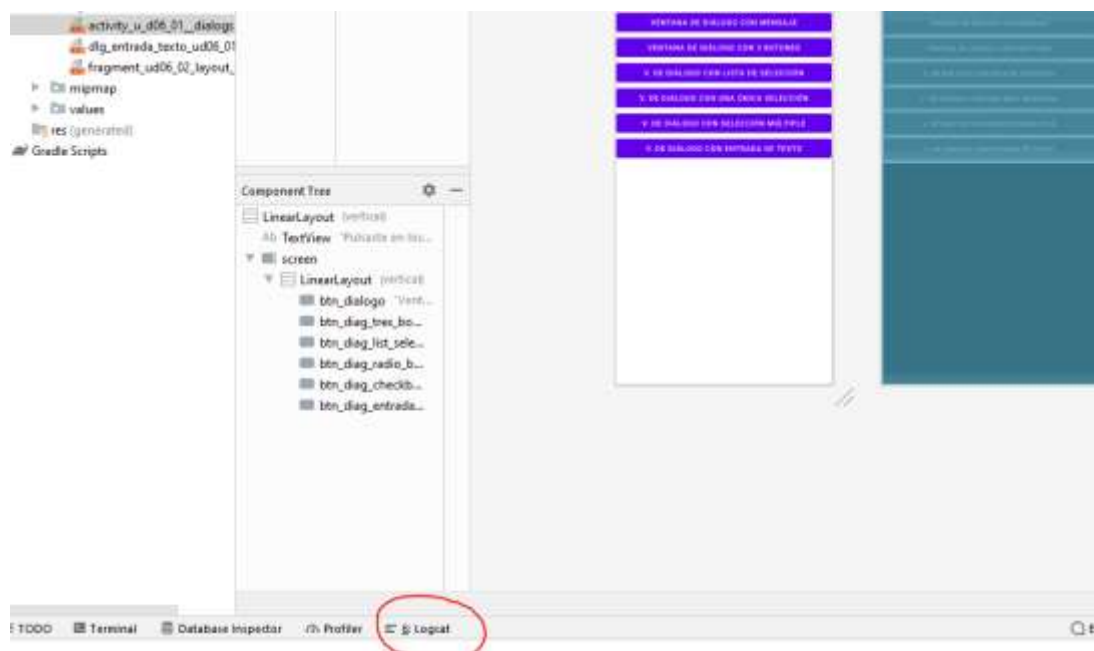


INDICE

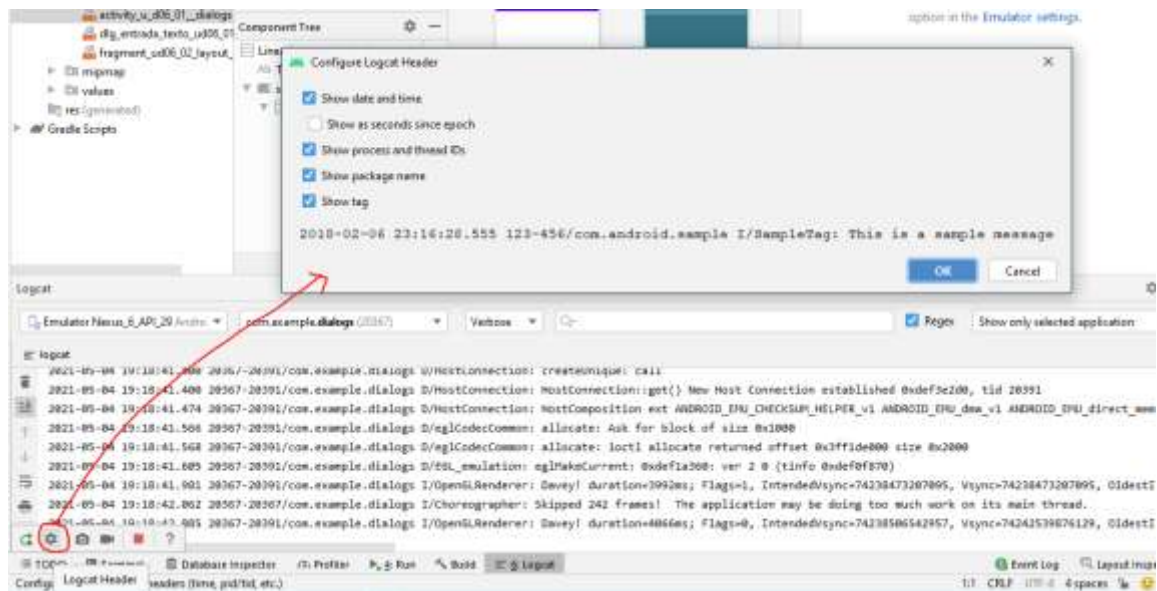
1. INTRODUCCIÓN	1
2. CASO PRÁCTICO	2
2.1. EL XML DEL LAYOUT	4
2.2. EL CÓDIGO JAVA DE LA APLICACIÓN	5

1. Introducción

- ✓ Más información [en este enlace](#).
- ✓ Una de las técnicas para realizar depuración y seguimiento de las aplicaciones es la creación de logs (registros) en tiempo de ejecución.
- ✓ En Android se usa la clase **Log**, que posee un conjunto de métodos estáticos que nos van ayudar en el proceso de depuración/seguimiento.



- ✓ La imagen muestra un conjunto de mensajes. Los campos que conforman el mensaje se pueden escoger como se muestra en la siguiente imagen:



- ✓ Los mensajes pueden tener diferentes color en función del nivel de gravedad, así estarán en rojo los que sean de tipo **Error**.
 - Nivel de criticidad: tipo de mensaje (Error, Warning, Info, Debug, etc)
- ✓ Para cada nivel de criticidad existe un método estático que nos ayuda a clasificar los mensajes que puede producir la aplicación: **e()**, **w()**, **i()**, **d()** e **v()**.
- ✓ Cada uno de los métodos recibe como parámetro la etiqueta (**tag**) y el texto del mensaje.
- ✓ **Referencias:**
 - La Clase Log: <http://developer.android.com/reference/android/util/Log.html>

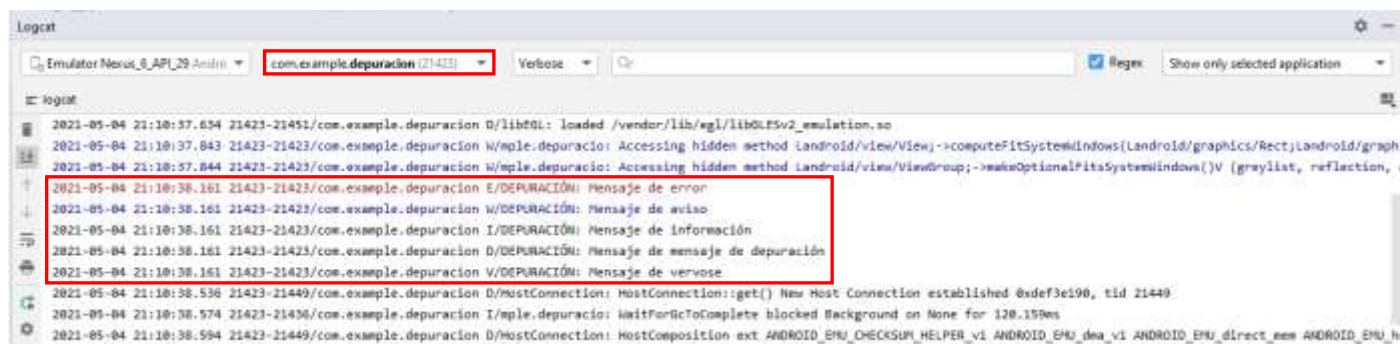
2. Caso práctico

- ✓ Crear el siguiente proyecto: **Depuracion**

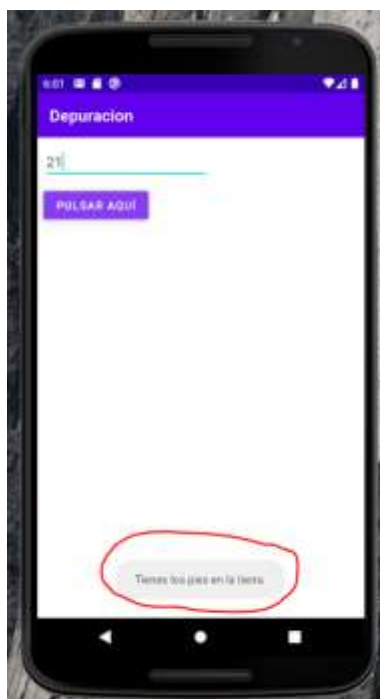
Una activity de nombre: **"UD07_02"**

- ✓ Vamos a realizar un pequeño proyecto donde mostrar como poder usar el LOG.
- ✓ La etiqueta (**tag**) que se está usando en el programa es: "DEPURACIÓN"
- ✓ Para ver los mensajes de log usaremos la ventana **LogCat**.

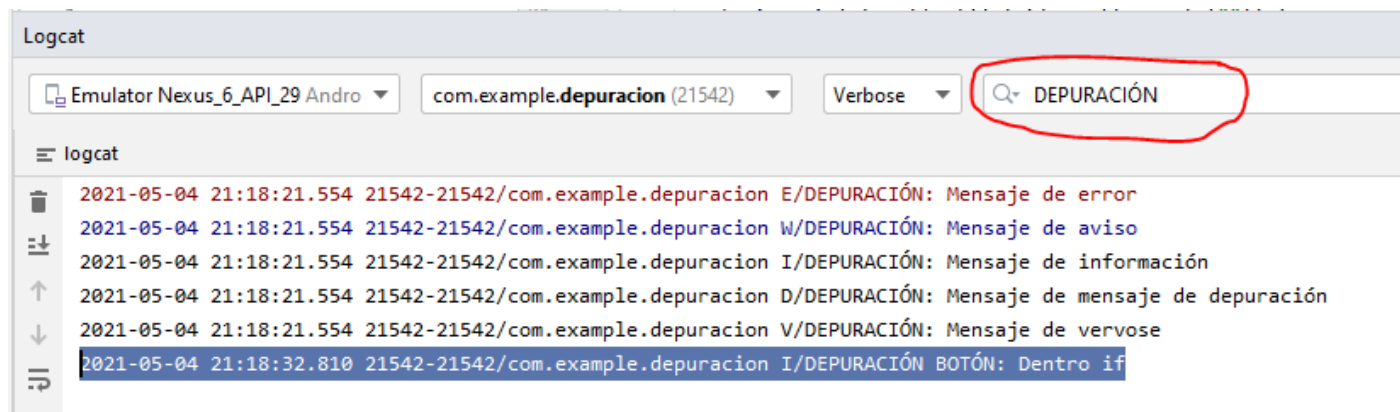
Depuración



Al iniciar la aplicación (el código está puesto a continuación) aparecerán mensajes con los diferentes niveles de criticidad. Estos fueron creadas dentro del método onCreate().



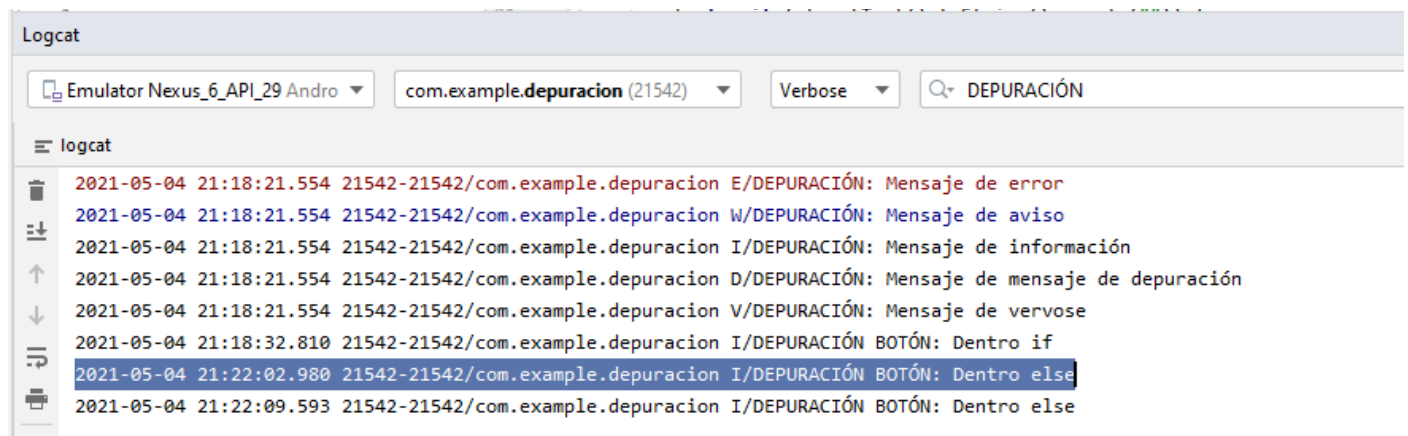
Escribimos un valor correcto en la aplicación.



En logcat podemos buscar por parte del texto del mensaje, luego buscaremos por el tag. Veamos por dónde 'anda' la aplicación...



Escribimos un valor incorrecto.



Y volvemos a ver por donde 'anda' la aplicación....

2.1. El XML del Layout

✓ A continuación se muestra el Layout de la aplicación:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".UD07_02">
```

```

<EditText
    android:id="@+id/edtSiglo"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginTop="8dp"
    android:ems="10"
    android:inputType="textPersonName"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    tools:text="Introduce el siglo en el que vives" />

<Button
    android:id="@+id/btnPulsar"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginTop="8dp"
    android:text="Pulsar aquí"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/edtSiglo"
    android:onClick="onButtonClick"
/>

</androidx.constraintlayout.widget.ConstraintLayout>

```

2.2. El código Java de la aplicación

```

package com.example.depuracion;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class UD07_02 extends AppCompatActivity {

    public void onButtonClick(View v) {
        EditText et = (EditText) findViewById(R.id.edtSiglo);

        if (et.getText().toString().equals("21")) {
            Toast.makeText(this, "Tienes los pies en la tierra ", Toast.LENGTH_SHORT).show();
            Log.i("DEPURACIÓN BOTÓN", "Dentro if");
        } else if (et.getText().toString().equals("")) {
            Toast.makeText(this, "No escribiste nada", Toast.LENGTH_SHORT).show();
            Log.i("DEPURACIÓN BOTÓN", "Dentro else if");
        } else
        {
            Toast.makeText(this, "Estás en otro planeta", Toast.LENGTH_SHORT).show();
            Log.i("DEPURACIÓN BOTÓN", "Dentro else");
        }
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }
}

```

```
setContentView(R.layout.activity_u_d07_02);

Log.e("DEPURACIÓN", "Mensaje de error");
Log.w("DEPURACIÓN", "Mensaje de aviso");
Log.i("DEPURACIÓN", "Mensaje de información");
Log.d("DEPURACIÓN", "Mensaje de mensaje de depuración");
Log.v("DEPURACIÓN", "Mensaje de verbose");
    }
}
```