Location and Maps

Sang Shin
Michèle Garoche
www.javapassion.com
"Learn with Passion!"



Disclaimer

- Portions of this presentation are modifications based on work created and shared by the Android Open Source Project
 - http://code.google.com/policies.html
- They are used according to terms described in the Creative Commons 2.5 Attribution License
 - http://creativecommons.org/licenses/by/2.5/

Topics

- Location services
- Providing mock location data
- Google Maps

Location Services

Android Location Services

- Provided by LocationManager system service
 - > Call getSystemService(Context.LOCATION_SERVICE)
- Once your application has a handle to a LocationManager instance, your application will be able to do three things:
 - Query for the list of all LocationProviders known to the LocationManager for its last known location.
 - Register/unregister for periodic updates of current location from a LocationProvider (specified either by Criteria or name).
 - Register/unregister for a given Intent to be fired if the device comes within a given proximity (specified by radius in meters) of a given lat/long.

Providing Mock Location Data

Mock Location Data - Two Schemes

- Using DDMS (Dalvik Debug Monitor Service)
 - Manually send individual longitude/latitude coordinates to the device.
 - Use a GPX file describing a route for playback to the device GPX (GPS Exchange Format) is a light-weight XML data format for the interchange of GPS data
 - Use a KML file describing individual placemarks for sequenced playback to the device - KML is a file format used to display geographic data in an earth browser, such as Google Earth, Google Maps, and Google Maps for mobile.
- "geo" command line option
 - Launch your application in the Android emulator and open a terminal/console in your SDK's /tools directory

Google Maps

What is Google Maps External Library?

- Makes it easier for you to add powerful mapping capabilities to your application
 - > Downloading, rendering, and caching of Maps tiles
 - Variety of display options and controls

com.google.android.maps.MapView

- A subclass of ViewGroup
- Displays a map with data obtained from the Google Maps service.
- When the MapView has focus, it will capture keypresses and touch gestures to pan and zoom the map automatically, including handling network requests for additional maps tiles.
- Provides all of the UI elements necessary for users to control the map
 - Zooming control

com.google.android.maps.MapView

 In general, the MapView class provides a wrapper around the Google Maps API that lets your application manipulate Google Maps data through class methods, and it lets you work with Maps data as you would other types of Views.

MapView View Mode

- Default
- Satellite
 - > mapView.setSatellite(true);
- StreetView
 - > mapView.setStreetView(true);

Obtaining a Maps API Key

- Because MapView gives you access to Google Maps data, you need to register with the Google Maps service and agree to the applicable Terms of Service before your MapView will be able to obtain data from Google Maps.
- This will apply whether you are developing your application on the emulator or preparing your application for deployment to mobile devices.

Thank you!

Sang Shin
Michèle Garoche
http://www.javapassion.com
"Learn with Passion!"

