

《Android 架构设计与模式高端培训》

■ 培训特点

- **教学方法独特** 祝成科技软件开发培训组有一组独特的教学方法即深入的原理剖析+实际的企业应用案例，剖析软件平台架构、分析软件开发技术。根据以前授课经验，这种教学方法得到了企业开发人员的热烈响应。
- **内容编排科学、实用** 课程是讲师多年企业应用开发、设计理论与实践经验的总结与提炼、并针对企业客户的特殊要求进行专门设计，非常适合有经验的企业学员。
- **分析透彻、深入** 该课程不是泛泛介绍与堆砌各种开发与设计技术，而是建立在讲师对企业级软件开发技术所具有的特点、分析与设计的深刻理解之基础上。根据以前的授课经验，学员普遍反映效果非常好。
- **注重案例，注重架构** 学习企业级软件开发技术的关键在“应用”，本培训通过丰富的案例教学，剖析常见的企业开发应用模式，帮助学员更好地掌握进行企业级应用程序开发时所涉及到的设计方法、模式与架构。
- **专家名师讲授** 祝成科技软件开发培训中心拥有一支高素质的软件开发培训专家队伍，尤其在 C++、.NET、Java、Android 等领域，祝成科技有着全国第一流的师资，在中国软件开发培训界享有极高的盛誉，并经常被微软 Tech·ED、Intel 公司、中国软件开发大会邀请进行技术演讲。中心保证派出去的全部是全国知名的讲师。

■ 典型客户



■ 培訓綱要

Part 1 設計樣式與 Android 框架設計

- 1.1 設計與樣式(即招式)
- 1.2 簡介設計樣式(Design Pattern)
- 1.3 從亞歷山大的建築樣式到 GoF 設計樣式
- 1.4 舉例說明應用框架之意義
- 1.5 舉例說明框架設計基本思維：「變與不變之分離」
- 1.6 欣賞 Android 裡的 13 項設計樣式

Part 2 Android 框架與應用類別的接合：卡樺(Hook)函數

- 2.1 何謂卡樺(Hook)函數
- 2.2 Android 的卡樺函數範例(一)
- 2.3 Android 的卡樺函數範例(二)
- 2.4 Android 的卡樺函數範例(三)
- 2.5 卡樺函數與樣式之關係
---- 以 *Factory Method* 和 *Template Method* 樣式為例

Part 3 Template Method 樣式

- 3.1 複習：「變與不變之分離」原則
- 3.2 複習：「變與不變之分離」手藝
- 3.3 複習：框架的反向控制
- 3.4 介紹 Template Method 樣式
- 3.5 Android 框架與 Template Method 樣式

Part 4 Factory Method 樣式

- 4.1 Factory Method 樣式美何在？
- 4.2 介紹 Factory Method 樣式
 - 4.2.1 誰來誕生應用類別之物件？
 - 4.2.2 GoF 的 Factory Method 樣式圖
 - 4.2.3 Factory Method 樣式的延伸
- 4.3 Android 框架與 Factory Method 樣式
 - 4.3.1 Factory Method 樣式範例之一
 - 4.3.2 Factory Method 樣式範例之二

Part 5 Observer 樣式

- 5.1 Observer 樣式美何在？
 - 5.1.1 互換性之意義
 - 5.1.2 提升互換性之途徑：降低相依性
 - 5.1.3 Observer 樣式之美：締造互換性
- 5.2 介紹 Observer 樣式



- 5.2.1 委託式的反向控制 (IoC)
- 5.2.2 GoF 的 Observer 樣式圖
- 5.2.3 Observer 樣式的延伸
- 5.3 Android 框架與 Observer 樣式

Part 6 Abstract Factory 樣式

- 6.1 Abstract Factory 樣式美何在?
 - 6.1.1 大格局的分合自如
 - 6.1.2 「不知而亦能用」之實踐
- 6.2 介紹 Abstract Factory 樣式
- 6.3 Android 框架與 Abstract Factory 樣式
 - 6.3.1 Abstract Factory 樣式範例
 - 6.3.2 Abstract Factory 與 Observer 樣式之組合

Part 7 Adapter 樣式

- 7.1 Adapter 樣式美何在?
 - 7.1.1 介面是王道
 - 7.1.2 精緻化 Use 介面
- 7.2 介紹 Adapter 樣式
 - 7.2.1 Adapter 像家裡的電源轉接器
 - 7.2.2 GoF 的 Adapter 樣式圖
 - 7.2.3 Adapter 樣式的延伸
- 7.3 Android 框架與 Adapter 樣式
 - 7.3.1 Adapter 樣式範例之一
 - 7.3.2 Adapter 樣式範例之二

Part 8 Composite 樣式

- 8.1 Composite 樣式美何在?
 - 8.1.1 亂中有序
 - 8.1.2 序中有亂
 - 8.1.3 簡單組合、無限成長
- 8.2 介紹 Composite 樣式
 - 8.2.1 包含關係之抽象
 - 8.2.2 GoF 的 Composite 樣式圖
 - 8.2.3 Composite 樣式之範例：販賣機
- 8.3 Android 框架與 Composite 樣式

Part 9 Strategy 樣式

- 9.1 Strategy 樣式美何在?
 - 9.1.1 序中的繁雜
 - 9.1.2 一致的序、不同組合策略
- 9.2 介紹 Strategy 樣式
 - 9.2.1 不同策略間之變與不變

9.2.2 GoF 的 Strategy 樣式圖

9.3 Android 框架與 Strategy 樣式

Part 10 State 樣式

- 10.1 State 樣式美何在?
 - 10.1.1 狀態變化之美
 - 10.1.2 以 Android 的畫面佈局為例
- 10.2 介紹 State 樣式
 - 10.2.1 GoF 的 State 樣式圖
 - 10.2.2 State 樣式之範例
- 10.3 Android 框架與 State 樣式

Part 11 Proxy 樣式

- 11.1 Proxy 樣式美何在?
 - 11.1.1 有朋自遠方來
 - 11.1.2 以 Android 的 MediaPlayer 類別為例
- 11.2 介紹 Proxy 樣式
 - 11.2.1 從 Adapter 樣式說起
 - 11.2.2 GoF 的 Proxy 樣式圖
- 11.3 Android 框架與 Proxy 樣式
 - 11.3.1 Proxy 樣式範例之一
 - 11.3.2 Proxy 樣式範例之二
 - 11.3.3 Proxy 樣式範例之三
 - 11.3.4 Proxy 樣式範例之四

Part 12 Bridge 樣式

- 12.1 Bridge 樣式美何在?
 - 12.1.1 實作(Implementation)變化之美
 - 12.1.2 變化與對稱之美
- 12.2 介紹 Bridge 樣式
 - 12.2.1 複習「委託」機制
 - 12.2.2 複習「變與不變分離」
 - 12.2.3 GoF 的 Bridge 樣式圖
- 12.3 Android 框架與 Bridge 樣式
 - 12.3.1 Bridge 樣式範例之一
 - 12.3.2 Bridge 樣式範例之二

Part 13 Iterator 樣式

- 13.1 Iterator 樣式美何在?
 - 13.1.1 從 Program Counter 說起
 - 13.1.2 以 Java 的 ArrayList<Type> 集合類別為例
- 13.2 介紹 Iterator 樣式
 - 13.2.1 GoF 的 Iterator 樣式圖
 - 13.2.2 Iterator 樣式之延伸



- 13.3 Android 框架與 Iterator 樣式
 - 13.3.1 Iterator 樣式範例之一
 - 13.3.2 Iterator 樣式範例之二

Part 14 Mediator 樣式

- 14.1 Mediator 樣式美何在?
 - 14.1.1 從主機板(Motherboard)說起
 - 14.1.2 整合核心與決策核心
 - 14.1.3 多介面整合之美
- 14.2 介紹 Mediator 樣式
 - 14.2.1 從 Adapter 說起
 - 14.2.2 GoF 的 Mediator 樣式圖
- 14.3 Android 框架與 Mediator 樣式
 - 14.3.1 Mediator 樣式範例之一
 - 14.3.2 Mediator 樣式範例之二

Part 15 Façade 樣式

- 15.1 Façade 樣式美何在?
 - 15.1.1 複習「貨櫃」之比喻
 - 15.1.2 Façade 軟體貨櫃
- 15.2 介紹 Façade 樣式
 - 15.2.1 從 Adapter 說起
 - 15.2.2 GoF 的 Façade 樣式圖
- 15.3 Android 框架與 Façade 樣式
 - 15.3.1 Façade 樣式範例之一
 - 15.3.2 Façade 樣式範例之二