Resources

http://developer.android.com/guide/topics/resources/overview.htm

Repo: https://github.com/karlmorris/Resources
Overview

- Working with resources In Android
- The R class
Android Resources

- All non-code assets associated with your application
  - Examples
    - layouts, strings, colors, styles, images
  - Full list
    - https://developer.android.com/guide/topics/resources/available-resources.html

- Resources are specified in various directories based on the resource type

- Accessed using @ in XML, and R in code
Purpose of Resources

- Allows independent maintenance of non-code assets
- Simplifies the provisioning of additional resources for various device configurations
- Allows automatic selection of alternate resources for various device configurations
Purpose of Resources

1) App without alternate resources

2) App with alternate resources
Defining Resources

- Resources are defined using Defaults, and Alternatives
- Default resources are used regardless of device configuration
  - e.g. res/layout – layout files
- Alternatives are selected when the device matches the resources stated configuration
  - Alternatives are specified with a suffix appended to the directory name
  - e.g. res/layout-land – layouts for landscape orientation
Defining Resources

- **Standard resource layout**

```
res/
    drawable/
        graphic.png
    layout/
        main.xml
        info.xml
    mipmap/
        icon.png
    values/
        strings.xml
```

- **Alternative resource layout**

```
res/
    drawable/
        icon.png
        background.png
    drawable-hdpi/
        icon.png
        background.png
...
```
Configuration Qualifiers

• Full list
  
  [link](https://developer.android.com/guide/topics/resources/providing-resources.html#table2)

• Multiple configurations can be specified for a single set of resources
  
  – e.g. **drawable-en-rUS-land**
  
  – Qualifiers separated by dashes
  
  – Must be specified in a predetermined order
Resource Selection

1. Eliminate qualifiers that contradict the device configuration
2. Identify the next qualifier in the table (MCC first, then MNC, then language, and so on)
3. Do any resource directories use this qualifier?
   - No
   - Yes
     - 5. Continue until only one directory for the desired resource is left
4. Eliminate directories that do not include this qualifier

* If the qualifier is screen density, the system selects the "best match" and the process is done
Accessing Resources

- Resources defined in your application are accessed using their **resource ID**
- Resource IDs are defined in a special class, `R`, which is automatically generated and maps names to IDs
- Resources are accessed by
  - In XML
    - `@<resource_type>/<resource_name>`
    - e.g. `android:text="@string/hello"`
  - In Code
    - `R.<resource_type>/<resource_name>`
    - e.g. `textView.setText(R.string.hello)`
Accessing Resources in XML

- Resources can be used in XML anywhere a simple value is expected
- Some situations require the use of resources
  - e.g.
    - Providing permission description text
    - Providing a drawable for a background
- Standard resources are accessed using @, style attributes are accessed using ?
  - ?[<package_name>:][<resource_type>/]<resource_name>
The R class

- Bridge between Assets and Code
- Automatically generated by Android
- Defines a subclass for each type of resource and a static integer for each resource
The R class

• Syntax

  [<package_name>].R
    .<resource_type>
    .<resource_name>

• package_name – optional when accessing resources in your own package

• resource_type – the subclass for the resource type

• resource_name – either the android:name attribute or the filename without the extension
Accessing Android Provided Resources

- Android provides numerous standard resources
  - styles, themes, layouts, strings, etc

- Accessible through the `android` package
  - XML e.g.
    `android:icon=@android:drawable/ic_camera`
  - Code e.g.
    ```java
    new ArrayAdapter<String>(this,
    android.R.layout.simple_list_item_1, myarray);
    ```