Si lanzamos una depracion los mensajes aparecerán en una pantalla llamada lovecat, en esta consola al pulsar log sirve para facilitar los mensajes

((button)findViewById(boton1)  
((button)findViewById(N.id,boton1.setText("Hola Mundo"):

Los.on(

Metemos el botn en una variable

Button botonPulsar =((button)findViewById(N.id,boton1.setText("Hola Mundo"):

Al pulsar le ponemos un listener

Le pondresmo el setonclikclistener

Botonpulsar.setonclicklistene();

Dentro log.d();

Lo primero q vemos es que onclicklistener es una interfaz

(lo que significa que tendremos que implementar los métodos de este listener.

Creo una clase nueva gestor de clicks (no es la mejor opción, de hecho no se hace) y implementamos el método que requiera en este caso el método onclik en otra clase

Log.on

Otra opcion es:

Botonpulsar.setonclicklistene( new view.onclikcListene{

@override

public class ...

});

Propiedad onclick

Tercera opción -> Creamos un método en las propiedas del botón, en la vista lo podemos poner

Cuarta opción -> crear un métodoonclicklistener

View.oncliklistene gestor = new view.onclikcListene{

@override

public class ...

};

Quinta opción -> con la expresión landa, que solo se pueden utilizar con interfaces con solo un método para desarrollar, como la que estamos utilizando

Botonpulsar.setonclicklistene( (boton) -> {log.d()});

Casteando a buton

Crear unoproyecto nuevo, casilla para insertar nombre y el pulsar el botón se añadira Hola mas los que se escribía en el nombre, esto se ira sumando, también se imprime por consola en kotlin y java

Findbyid, setonclicklistener, declarar botón, botón.text = “pulsado”

Teoría kotlin, la interrogación es porque kotlin evita el NullPointedException, bloqueando todos los objetos no declarados

Casteo kotlin findon…<Button>(B.id… o findon…as Button .setLis…