

EJERCICIO KOTLIN LISTENER

1.-java-main:

```
class MainActivity : AppCompatActivity(), View.OnClickListener {  
  
    private lateinit var binding: ActivityMainBinding  
  
    Edit | Explain | Test | Document | Fix  
    override fun onCreate(savedInstanceState: Bundle?) {  
  
        super.onCreate(savedInstanceState)  
        enableEdgeToEdge()  
  
        binding = ActivityMainBinding.inflate(layoutInflater)  
        setContentView(binding.root)  
  
        binding.calcular.setOnClickListener(this)  
  
    }  
}
```

Establecemos el binding y asignamos el método setOnClickListener al botón calcular.

```

override fun onClick(v: View?) {

    when (v?.id) {
        R.id.calcular -> {

            Log.d( tag: "Boton calcular", msg: "Botón pulsado")

            val inputMath = binding.editTextTextMatematicas.text.toString().toIntOrNull()
            val inputFisica = binding.editTextTextFisica.text.toString().toIntOrNull()
            val inputQuimica = binding.editTextTextQuimica.text.toString().toIntOrNull()

            if ( inputMath == null || inputFisica == null || inputQuimica == null ) {
                Toast.makeText(
                    context: this,
                    text: "Por favor, introduce valores numéricos",
                    Toast.LENGTH_SHORT
                ).show()
                return
            } else if (inputMath < 0 || inputMath > 100 || inputFisica < 0 || inputFisica > 100 || inputQuimica < 0 || inputQuimica > 100) {
                Toast.makeText(
                    context: this,
                    text: "Por favor, introduce valores entre 0 y 100",
                    Toast.LENGTH_SHORT
                ).show()
                return
            }

            val media = (inputMath + inputFisica + inputQuimica) / 3

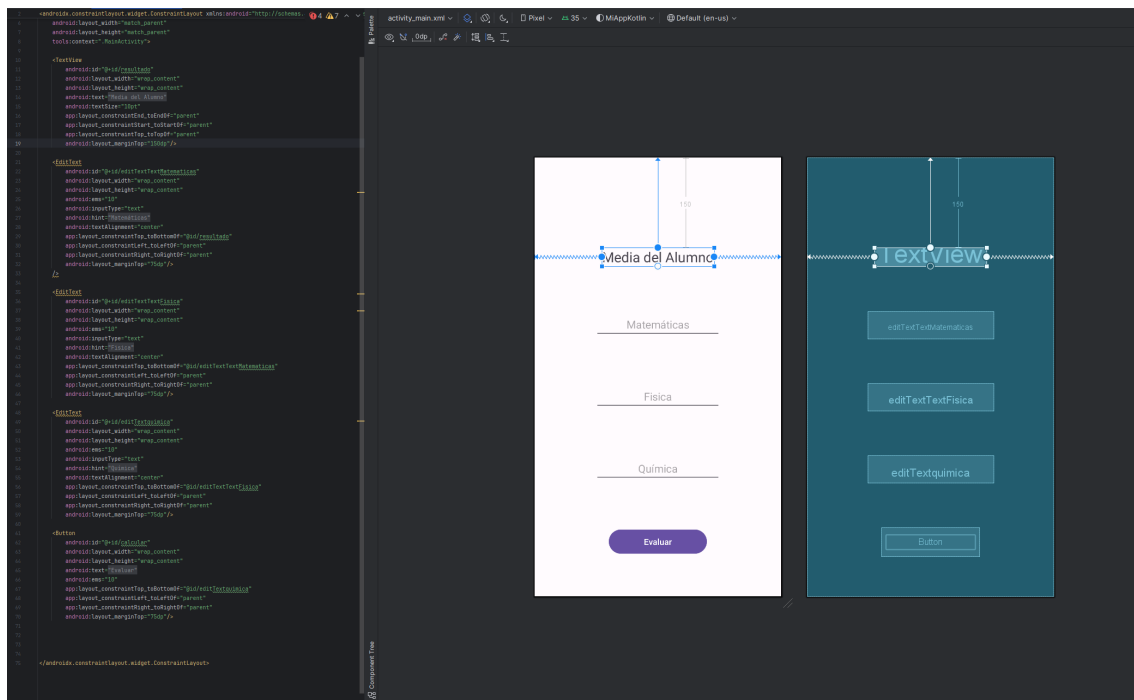
            binding.resultado.setText("Tu media es: $media")

        }
    }
}

```

Sobrescribimos el método onClick para que compruebe los valores en los campos de texto, compruebe si los valores son nulos y si están entre los valores que se esperan y si todo sale bien cambiará el textView resultado a “Tu media es: <media>”


2.-layout-activity_main:



En el layout se han añadido los elementos necesarios que se requerían para el ejercicio, tres editText, uno por asignatura, un textView que cogerá el valor del resultado cuando se calcule y un botón que al pulsarlo realiza la acción de calcular la media y cambiar el textView por el texto que le indiquemos.

3.-gradle:

```

1  plugins {
2      alias(libs.plugins.android.application)
3      alias(libs.plugins.kotlin.android)
4  }
5
6  android {
7      namespace = "es.jmarara529.miappkotlin"
8      compileSdk = 35
9
10     defaultConfig {
11         applicationId = "es.jmarara529.miappkotlin"
12         minSdk = 24
13         targetSdk = 35
14         versionCode = 1
15         versionName = "1.0"
16
17         testInstrumentationRunner = "androidx.test.runner.AndroidJUnitRunner"
18     }
19
20     buildTypes {
21         release {
22             isMinifyEnabled = false
23             proguardFiles(
24                 getDefaultProguardFile("name: "proguard-android-optimize.txt"),
25                 "proguard-rules.pro"
26             )
27         }
28     }
29     compileOptions {
30         sourceCompatibility = JavaVersion.VERSION_11
31         targetCompatibility = JavaVersion.VERSION_11
32     }
33     kotlinOptions {
34         jvmTarget = "11"
35     }
36
37     buildFeatures {
38          viewBinding = true
39     }
40
41 }
42
43 dependencies {
44
45     implementation(libs.androidx.core.ktx)
46     implementation(libs.androidx.appcompat)
47     implementation(libs.material)
48     implementation(libs.androidx.activity)
49     implementation(libs.androidx.constraintlayout)
50     testImplementation(libs.junit)
51     androidTestImplementation(libs.androidx.junit)
52     androidTestImplementation(libs.androidx.espresso.core)
53 }

```

lo único que se ha añadido al gradle ha sido el viewBinding, que nos permite conectar de forma mas sencilla la parte lógica de la visual.