Android – Funciones de extensión

OpenWebinars

1.

Utilidades

View

```
fun View.isVisible() : Boolean {
    return this.visibility == View.VISIBLE
}
```

ImageView y Picasso

```
fun ImageView.loadUrl(url: String) {
    Picasso.with(this.context).load(url).into( target: this)
}
```

"

TextView

```kotlin
fun TextView.setTextAndHideViewIfIsNeeded(text: String) {
    if (!TextUtils.isEmpty(text)) {
        this.text = text
        this.visibility = View.VISIBLE
    } else {
        this.text = ""
        this.visibility = View.GONE
    }
}
```

RecyclerView

```
fun RecyclerView.setUp(layoutManager: RecyclerView.LayoutManager,
                       adapter: RecyclerView.Adapter<RecyclerView.ViewHolder>) {
    this.layoutManager = layoutManager
    this.adapter = adapter
}
```

"

Activity

```kotlin
fun Activity.showToast(text: String, duration: Int = Toast.LENGTH_SHORT) {
    Toast.makeText( context: this, text, duration).show()
}


fun <T : View> Activity.bindView(@IdRes res: Int): Lazy<T> {
    return lazy(LazyThreadSafetyMode.NONE) { findViewById<T>(res) }
}

fun <T : View> Activity.bindViews(@IdRes resList: IntArray): List<Lazy<T>> {
    return resList.map { bindView<T>(it) }.toList()
}
```

2.

¿Dónde las declaramos?

> Crearemos un archivo con la palabra *Extension* al final.
> Ej: *ActivityExtension.kt*
> *ViewExtension.kt*

3.

¿Cómo las usamos?

"
El *import* necesario

```
import com.openwebinars.funcionesdeextension.extension.*

var titleLabel: Lazy<TextView> = bindView(R.id.main__title_label)


titleLabel.value.isVisible()

titleLabel.value.setTextAndHideViewIfIsNeeded("")

showToast( text: "Hola Mundo")
showToast( text: "Hola Mundo",   duration: 500)
```