1. What is Encapsulation? Try to list out the benefits of Encapsulation.

封裝是物件導向的特性之一，是指將類別的成員(屬性、方法等)定義在同個類別之內，能夠限制該類別的成員被外部存取，透過修飾字(public、private..)可以控制類別成員能被存取的範圍，封裝的好處有：1.隱藏內部的細節、降低複雜度；2.增加安全性，保護內部程式不被任意修改3.方便維護及擴展

1. Try to explain what is enum and why we use them.

列舉是種特殊的類別，可以將程式中常用的常數定義在一個列舉類別內統一管理，方便程式取用，透過將常用的常數定義在列舉內，可以方便後續的維護

1. How to use enum in Kotlin? Attach a sample code that use enum in Kotlin.

enum class Card(val num : Int){  
 spade(1),  
 heart(2),  
 diamond(3),  
 club(4)  
}  
  
fun main(args: Array<String>) {  
 var card1 = Card.spade  
  
 *println*(card1)  
 *println*(card1.num)  
}

1. What are the differences between LinearLayout and RelativeLayout? Try to explain in details.

以LinearLayout設計的layout，內部的元素只能設定為水平或垂直排列，所以每一行或一列只會有一個元素，元素間可以保持固定的間隔，畫面相對整齊，適合用來設計較簡單的畫面；RelativeLayout的設計方式為元素間的相對位置或對父元素的位置做設計，你可以在同一行或列上放上多個元素，可以用來設計較複雜的畫面。

1. Try to explain the benefits of ConstraintLayout. Why you should use it?

ConstraintLayout常用Design模式做設計，透過元素間的約束，能夠使各種大小的裝置呈現出的畫面位置比例看起來較一致，這點是RelativeLayout相對來說較能做到的優勢。