Lifecycle Management

Android

Mobile and Ubiquitous Games
ICS 163
Donald J. Patterson
Unlike traditional Java, Android does not use a “main” function.

It uses a sophisticated set of callbacks.

Each step of the callback corresponds to a step in the lifecycle of the app.

This is so that the phone can shut your app down when important things happen, like a phone call arriving or when a user switches apps.

An implementation of the Activity class contains the callbacks.
Callback


Your Activity
Callback

Android OS


notifyObservers()
for observer in observerCollection
    call observer.notify()
Lifecycle Management
Activity Lifecycle

- Key loops
- Entire Lifetime
  - `onCreate()` - `onDestroy()`

```
+notify()
```

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

- Key loops
- Visible Lifetime
  - `onStart()` - `onStop()`

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

- Key loops
- Foreground Lifetime
  - `onResume()` - `onPause()`
Activity Lifecycle

- onPause() may be followed by kill

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

Why do you care? So that your app ...

- Does not crash if the user receives a phone call or switches to another app while using your app.
- Does not consume valuable system resources when the user is not actively using it.
- Does not lose the user's progress if they leave your app and return to it at a later time.
- Does not crash or lose the user's progress when the screen rotates between landscape and portrait orientation.

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Stack

Diagram showing the concepts of stack with terms like Active, Front, Visible, Hidden, LIFO.
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

[Diagram showing the lifecycle of an Android activity, with states and corresponding methods like onCreate(), onStart(), onResume(), onPause(), onStop(), onDestroy().]

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

- Created
  - onCreate()
  - onRestart()
- Started (visible)
  - onStart()
  - onResume()
- Resumed (visible)
  - onResume()
  - onPause()
- Paused (partially visible)
  - onStop()
- Stopped (hidden)
  - onDestroy()
- Destroyed
Activity Lifecycle

http://developer.android.com/training/basics/activity-lifecycle/starting.html
Activity Lifecycle

- Created
- Started (visible)
  - onStart()
  - onCreate()
  - onRestart()
- Resumed (visible)
  - onResume()
- Paused (partially visible)
  - onPause()
- Stopped (hidden)
  - onStop()
- Destroyed

http://developer.android.com/training/basics/activity-lifecycle/starting.html
package ics163.luci.ics.uci.edu.gpsdrawapp;

import ...

public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        // Handle action bar item clicks here.
        // The action bar overlay will automatically handle clicks on the Home/Up button, so long
        // as you specify a parent activity in AndroidManifest.xml.
        int id = item.getItemId();

        //noinspection SimplifiableIfStatement
        if (id == R.id.action_settings) {
            return true;
        }

        return super.onOptionsItemSelected(item);
    }
}
The Manifest