

Kotlin 讀書會 S3E6

補充資料

請關注第三梯次讀書會活動：<https://tw.kotlin.tips/study-jams/3>

共讀書本及討論群

- 導讀投影片：<https://www.slideshare.net/ssuser080ac4/2021-kotlin-iinterface>
- 導讀錄影：https://youtu.be/hy5F_QifQNE
- 共讀書本：<https://www.tenlong.com.tw/products/9789865021870>
- 讀書會討論群
 - Line 群 https://line.me/R/ti/g/jAG_c1DMaO (請先加聖佑 line id: shengyoufan 後傳訊息給聖佑, 聖佑再邀請入群)
 - Telegram 群 https://t.me/joinchat/DAUjOBfu7_05sa9YYb5FQ

導讀心智圖



抽象類別 Abstract Class

- 無法明確被界定的東西, 抽象模糊的概念
- abstract Class使用方法:
 - 在class前加上abstract, 不用寫open
 - 屬性跟函式加上abstract就要被繼承的class覆寫

介面 Interface

- Class 只能繼承一個, 但 Interface 可以多個
- 使用方式:
直接interface 加名稱 ->不用像class需要加建構式
- 可以只複寫某個介面

```
abstract class Y(){  
}  
class X : Y(), A, B {  
    override fun myFunction() {  
        super<A>.myFunction()  
        // super<B>.myFunction()  
    }  
}
```

型別檢查與轉型

is 運算子

- is 運算子回傳 boolean
- is 檢查過的 kotlin 會幫我們做智慧轉型
- 確認物件型態是否符合預期的型態

```
fun demo(x: Any) {  
    if (x is String) {  
        print(x.length) // x is automatically cast to String  
    }  
}
```

as 明確轉型

- 強制轉型

- 因為編譯器無法確定型態，但若我們寫Code是確定某個型態，可以用as強制將物件轉型使用

```
val x: String = y as String
```

Q&A:

interface 跟 class 很像 只是interface 可以繼承很多而已 嗎？

- <https://chinchilla-shuan.medium.com/%E6%88%91%E5%9C%A8-kotlin%E8%AE%80%E6%9B%B8%E6%9C%83%E7%9A%84%E6%97%A5%E5%AD%90-%E4%B8%89-%E5%B0%8E%E8%AE%80-interface-and-abstract-class-2c6192fdd711>
- 參考課本175頁的說明
- http://andrewchen.tw/2017/04/09/20170409_NOTE_%E8%A8%AD%E8%A8%88%E6%A8%A1%E5%BC%8F%E4%BA%94%E5%A4%A7%E5%9F%BA%E6%9C%AC%E5%8E%9F%E5%89%87SOLID/

interface的屬性跟 method是不是只能是open的？

- 通常interface被定義出來就是要拿來被繼承實作使用，當宣告為private就沒人可以使用了。所以在interface寫了private method就要把內文寫出來。

好奇kotlin哪些東西是新的Java沒辦法取代的？

- Null safety
- Unchecked exception
- Function Programming