Android Resources (1 – 8)

[Group 1] Hanh Tran

hanh.usth@gmail.com

2 Nov 2016

I.   Chapter objectives

- Understand resources including definition and XML, basic layouts, ViewPager and TabLayout in Android and apply to practice.

II.   Resources

- <u>Description</u>: Things embedded into the app, in res/directory.
  Access through code **R.<category>.<resourceName>**.

- <u>Example</u>:
  ```
  MyProject/
      src/
          MyActivity.java
      res/
          drawable/
              graphic.png
          layout/
              main.xml
              info.xml
          mipmap/
              icon.png
          values/
              strings.xml
  ```

  Types of resources are mentioned below.

1.   Layouts

- <u>Description</u>: A way to organize View inside an UI.
  Hierarchical structure of one UI.
  Can be created by code, can be nested.
  XML files in **res/layout**.
  WYSIWYG or manual editor.

- <u>Example</u>: Popular layout classes:

- o FrameLayout
  - Can contain multi-children (Views).
  - Multi layers, Z-based order.
  - Support child margins and gravity.
  - First child will be at the bottom.

- o LinearLayout
  - One direction: horizontal or vertical.
  - Use layout_weight (no layout_weight → no stretch).
  - Based on orientation.
    - Horizontal: stretch width.
    - Vertical: stretch height.
  - Width/height w_i formula:

$$\omega_i = \frac{\gamma_i}{\sum_{j=0}^{n-1} \gamma_j} \times \left(\omega_{parent} - \sum_{k=0}^{n-1} \omega_k | \gamma_k = 0\right)$$

- o RelativeLayout
  - Multiple layers, Z-order based.
  - Relativity of children's position and size:
    - To parent.
    - To each other.
 - Children are relative to parent and to each other.

- o ViewPager
  - Tab-like container.
  - No header. Use a separate view for that.
    - Android design library's TabLayout.
  - Each tab content is a fragment.
  - ViewPager in XML Layout.
    - Example:

```xml
<android.support.v4.view.ViewPager
    android:id="@+id/pager"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />
```

  - "Adapter" in Java class (in parent Activity or Fragment), specify what fragment is in what page.

- Example:
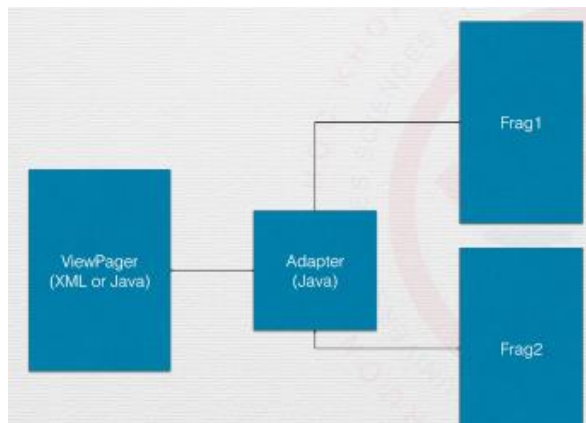
- Header for ViewPager:
  - build.gradle (of your app, not your project)
  - Example:

   compile "com.android.support:design:23.1.0"
  - Layout XML : above ViewPager.

2. Values

- <u>Description</u>:
  - Central point of all "constant", themeable, i18n
  - Size-, orientation-dependent (-large,-land…)
- <u>Example</u>: Default: **res/values/name-of-value-type.xml**
  - Strings
    - Description
      - Default: res/values/strings.xml
      - i18n: res/values-fr/strings.xml
    - Example:
      - Default: <string name=" mon" >Monday</string>
      - i18n: <string name=" mon" >Lundi</string>
  - Integers
    - Description
      - Default: res/values/integers.xml
      - Landscape: res/values-land/integers.xml
    - Example
      - Default: <integer name=" column_count" >1</integer>
      - Landscape: <integer name=" column_count" >2</integer>
  - Booleans

- Description
  - Default: res/values/bools.xml
  - Tablet: res/values-large/bools.xml
- Example
  - Default: <bool name=" is_tablet>false</bool>
  - Tablet: <bool name=" is_tablet>true</bool>
  - Colors
    - Description
      - Default: res/values/colors.xml
      - Tablet: res/values-large/colors.xml
    - Example
      - Default: <color name=" colorPrimary" >#3F51B5</color>
      - Tablet: <color name=" colorPrimary" >#FF4081</color>
  - Dimensions
    - Description
      - Default: res/values/dimens.xml
      - Tablet: res/values-large/dimens.xml
    - Example
      - Default: <dimen name=" title_width" >50dp</dimen>
      - Tablet: <dimen name=" title_width" >750dp</dimen>

3.   Drawables

- Description:
  - 2 types:
    - XML
    - Bitmap drawable
  - ImageView:scr=" @drawable/name"
  - View: background=" @drawable/name"
  - 9patch:Transparency,stretch certain part of the image,padding
    - Top & Left edge: black pixels define stretching area
    - Bottom & Eight edge: black pixel define paddings
    - Can be edited with many image tool.
- Example:
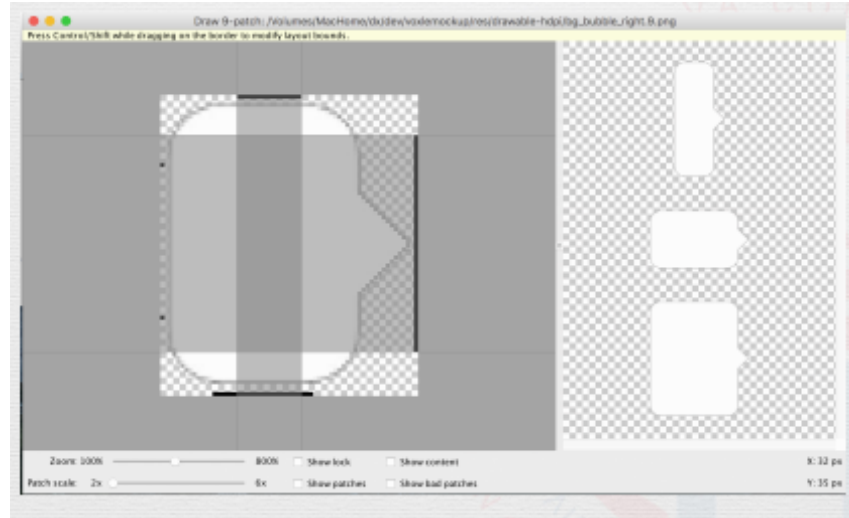
4.　Raw

- o Anything not common types embedded in your apps.
  - ▪ <u>Example</u>: Fonts,JSONs,Audio like MP3, even HTML/CSS/JS
- o Store in:
  - ▪ **res/raw/**
    - • <u>Description:</u>
      - o A resource
      - o Accessible with **Context.getRessources()**
      - o **R.raw.<name>**
    - • <u>Example:</u>
      ```
      InputStream is = context.getResources()
                          .openRawResource(R.raw.resid);
      ```
  - ▪ **assets/**
    - • <u>Description:</u>
      - o No resource, no **R.<assets>**
      - o Name it whatever you like
      - o Use input stream to access binary data
      - o Use file name with **Context.getAssets()**
    - • <u>Example:</u>
      ```
      InputStream is = getAssets().open("<filename>");
      ```

5.　Menu

- • <u>Description:</u>
  - o Reason: common controls, less crammed components, consistent...
  - o Types: Context Menu, Popup Menu, AppBar...

- Example:
  - App Bar
    - Description:
      - Android Support Library: ToolBar
      - Contains most common functions with app-wide scope
      - Create Layout:
        - Add ToolBar to Activity's layout
        - Setup it in **onCreate()**
        - Define Menu resource for AppBar
        - Inflate menu xml in **onCreateOptionsMenu()**
        - Response in **onOptionsItemSelected()**
    - Example: