

# 客製化LISTVIEW

2017/05/31

# 範例下載處

- <https://github.com/silencecork/SHUWorkshop2017>
  - 可以使用Download Zip下載
  - 建議：使用git抓取  
git clone  
https://github.com/silencecork/SHUWorkshop2017.git

對應專案

ListView/listview-01

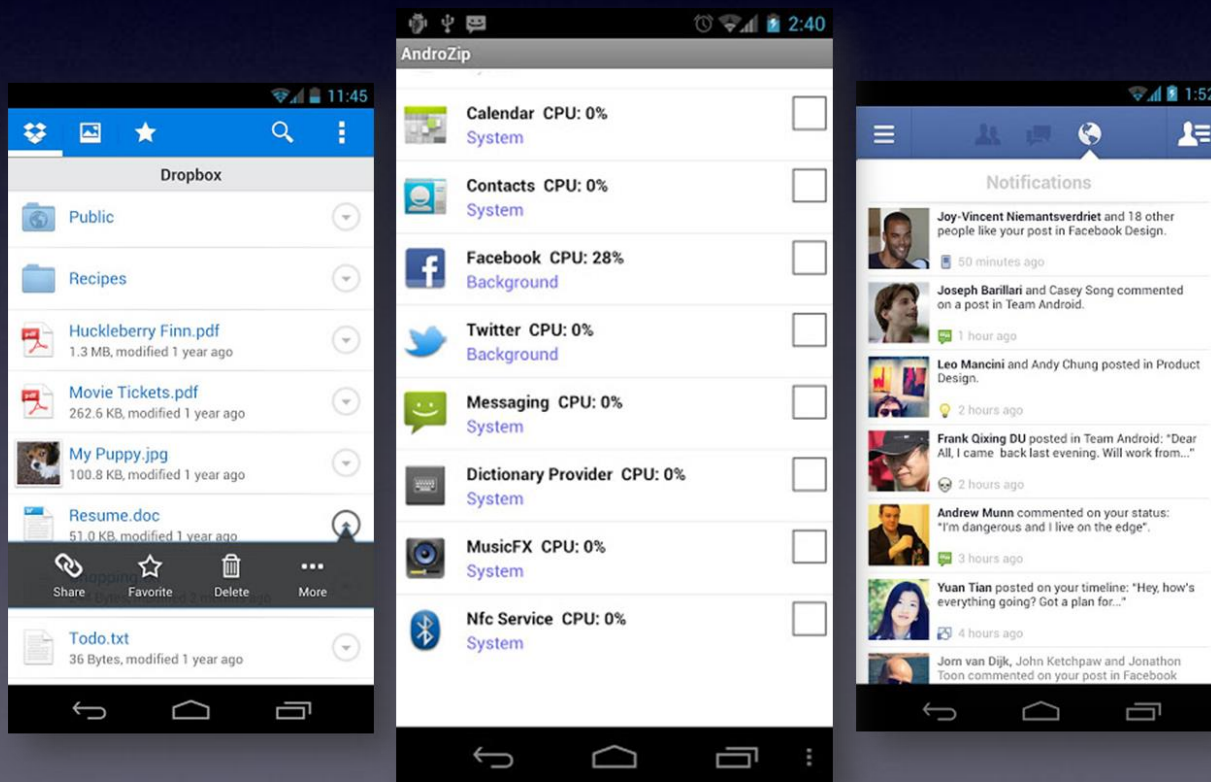
ListView/listview-02

客製化ListView

# 簡單的客製化LISTVIEW

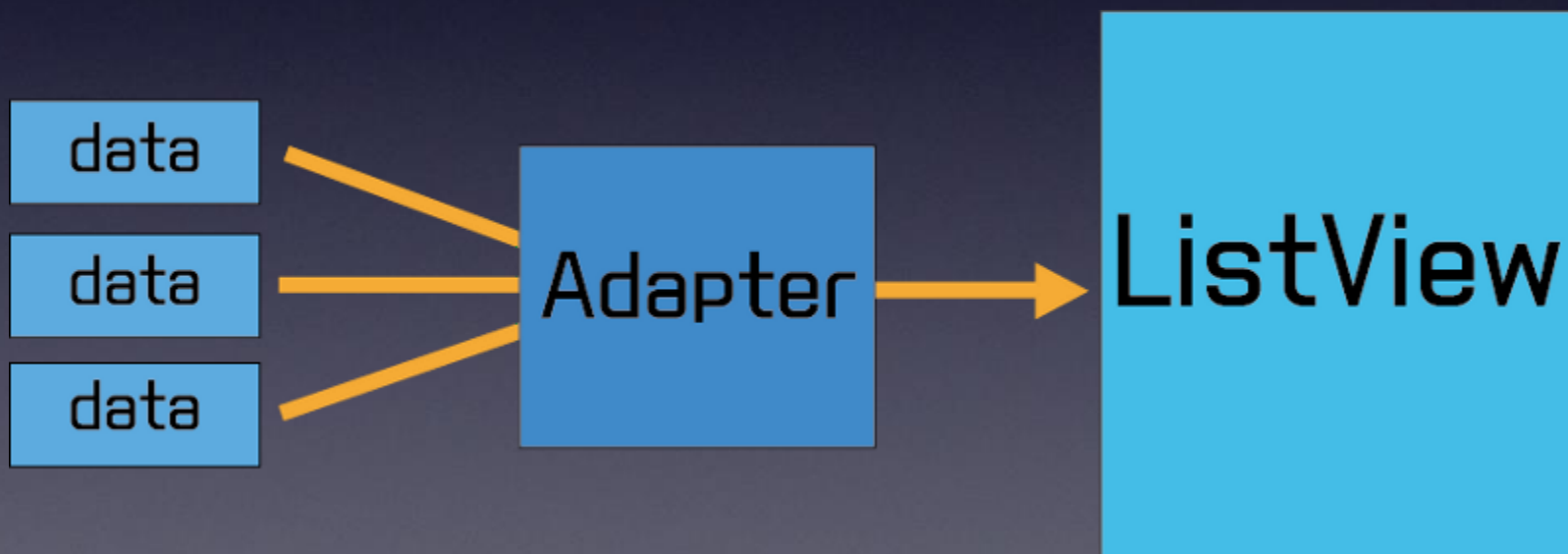
# 客製化LISTVIEW

- ListView可以依照Adapter的組成而產生出不同的各種樣式的列表



# 客製化LISTVIEW

- 資料是要呈現的內容
- ListView負責呈現結果的容器
- 資料怎麼擺放在UI上，則是Adapter所負責



# ADAPTER重點提示

- 客製化ListView使用的Adapter為BaseAdapter

```
import android.widget.BaseAdapter;  
  
public class WordAdapter extends BaseAdapter {  
    .....  
}
```

# ADAPTER重點提示

- BaseAdapter有三大基本function
  - getCount() – 共有多少筆資料
  - getItem() – 依照位置取得資料
  - getView() - 資料和要呈現的畫面合併處

# ADAPTER重點提示

```
public class WordAdapter extends BaseAdapter {  
    @Override  
    public int getCount() {  
        return 0;  
    }  
  
    @Override  
    public Object getItem(int position) {  
        return null;  
    }  
  
    @Override  
    public long getItemId(int position) {  
        return 0;  
    }  
  
    @Override  
    public View getView(int position, View convertView, ViewGroup  
parent) {  
        return null;  
    }  
}
```



# ADAPTER重點提示

- getView()
  - convertView - ListView的重複使用機制
  - 讀取Layout的方式 – LayoutInflater

```
@Override
public View getView(int position, View convertView, ViewGroup parent)
{
    if (convertView == null) {
        convertView =
LayoutInflater.from(parent.getContext()).inflate(R.layout.list_item,
null);
    }
    .....
    return convertView;
}
```

# ADAPTER重點提示

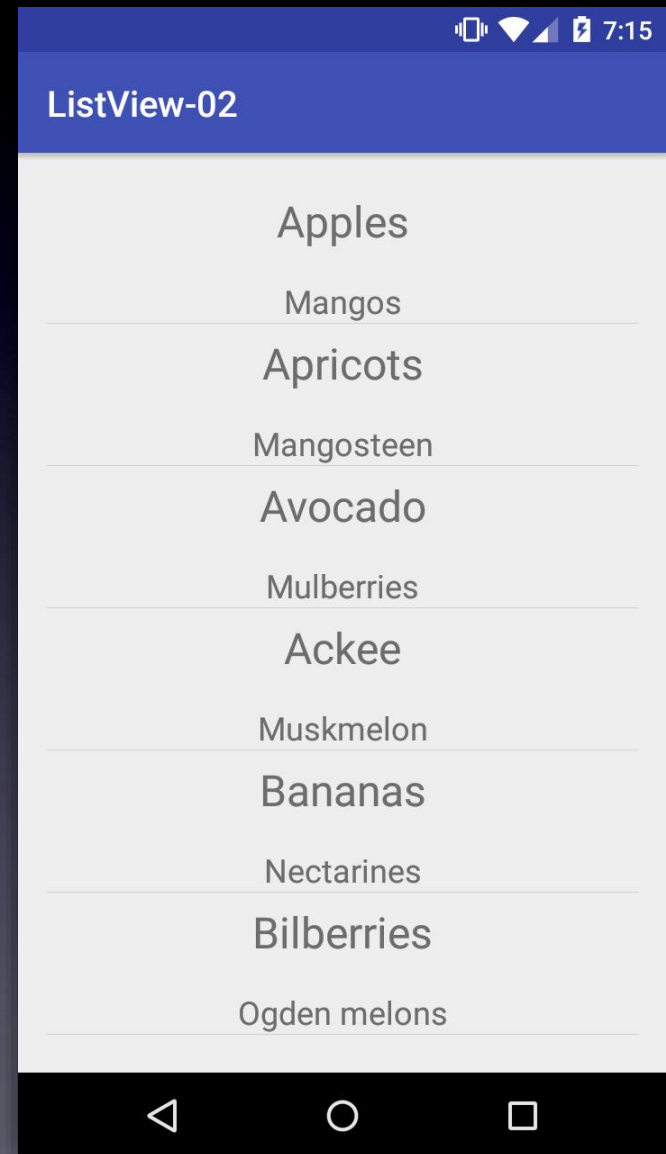
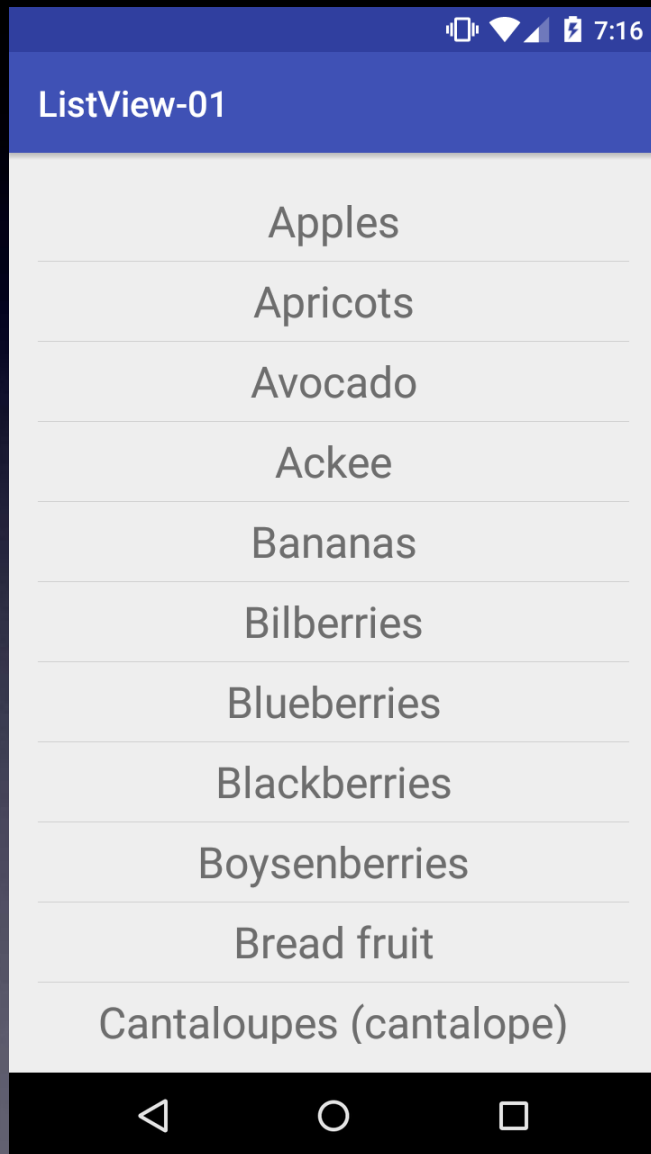
- getView()
  - ListView中的橫列，需要另外建立Layout來負責呈現

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/title"
        android:layout_width="match_parent"
        android:layout_height="44dp"
        android:gravity="center"
        android:textSize="24sp"/>

</RelativeLayout>
```

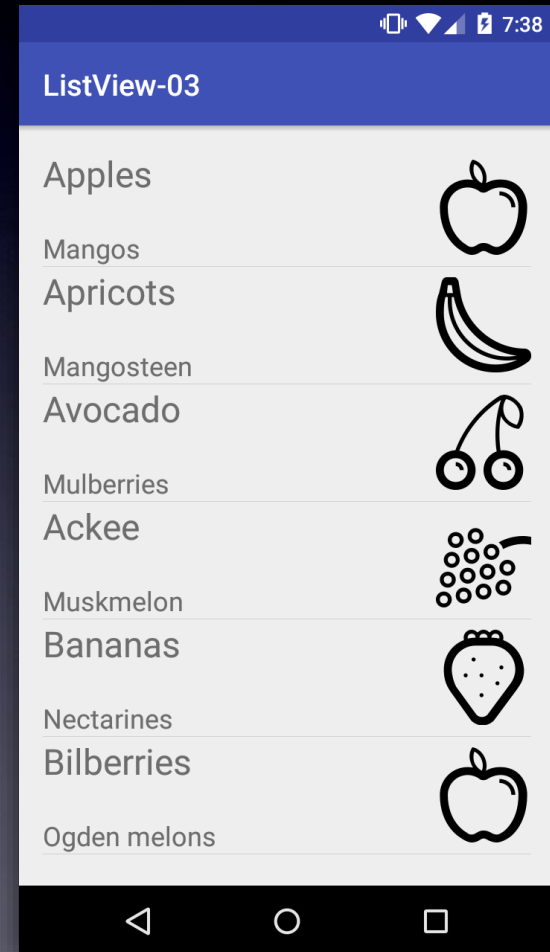
# 範例程式碼說明



# 練習

- 若要更多呈現如右，  
你要如何修改？
- 答案在branch  
ListView/listview-03-answer

對應專案  
ListView/listview-03



客製化ListView

物件化資料

# 物件化資料

- 目前問題？
  - 針對新的項目，Adapter就必須接受多一個資料來源
  - 擴充性差
  - 容錯性差
  - 可讀性差

```
WordAdapter adapter = new WordAdapter(TITLES, SUB_TITLES, ICONS);
```

# 物件化資料

- 解決方法
  - 將多個資料合併為一個物件
  - 增加擴充性
  - 增加容錯處理
  - 增加可讀性

```
public class DisplayWord {  
    int iconRedId;  
    String title;  
    String subTitle;  
}
```

# 物件化資料重點提示

- **ListView**每一個橫列所需的資料，以一個物件統整
- 物件的建立，獨立於**Adapter**之外
- 可多使用**ArrayList**來增加程式碼的簡易性



客製化ListView

# 建立VIEW HOLDER

# 建立VIEW HOLDER

- 目前問題？
  - getView()呼叫太多次findViewById()
  - ListView每一列與資料物件之間的組合不夠直覺

# 建立VIEW HOLDER

- 解決方法？
  - 建立ViewHolder
  - 讓findViewById()少做很多次
  - 讓ViewHolder與資料的物件一對一的作對應

# 建立VIEW HOLDER重點提示

- ViewHolder要使用convertView的setTag()跟View帶著一起走
- ViewHolder使用convertView的getTag()取出來重複使用

客製化ListView

使用GRADLE來擴增功能

# 使用GRADLE來擴增功能

- 目前問題？
  - 寫findViewById()寫了很多次，很麻煩
- 解決方法？
  - 使用奶油刀

# 使用GRADLE來擴增功能

- Gradle
  - Android Studio所使用的編譯系統
  - 可以藉由簡單宣告，便可加入第三方函式庫到App中
- gitHub專案：ButterKnife奶油刀
  - <http://jakewharton.github.io/butterknife/>
  - 使用Java annotation的機制，自動處理完findViewById()

# 使用奶油刀重點提示

- 打開Gradle Script的Module app的build.gradle，在dependencies加入以下項目

```
dependencies {  
    .....  
    compile 'com.jakewharton:butterknife:8.6.0'  
    annotationProcessor 'com.jakewharton:butterknife-compiler:8.6.0'  
}
```



# 使用奶油刀重點提示

- 在代表View的成員變數上，使用@BindView()即可

```
@BindView(R.id.listview)  
ListView mListView;
```

- 在Activity的setContentView()後，呼叫bind()即可

```
ButterKnife.bind(this);
```

- 在ViewHolder的建構子，也是使用bind()

```
ButterKnife.bind(this, convertView);
```

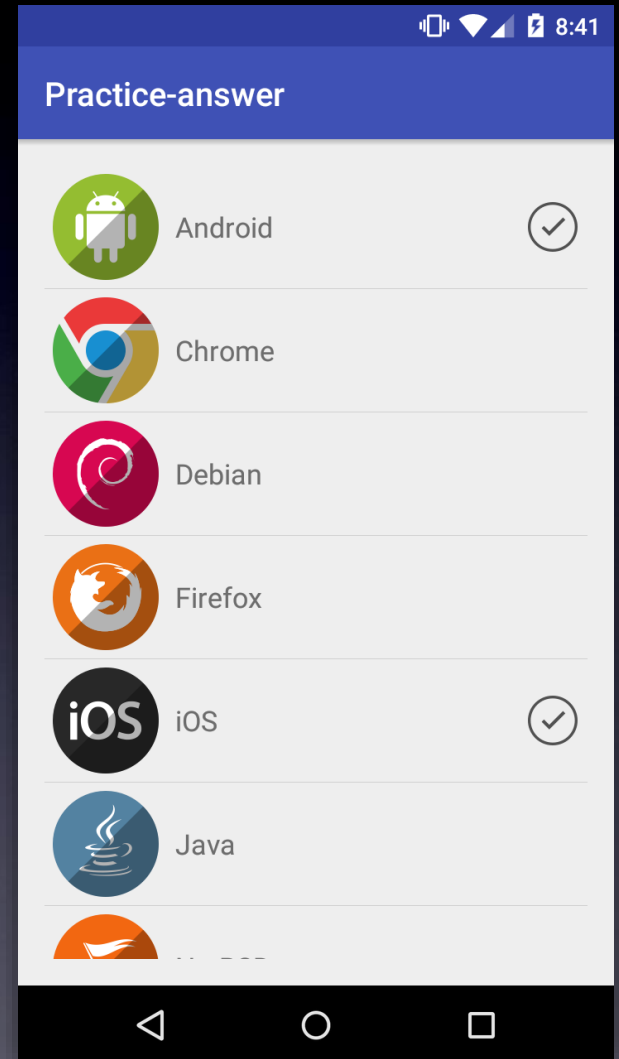
對應專案  
ListView/practice

客製化ListView

練習題

# 練習題

- 在專案中有眾多作業系統及技術的名稱
- 請呈現如右方畫面
  - 名稱都搭配正確的Icon
  - 名稱是為手機作業系統，最右方要顯示打勾
- 答案在branch  
ListView/practice-answer



Q & A