Accelerating Information Technology Innovation

http://aiti.mit.edu

Cali, Colombia
Summer 2012
Lesson 9 – MultiThreading

MISTI

MultiThreading

- Execution of multiple tasks “in parallel”.
  - Asynchronously
- Incentive: Execute a task that may take a long time without blocking the main task
  - Does not affect user experience
- Solution: Use of AsyncTask()

AsyncTask()

- Performs background operations and publishes results to the UI thread
- Should ideally be used for short operations (a few seconds)
  - Example: Call web server, download file
  - For longer tasks use: Executor, FutureTask, and ThreadPoolExecutor
- Must be subclassed (cannot call AsyncTask directly)

AsyncTask() (cont)

- Uses generic type parameters

```
private class MyTask extends AsyncTask<Void, Void, Void> { ... }
```

- Input Parameters:
  - Params: Sent to task upon execution
  - Progress: Units published during background operation
  - Result: Type of result of the background operation

- Use Void for any unused type

Steps (methods) of AsyncTask()

- Execution methods
  - `onPreExecute()`
  - `doInBackground( Params… )`
  - `onProgressUpdate( Progress… )`
  - `onPostExecute( Result )`

- Task invoked (executed) by calling
  - `execute( Params )`

Example of AsyncTask()

```
private class DownloadFilesTask extends AsyncTask<URL, Integer, Long> {
    protected Long doInBackground(URL... urls) {
        int count = urls.length;
        long totalSize = 0;
        for (int i = 0; i < count; i++) {
            totalSize += Downloader.downloadFile(urls[i]);
            publishProgress((int) ((i / (float) count) * 100));
            // Escape early if cancel() is called
            if (isCancelled()) break;
        }
        return totalSize;
    }

    protected void onProgressUpdate(Integer... progress) {
        setProgressPercent(progress[0]);
    }

    protected void onPostExecute(Long result) {
        showDialog("Downloaded " + result + " bytes");
    }
}
```

Example of AsyncTask()

- Execution (invoke) example:

```
new DownloadFilesTask().execute(url1, url2, url3);
```

Rules of AsyncTask()

- The AsyncTask class must be loaded using the UI thread (foreground Activity)
- `execute(Params …)` must be invoked in the UI thread
- Do not call any of the other class methods manually.

Resources

- AsyncTask description
  - http://developer.android.com/reference/android/os/AsyncTask.html

- Gargenta – Learning Android
  - Chapter 6 -> Threading in Android