# Chapter4

Kotlin和java差別：

1. 回調函數定義：

Kotlin:

Kotlin 版本使用 lambda 表達式定義 ActivityResultLauncher 的回傳函數: { result: ActivityResult ->

private val startForResult = registerForActivityResult(  
 ActivityResultContracts.StartActivityForResult()  
) **{** result: ActivityResult **->**

java:

Java 版本使用匿名內建類別 ActivityResultCallback，實現ActivityResultCallback<ActivityResult>

private ActivityResultLauncher<Intent> startForResult = registerForActivityResult(  
new ActivityResultContracts.StartActivityForResult(),  
new ActivityResultCallback<ActivityResult>() {  
 @Override  
 public void onActivityResult(ActivityResult result) **{**

**Kotlin 的 lambda 表達更加簡潔、易懂**

1. 空值處理

Kotlin:

Kotlin 版本使用 ?. 來安全地獲取 Intent 中的數據:

val drink = intent?.getStringExtra("drink")  
val sugar = intent?.getStringExtra("sugar")  
val ice = intent?.getStringExtra("ice")

java:

Java 版本需要檢查 Intent 是否為 null

if (intent != null) { *intent*.getStringExtra("drink") }

**Kotlin 安全機制可以更好地避免 NullPointerException**

1. 字串拼接

Kotlin:

使用$符號

"飲料：$drink\n\n甜度：$sugar\n\n冰塊：$ice"

Java:

"飲料: " + drink + "\n\n甜度: " + sugar + "\n\n冰塊: " + ice

1. 類型推斷

Kotlin:

Kotlin 版本可以自動推斷 startForResult 變量的類型為 ActivityResultLauncher<Intent>

Java:

Java 版本需要聲明類型: ActivityResultLauncher<Intent> startForResult =……