Application Basics Resources

Santa Clara University

Center for Science, Technology, and Society

What Are Resources?

All Android applications are composed of two things:

- Functionality (code instructions)
  - Functionality is the code that determines how your application behaves.
  - This includes any algorithms that make the application run.
- Data (resources)
  - Resources include text strings, styles and themes, dimensions, images and icons, audio files, videos, and other data used by the application.

Storing Application Resources

Android resource files are stored separately from the .java class files in the Android project.

- Most common resource types are stored in XML.
- You can also store raw data files and graphics as resources.
- Resources are organized in a strict directory hierarchy.
- All resources must be stored under the res/ project directory in specially named subdirectories that must be lowercase.

Resource Value Types

- Android applications rely on many different types of resources:
  - These include text strings, graphics, color schemes, and other resources for user interface design.
- These resources are stored in the `res` directory of your Android project in a strict (but reasonably flexible) set of directories and files.
- All resource filenames must be lowercase and simple.
  - They must use letters, numbers, and underscores only.

Storing Application Resources

Resource Subdirectory	Purpose
res/drawable/	Graphics resources
res/layout/	User interface resources
res/menu/	Menu resources for showing options or actions in your activities
res/mipmap/	App launcher icon resources
res/v	es and themes, and

Property animations	`res/animator/`
Tweened animations	`res/anim/`
Color state lists	`res/color/`
Drawables	`res/drawable/`
Layouts	`res/layout/`
Menus	`res/menu/`
Arbitrary raw files	`res/raw/`
Simple values	`res/values/`
Arbitrary XML	`res/xml/`

Storing Application Resources (2)

- **Resources can be further organized in a variety of ways using even more specially named directory qualifiers.**
  - **example:**
    - `res/drawable-hdpi/` stores graphics for high-density screens
    - `res/drawable-ldpi/` stores graphics for low-density screens
    - `res/drawable-mdpi/` stores graphics for medium-density screens
    - `res/drawable-xhdpi/` stores graphics for extra-high-density screens
    - `res/drawable-xxhdpi/` stores graphics for extra-extra-high-density screens
  - **If you had a graphic resource that was shared by all screens, you would simply store that resource in the `res/drawable/` directory.**