java 8

Optional, la alternativa a NULL.

Optional aparece en java 8 para evitar los nullPointerExcption que aparecen en la ejeución. Con la introducción de optional, se obliga al desarrollador a pensar que un objeto o una función puede tener o no algún objeto.

Para controlar que un ordenador tiene tarjeta de sonido y que esta tiene puerto USB, el código quedaría:

String version = "UNKNOWN";

if(computer != null){

Soundcard soundcard = computer.getSoundcard();

if(soundcard != null){

USB usb = soundcard.getUSB();

if(usb != null){

version = usb.getVersion();

}

}

}

La alternativa con optional queda:

class Computer {

private Optional<Soundcard> soundcard;

public Optional<Soundcard> getSoundcard() {

return soundcard;

};

}

class Soundcard {

private Optional<USB> usb;

public Optional<USB> getUSB() {

return usb;

}

}

class USB{

public String getVersion(){

return "v1.0";

}

}

String name = computer.flatMap(Computer::getSoundcard)

.flatMap(Soundcard::getUSB)

.map(USB::getVersion)

.orElse("UNKNOWN");

Creación de objetos Optional

En primer lugar, ¿cómo se crean objetos Optional? Existen diversos modos:

Este es un Optional vacío:

Optional<Soundcard> sc = Optional.empty();

Y este es un Optional con un valor no nulo:

SoundCard soundcard = new Soundcard();

Optional<Soundcard> sc = Optional.of(soundcard);

Si soundcard fuera nulo, se arrojaría de inmediato una excepción NullPointerException (en lugar de obtener un error latente cuando se intente acceder a las propiedades de soundcard).

Asimismo, utilizando ofNullable, es posible crear un objeto Optional que puede contener un valor nulo:

Optional<Soundcard> sc = Optional.ofNullable(soundcard);

Si soundcard fuera nulo, el objeto Optional resultante estaría vacío.

Valores predeterminados y acciones:

Soundcard soundcard =

maybeSoundcard != null ? maybeSoundcard

: new Soundcard("basic\_sound\_card");

Si se usa un objeto Optional, es posible reescribir el código anterior empleando el método orElse(), que proporciona un valor predeterminado si Optional está vacío:

Soundcard soundcard = maybeSoundcard.orElse(new Soundcard("defaut"));

//soundcar con objeto null

Soundcard soundcard2 = null;

Optional<Soundcard> scNull = Optional.ofNullable(soundcard2);

scNull.ifPresent(System.out::println);

Soundcard soundcard3 = scNull.orElse(new Soundcard("defaut"));

System.out.println(soundcard3.name);