Elegance of Kotlin

Anton Bannykh
JetBrains

goo.gl/RcE9iT

Kotlin

➔ Statically typed

➔ Pragmatic

➔ Concise

➔ Safe

➔ Interoperable

➔ Tool-friendly

https://kotlinlang.org/

Example

- Android
- Text on the screen
- Contains an integer
- Task: increment it

```
public class TextView … {

    public CharSequence getText() {...}

    public final void setText(CharSequence text) {...}

}
```

Java

```java
public void increment(@NonNull View view) {
    TextView incView = (TextView) findViewById(R.id.textView);
    int value = Integer.parseInt(incView.getText().toString()) + 1;
    incView.setText(Integer.toString(value));
}
```

Kotlin

```kotlin
fun increment(view: View) {
    val incView = findViewById(R.id.textView) as TextView
    val value = Integer.parseInt(incView.text.toString()) + 1
    incView.text = Integer.toString(value)
}
```

Java

```
public void increment(@NonNull View view) {
    TextView incView = (TextView) findViewById(R.id.textView);
    int value = Integer.parseInt(incView.getText().toString()) + 1;
    incView.setText(Integer.toString(value));
}
```

Kotlin

```
import kotlinx.android.synthetic.main.activity_main.*
...
fun increment(view: View) {
    val incView = textView
    val value = Integer.parseInt(incView.text.toString()) + 1
    incView.text = Integer.toString(value)
}
```

Java

```java
public void increment(@NonNull View view) {
    TextView incView = (TextView) findViewById(R.id.textView);
    int value = Integer.parseInt(incView.getText().toString()) + 1;
    incView.setText(Integer.toString(value));
}
```

Kotlin

```kotlin
fun increment(view: View) {
    val incView = textView
    val value = Integer.parseInt(incView.text.toString()) + 1
    incView.text = Integer.toString(value)
}
```

Java

```java
public void increment(@NonNull View view) {
    TextView incView = (TextView) findViewById(R.id.textView);
    int value = Integer.parseInt(incView.getText().toString()) + 1;
    incView.setText(Integer.toString(value));
}
```

Kotlin

```kotlin
fun increment(view: View) {
    val value = Integer.parseInt(textView.text.toString()) + 1
    textView.text = Integer.toString(value)
}
```

Java

```java
public void increment(@NonNull View view) {
    TextView incView = (TextView) findViewById(R.id.textView);
    int value = Integer.parseInt(incView.getText().toString()) + 1;
    incView.setText(Integer.toString(value));
}
```

Kotlin

```kotlin
fun increment(view: View) {
    intView++
}

var intView: Int
    get() = textView.text.toString().toInt()
    set(v) {
        textView.text = v.toString()
    }
```

Java

```java
public void increment(@NonNull View view) {
    TextView incView = (TextView) findViewById(R.id.textView);
    int value = Integer.parseInt(incView.getText().toString()) + 1;
    incView.setText(Integer.toString(value));
}
```

Kotlin

```kotlin
fun increment(view: View) {
    textView.intText++
}

var TextView.intText: Int
    get() = text.toString().toInt()
    set(v) {
        text = v.toString()
    }
```

Java

```
public void increment(@NonNull View view) {
    TextView incView = (TextView) findViewById(R.id.textView);
    int value = Integer.parseInt(incView.getText().toString()) + 1;
    incView.setText(Integer.toString(value));
}
```

Kotlin

```
fun increment(view: View) {
    textView.intText++
}
```

Java

```java
public void increment(@NonNull View view) {
    TextView incView = (TextView) findViewById(R.id.textView);
    int value = Integer.parseInt(incView.getText().toString()) + 1;
    incView.setText(Integer.toString(value));
}
```

Kotlin

```kotlin
fun increment(view: View) = textView.intText++
```

Java

```java
public void increment(@NonNull View view) {
    TextView incView = (TextView) findViewById(R.id.textView);
    int value = Integer.parseInt(incView.getText().toString()) + 1;
    incView.setText(Integer.toString(value));
}
```

Kotlin

```kotlin
fun increment(view: View) {
    val incView = findViewById(R.id.textView)
    if (incView is TextView) {
        val value = Integer.parseInt(incView.text.toString()) + 1
        incView.text = Integer.toString(value)
    } else throw Error()
}
```

Java

```java
public void increment(@NonNull View view) {
   TextView incView = (TextView) findViewById(R.id.textView);
   int value = Integer.parseInt(incView.getText().toString()) + 1;
   incView.setText(Integer.toString(value));
}
```

Kotlin

```kotlin
fun increment(view: View) {
   val incView = findViewById(R.id.textView) as? TextView
           ?: throw Error()
   val value = Integer.parseInt(incView.text.toString()) + 1
   incView.text = Integer.toString(value)
}
```

Advanced

Delegate

```
var intView by intViewDelegate(textView)
```

Standard library

```
data class Person(val name: String, val age: Int)

val people = listOf(Person("John", 30))

for ((name, age) in people) {
   println("$name is $age years old")
}

val adults = people.filter { it.age > 21 }
```

Advanced

DSL

```
verticalLayout {
    val name = editText()
    button("Say Hello") {
        onClick { toast("Hello, ${name.text}!") }
    }
}
```

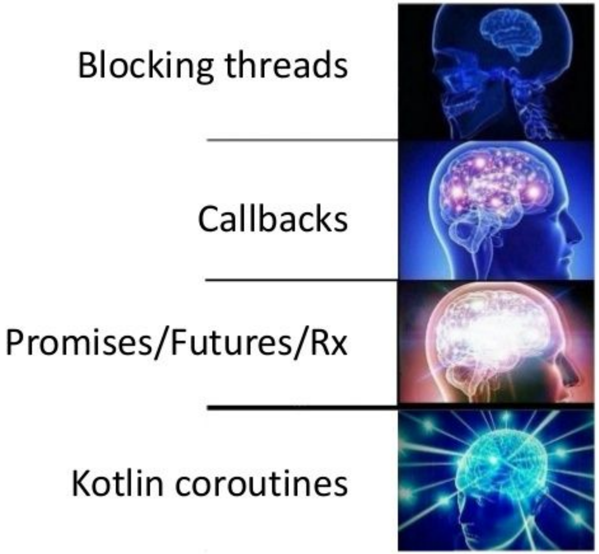
https://github.com/Kotlin/anko

Bleeding edge

Coroutines

```
async(UI) {
    val data: Deferred<Data> = bg {
     // Runs in background
     getData()
    }

    // This code is executed on the UI thread
    showData(data.await())
}
```

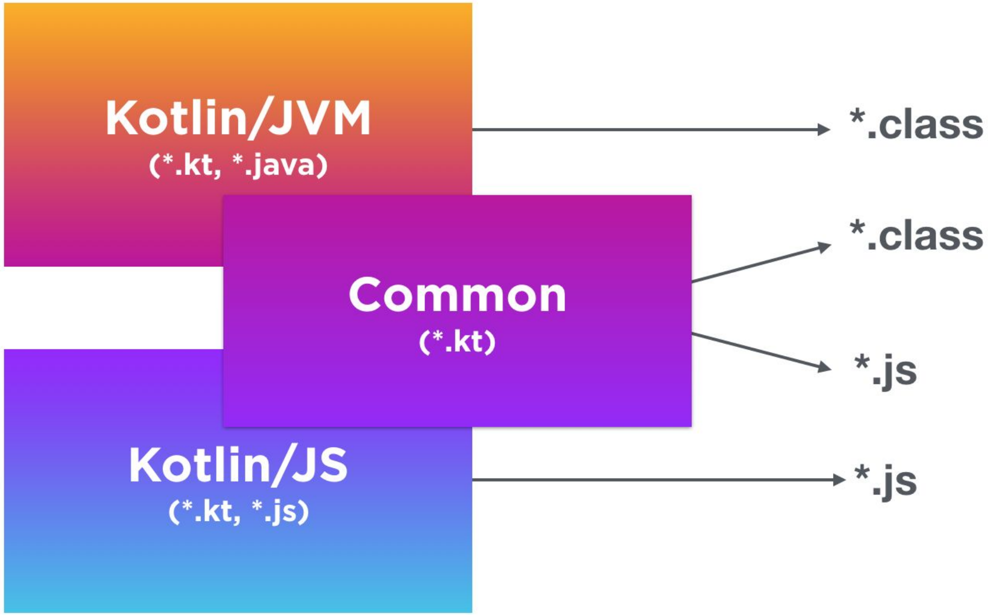


https://kotlinlang.org/docs/reference/coroutines.html

Bleeding edge

Multiplatform

- JVM
- JS
  - EcmaScript 5
- Native
  - Tech preview
  - Based on LLVM
  - Cool demos!



Links

kotlinlang.org - documentation and marketing

try.kotl.in - try online, complete koans

github.com/Kotlin/anko - goodies for Android development

github.com/Kotlin/kotlinx.html - isomorphic HTML DSL

github.com/Kotlin/kotlinx.coroutines - official coroutine support library