Specifying the Code to Run on a Thread

This lesson teaches you to

- Define a Class that Implements Runnable
- Implement the run() Method

You should also read

- Processes and Threads

Try it out

DOWNLOAD THE SAMPLE

ThreadSample.zip

This lesson shows you how to implement a `Runnable` (https://developer.android.com/reference/java/lang/Runnable.html) class, which runs the code in its `Runnable.run()` (https://developer.android.com/reference/java/lang/Runnable.html#run()) method on a separate thread. You can also pass a `Runnable` (https://developer.android.com/reference/java/lang/Runnable.html) to another object that can then attach it to a thread and run it. One or more `Runnable` (https://developer.android.com/reference/java/lang/Runnable.html) objects that perform a particular operation are sometimes called a *task*.

`Thread` (https://developer.android.com/reference/java/lang/Thread.html) and `Runnable` (https://developer.android.com/reference/java/lang/Runnable.html) are basic classes that, on their own, have only limited power. Instead, they're the basis of powerful Android classes such as `HandlerThread` (https://developer.android.com/reference/android/os/HandlerThread.html), `AsyncTask` (https://developer.android.com/reference/android/os/AsyncTask.html), and `IntentService` (https://developer.android.com/reference/android/app/IntentService.html). `Thread` (https://developer.android.com/reference/java/lang/Thread.html) and `Runnable` (https://developer.android.com/reference/java/lang/Runnable.html) are also the basis of the class `ThreadPoolExecutor` (https://developer.android.com/reference/java/util/concurrent/ThreadPoolExecutor.html). This class automatically manages threads and task queues, and can even run multiple threads in parallel.

Define a Class that Implements Runnable

Implementing a class that implements `Runnable` (https://developer.android.com/reference/java/lang/Runnable.html) is straightforward. For example:

```
public class PhotoDecodeRunnable implements Runnable {
    ...
    @Override
    public void run() {
        /*
         * Code you want to run on the thread goes here
         */
        ...
    }
    ...
}
```

Implement the run() Method

In the class, the `Runnable.run()` (https://developer.android.com/reference/java/lang/Runnable.html#run()) method contains the code that's executed. Usually, anything is allowable in a `Runnable` (https://developer.android.com/reference/java/lang/Runnable.html). Remember, though, that the `Runnable` (https://developer.android.com/reference/java/lang/Runnable.html) won't be running on the UI thread, so it can't directly modify UI objects such as `View` (https://developer.android.com/reference/android/view/View.html) objects. To communicate with the UI thread, you have to use the techniques described in the lesson Communicate with the UI Thread (https://developer.android.com/training/multiple-threads/communicate-ui.html).

At the beginning of the `run()` (https://developer.android.com/reference/java/lang/Runnable.html#run()) method, set the thread to use background priority by calling `Process.setThreadPriority()` (https://developer.android.com/reference/android/os/Process.html#setThreadPriority(int)) with `THREAD_PRIORITY_BACKGROUND` (https://developer.android.com/reference/android/os/Process.html#THREAD_PRIORITY_BACKGROUND). This approach reduces resource competition between the `Runnable` (https://developer.android.com/reference/java/lang/Runnable.html) object's thread and the UI thread.

You should also store a reference to the `Runnable` (https://developer.android.com/reference/java/lang/Runnable.html) object's `Thread` (https://developer.android.com/reference/java/lang/Thread.html) in the `Runnable` (https://developer.android.com/reference/java/lang/Runnable.html) itself, by calling `Thread.currentThread()` (https://developer.android.com/reference/java/lang/Thread.html#currentThread()).

The following snippet shows how to set up the `run()` (https://developer.android.com/reference/java/lang/Runnable.html#run()) method:

```
class PhotoDecodeRunnable implements Runnable {
...
    /*
     * Defines the code to run for this task.
     */
    @Override
    public void run() {
        // Moves the current Thread into the background
        android.os.Process.setThreadPriority(android.os.Process.THREAD_PRIORITY_BACKGROUND
        ...
        /*
         * Stores the current Thread in the PhotoTask instance,
         * so that the instance
         * can interrupt the Thread.
         */
        mPhotoTask.setImageDecodeThread(Thread.currentThread());
        ...
    }
...
}
```