Multithreaded Programming: Basics

1. Make a coin-flipping class that implements Runnable. The run method should flip 1000 coins and print out whenever they get 3 or more consecutive heads. Start 5 instances of the coin-flipping task. In the printouts, you can use the Thread.currentThread().getName() to identify the thread. Remember to look in the DDMS window to see the print results.

   You are following variation 1 of the basic threading approach, so your code will look something like this:

   ```java
   public class Flipper implements Runnable {
       public void run() { loop and do coin flipping }
   }
   -----------------------------------------------------
   public class MainClass extends Activity {
       public void buttonHandler(View clickedButton) {
           ExecutorService taskList = ...;
           for(int i=0; i<5; i++) {
               taskList.execute(new Flipper());
           }
       }
   }
   ```

2. Do a similar task, but this time have your Activity implement Runnable. Now your code will look roughly like this:

   ```java
   public class MainClass extends Activity implements Runnable {
       public void buttonHandler(View clickedButton) {
           ExecutorService taskList = ...;
           for(int i=0; i<5; i++) {
               taskList.execute(this);
           }
           public void run() { loop and do coin flipping }
       }
   }
   ```

3. Do a similar task again, but this time use an inner class for the Flipper (the class that implements Runnable).