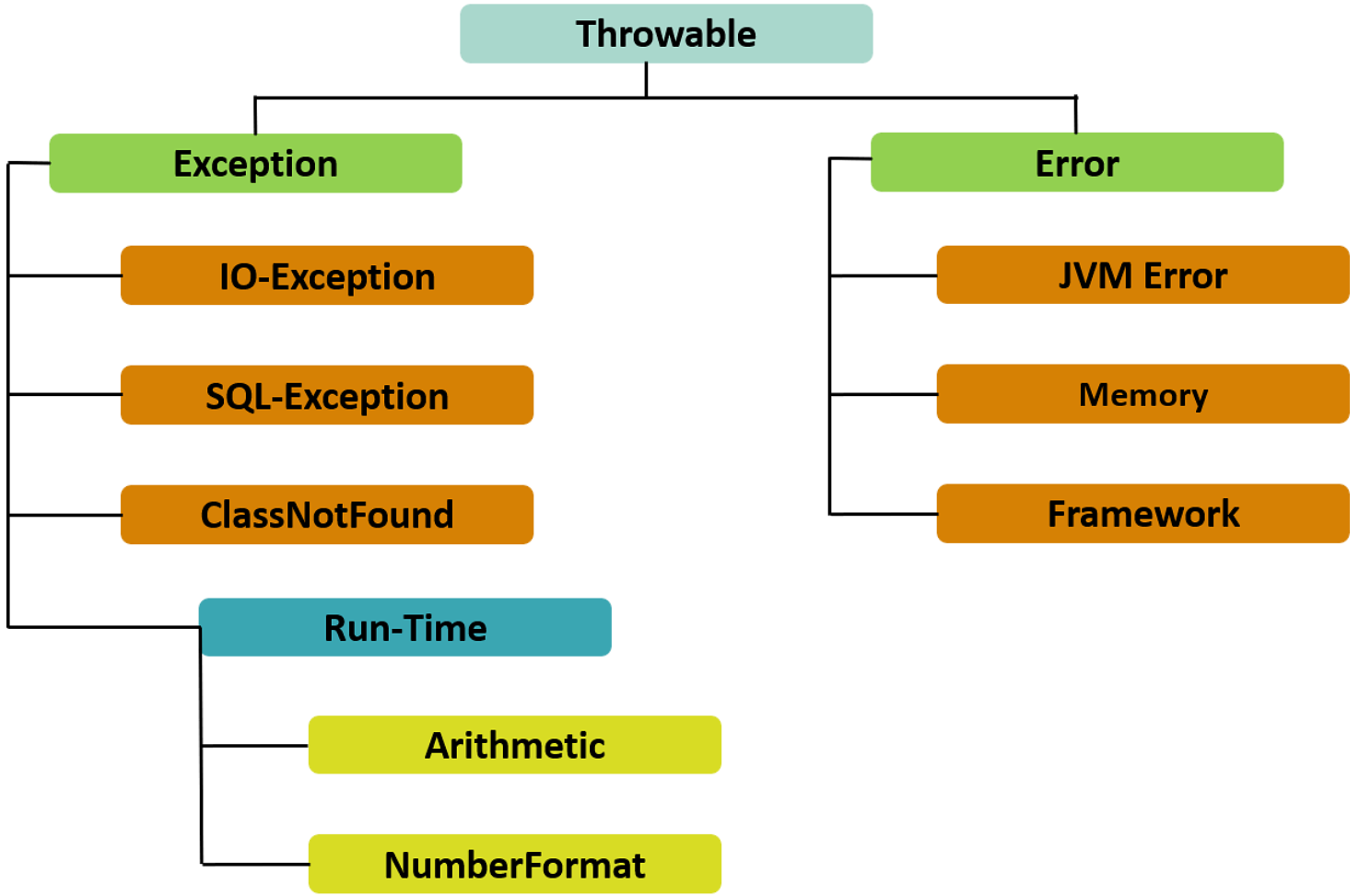
Compilación

Ejecución

Condición No Recuperable

Tomado de: https://techvidvan.com/tutorials/java-exception/

Tipos de excepciones



Tomado de: https://www.simplilearn.com/tutorials/java-tutorial/exception-handling-in-java

Nueva clase de excepción

```java
public class MissingValueException extends Exception {

    public MissingValueException (String message){
        super(message);
    }
}
```

Lanzar excepción

```
public void setName(String n) {

        // we want to report an error if the string that has
        // been passed is in blank.
        if(n.equals( "")){
            throw new MissingValueException( "A student's name can not be blank.");
        }else{
            name = n;
        }
}
```

Propagar excepción

```java
public void setName(String n) throws MissingValueException{

    // we want to report an error if the string that has
    // been passed is in blank.
    if(n.equals("")){
        throw new MissingValueException("A student's name can not be blank.");
    }else{
        name = n;
    }
}
```

```java
public void update(String n, String s){
        setName(n);
        setSsn(s);
        setMajor("UNDECLARED");
}
```

Atender excepción

```
public class ExceptionClassExample{

    private Student student;

    public static void main(String[] args) {
        try{
            student.setName("laura");
        }catch(MissingValueException e){
            System.out.println(e.getMessage());
        }
    }
}
```

Palabras reservadas

Palabra	Nivel	Objetivo
Extends	Clase	Herencia
Final	Clase	Nivel máximo de especialización
Final	Método	No se permite sobreescritura
Abstract	Clase	No se puede instanciar
Abstract	Método	Las subclases tienen que tener implementado el método
Interface	Clase	Es abstracta y establecen la forma que debe tener una clase. Define qué y no el cómo *.
Implements	Clase	Una clase implementa una interfaz
Throw	**Método**	**Lanzar excepción**
Throws	**Método**	**Propagar excepción**
Try / Catch	**Método**	**Controlar excepción**