

1. Encapsulation(封裝) · 物件導向原則之一，將功能與資料裝成一個物件，可

以透過內部的標籤來決定外界可存取的資料，像是: `private`、`package`、

`protected`、`public`

優點：

a. 相同功能，程式可以重複使用，變成模組的概念

b. 降低撰寫錯誤產生

c. 內部可以隱藏資料，不被外界更改

2. Enum(列舉) · 利用關鍵字來做宣告，預設為 `int`，可宣告 `byte`、`sbyte`、`short`、

`ushort`、`int`、`uint`、`long`、`ulong`，可以讓某些程式碼讀起來更輕鬆，像是

遊戲的上下左右判斷

3.

```
enum class directions {  
    up,  
    down,  
    left,  
    right  
}  
  
fun main(args: Array<String>) {  
  
    println("name=" + directions.up.name)  
    println("name=" + directions.down.name)  
    println("name=" + directions.left.name)  
}
```

```
println("name=" + directions.right.name)
}
```

result:

name=up
name=down
name=left
name=right

4. **LinearLayout**:線性方向的排版，有分垂直(Horizontal)跟水平(vertical)

RelativeLayout:相對性的排版，利用圖跟圖或字之間的將對位置去做排版

5. **ConatraintLayout**:

- a.可以快速調整兩個物件之間的間隔
- b.可以設定物件的長寬比
- c.可以同時展現 **RelativeLayout** 跟 **LinearLayout** 的屬性
- d.快速對齊物件
- e.螢幕變換長寬時，會自動進行調整

->我們可以更有效率地將介面排版出來