

Android KTX

Könnyed Android fejlesztés Kotlinnal

Braun Márton • AutSoft • braun.marton@autsoft.hu

Kotlin és Android

- »» Bizonyos Android API-kat nehéz használni
- »» A Kotlinnal lehetőségünk van átformálni őket
 - »» Extension functions and properties
 - »» Named and default parameters
 - »» Operator overloading
 - »» Lambdas, higher order functions, and inline functions
- »» Mindenki készített saját könyvtárat...



Android KTX

- »»» A Google létrehozta a sajátját
 - »»» Első megjelenés 2018. február 2-án
- »»» Célok
 - »»» Csak kifejezetten Kotlinos API-k nyújtása
 - »»» Semmi új implementáció, csak jobb szintaxis
 - »»» Minimális költség, “zero overhead abstractions”



Példák

Visibility

»» Extension properties

```
view.setVisibility(View.VISIBLE)
```

```
view.visibility = View.VISIBLE
```

```
view.isVisible = true
```

Padding

»» Default and named parameters

```
view.setPadding(20, view.topPadding,  
                20, view.bottomPadding)
```

```
view.updatePadding(left = 20, right = 20)
```

ViewGroup

»» Operator overloading and extension functions

```
linearLayout.addView(view)
```

```
linearLayout += view
```

```
linearLayout.removeView(view)
```

```
linearLayout -= view
```

ViewGroup

»» Higher order, inline extension functions

```
for (i in 0 until linearLayout.childCount) {  
    val view = linearLayout.get(i)  
    Timber.d(view.tag)  
}
```

```
linearLayout.forEach { view ->  
    Timber.d(view.tag)  
}
```


Bitmap

»» Extension functions

```
val bitmap = Bitmap.createBitmap(  
    view.width, view.height, Bitmap.Config.ARGB_8888)  
view.draw(Canvas(bitmap))
```

```
val bitmap = view.toBitmap()
```

Bitmap

»» Operator overloading

```
val color = bitmap.getPixel(100, 50)
```

```
val color = bitmap[100, 50]
```

```
bitmap.setPixel(100, 50, color)
```

```
bitmap[100, 50] = color
```

Toast

»» Extension functions, default parameters

```
Toast.makeText(this, "Hello world", Toast.LENGTH_SHORT)  
    .show()
```

```
toast("Hello world")
```

Toast

»» Extension functions, default parameters

```
Toast.makeText(this, "Hello world", Toast.LENGTH_LONG)  
    .show()
```

```
toast("Hello world", duration = Toast.LENGTH_LONG)
```

SharedPreferences

»» Higher order, inline extension functions

```
SharedPreferences.edit()  
    .putBoolean("drawerOpened", true)  
    .apply()
```

```
SharedPreferences.edit {  
    putBoolean("drawerOpened", true)  
}
```

SQLiteDatabase

»» Higher order, inline extension functions

```
db.beginTransaction()  
db.insert("users", null, values)  
db.setTransactionSuccessful()  
db.endTransaction()
```

```
db.transaction {  
    insert("users", null, values)  
}
```

SQLiteDatabase

```
inline fun <T> SQLiteDatabase.transaction(...): T {  
    beginTransaction()  
    try {  
        val result = body()  
        setTransactionSuccessful()  
        return result  
    } finally {  
        endTransaction()  
    }  
}
```

Location

»» Destructuring

```
val lat = location.latitude
```

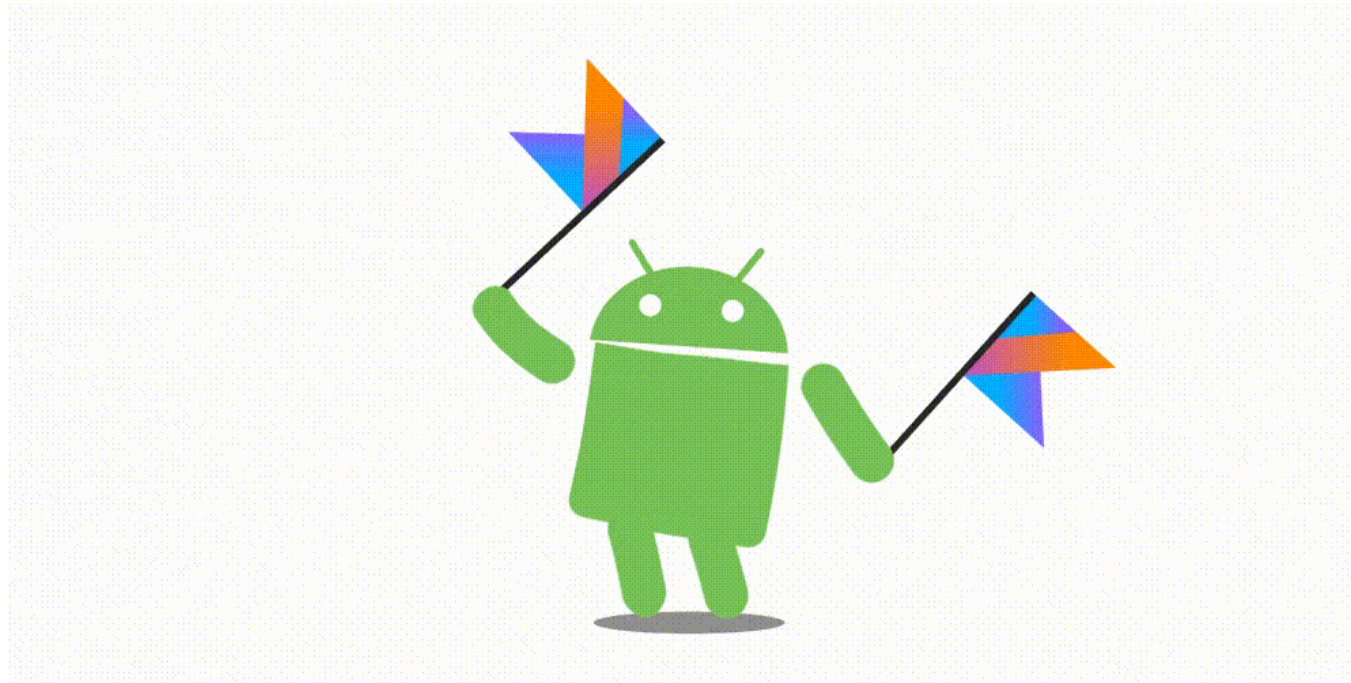
```
val lon = location.longitude
```

```
val (lat, lon) = location
```


Összefoglalás

- »»» Érdemes felfedezni és használni az Android KTX-et
 - »»» <https://github.com/android/android-ktx/>
 - »»» <https://android.github.io/android-ktx/core-ktx/>
- »»» Írjunk bátran saját extension-öket a rossz API-khoz
 - »»» Ha ez az Android keretrendszer, járuljunk hozzá a KTX-hez

Kérdések?



Braun Márton • AutSoft • braun.marton@autsoft.hu