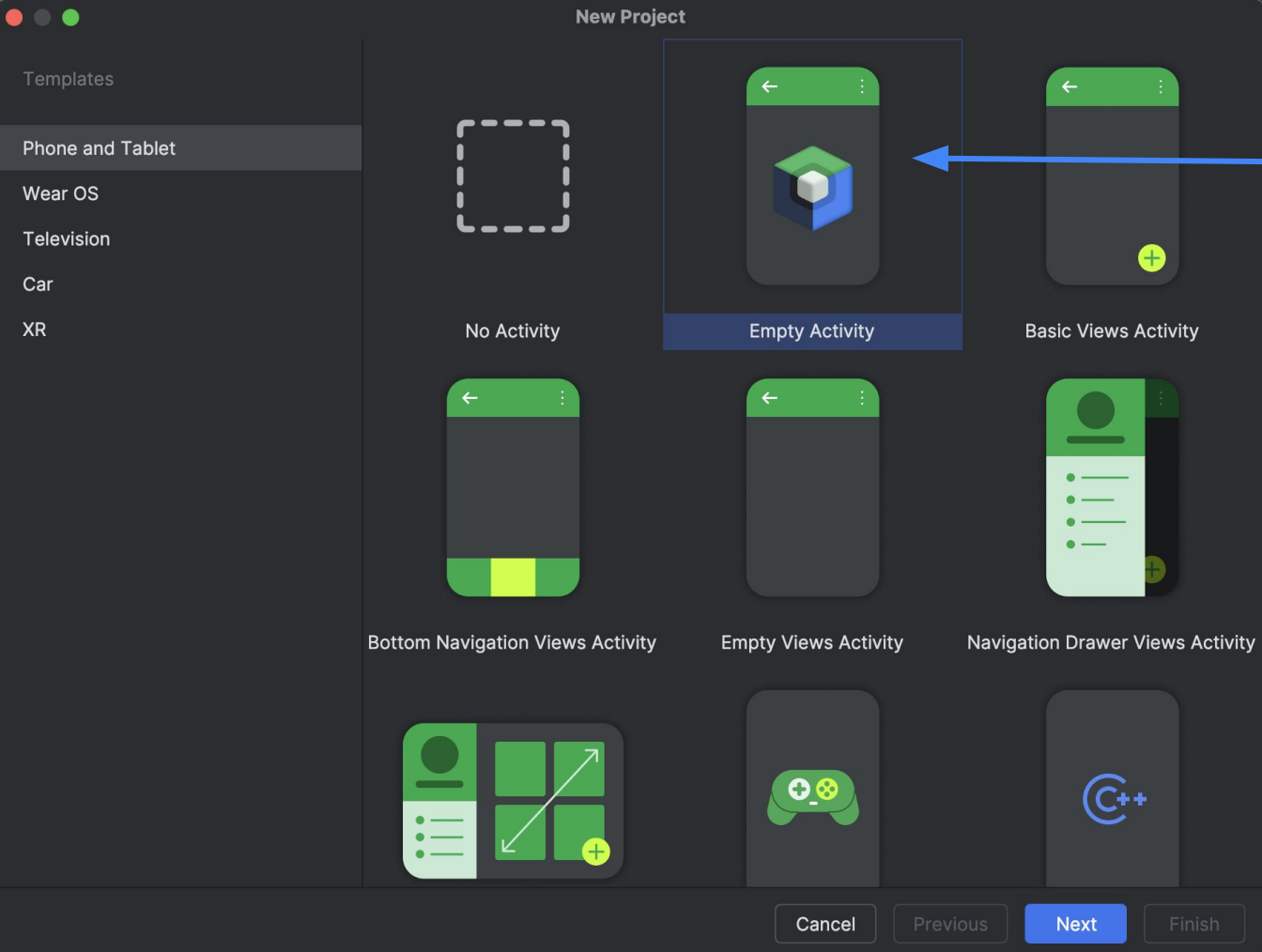




Razones para desarrollar apps con Kotlin



- Es simple y potente
- Totalmente integrado en Android Studio
- Dos grandes empresas apuestan por él.
 - JetBrains, Google
- Incluye conceptos y herramientas de programación modernos.
 - Lambdas, clases selladas, colecciones
- Buen tratamiento de los valores nulos.
- Muy presente en ofertas de trabajo.
- Multiplataforma
- Curva de aprendizaje más fácil que otros lenguajes
- Compaginable con Java
 - Incluso en el mismo proyecto



**JetPack
Compose**

GUI nativa de
Android

~~XML~~

Jetpack Compose

Características



- Declarativo

~~Imperativa~~

- Construido sobre Kotlin

- Menos código

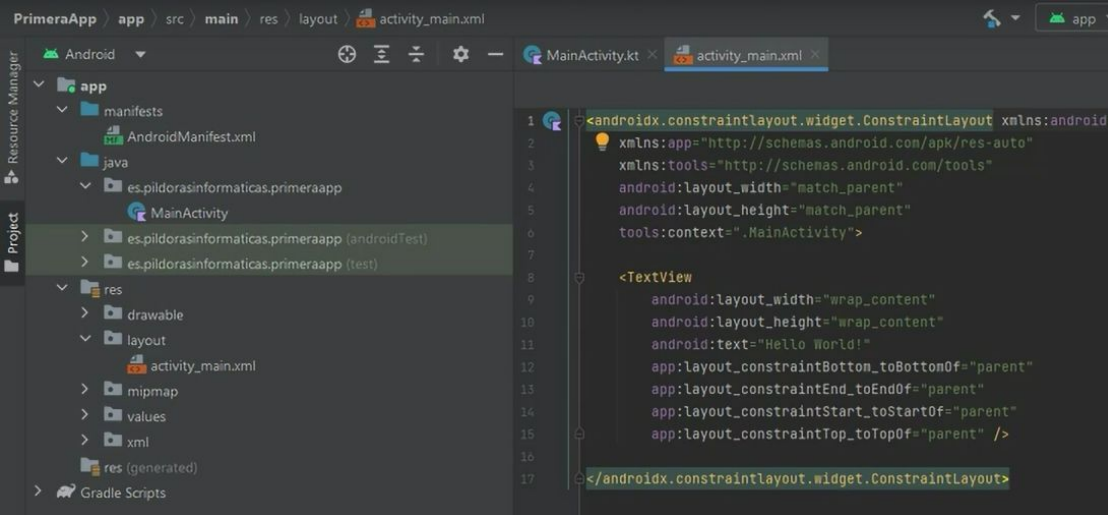
sólo un
archivo

- Interoperable

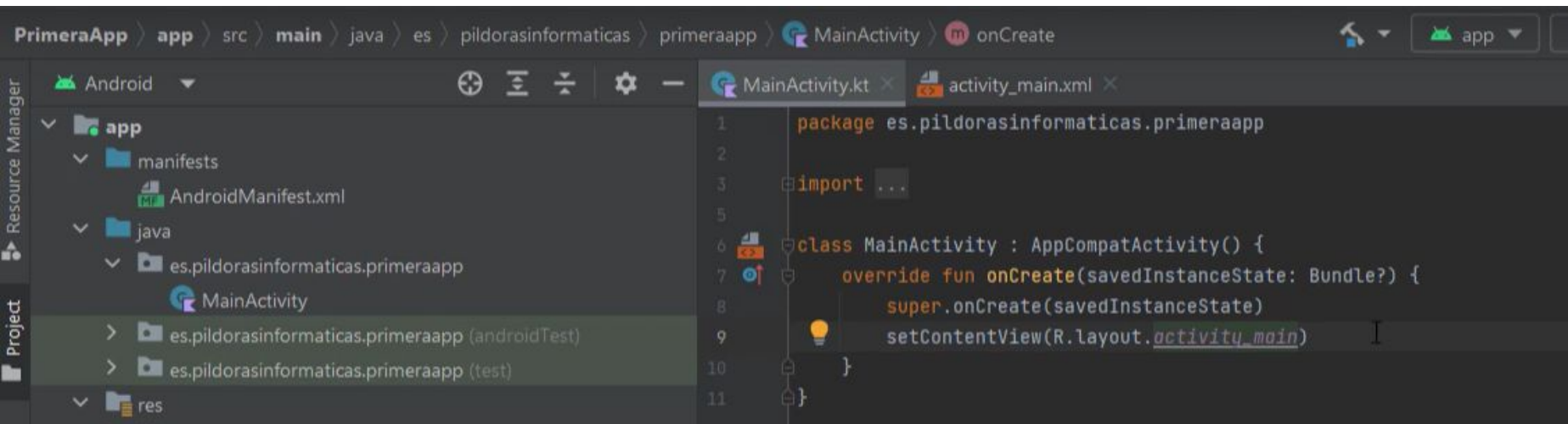
Puede mezclar XML

- Reutilizable





TRADICIONAL:
Kt + XML



EMPEZAMOS A PROGRAMAR

1. Insertar botón (forma gráfica) y dar formato y texto (“Pincha aquí”) e Id (MiBoton)
(Elimina el “Hello World”)
2. Insertar botón (forma lógica)

```
val button: Button=findViewById(R.id.MiBoton)
```

3. Orden para que escuche el evento click:

```
button.setOnClickListener{  
    Toast.makeText(...).show()  
}
```

<Button

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/MiBoton"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    android:text="Presiona aquí"/>
```

Gráfica

Lógica

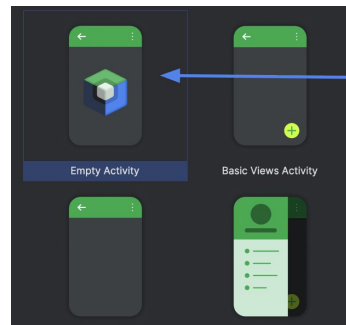
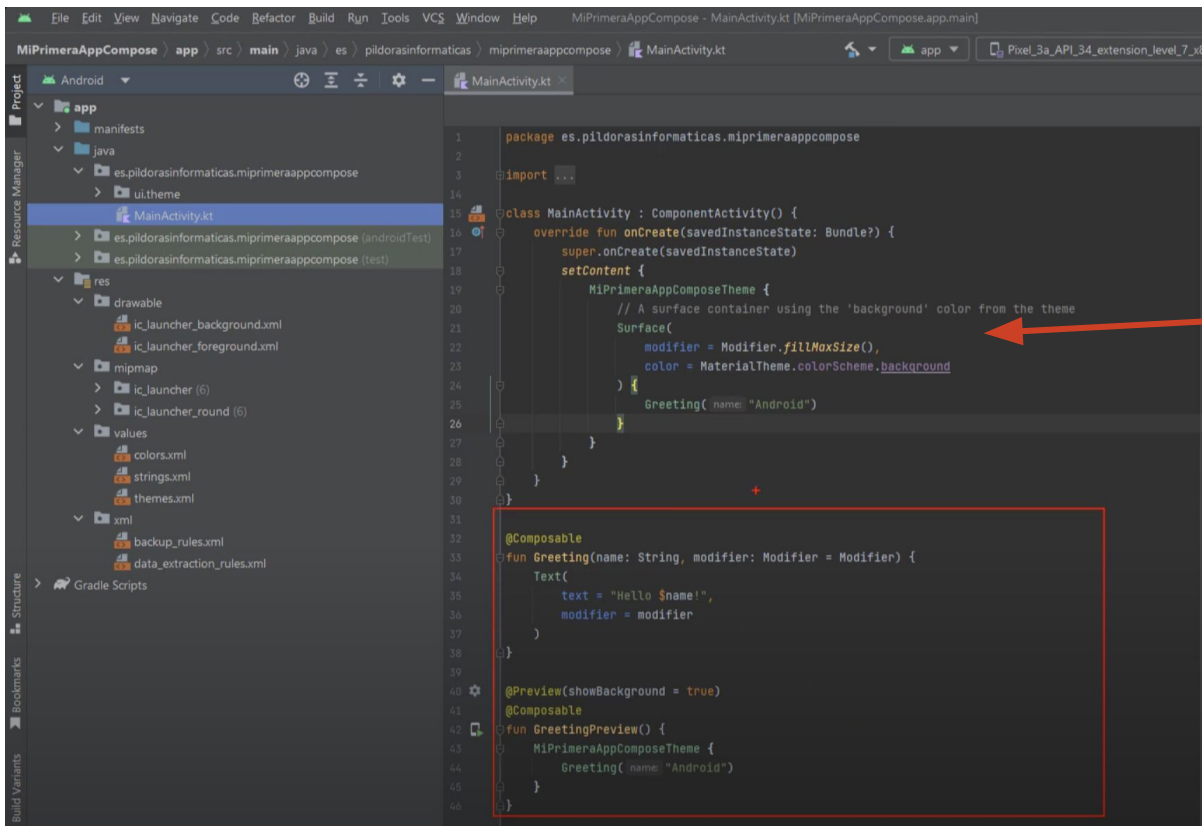
```
val button: Button = findViewById(R.id.MiBoton)
```

```
button.setOnClickListener { it: View!
```

```
    Toast.makeText(context: this, text: "Has presionado el botón", Toast.LENGTH_SHORT).show()
```

```
}
```

En Empty Activity



JetPack
Compose
GUI nativa de
Android

XML

También el
contenido del
setContent

Por el momento
borramos estos
métodos

En Empty Activity

Ahora todo estará en “MainActivity”

Haremos lo necesario para un botón como lo realizado con el ejemplo anterior.

```
class MainActivity : AppCompatActivity() {  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        enableEdgeToEdge()  
        setContent {  
  
            botonComponente()  
  
        }  
    }  
}
```

1 Usage

@Composable

```
fun botonComponente() {  
  
    Box(  
        modifier = Modifier.fillMaxSize(),  
        contentAlignment = Alignment.Center  
    ) {  
  
        val context = LocalContext.current  
        Button(onClick= {  
            Toast.makeText(context, text = "Has presionado el botón", duration = Toast.LENGTH_SHORT).show()  
        }) { Text(text = "Click aquí") }  
  
    }  
}
```

