

package com.example.ai.application.HandlerThreadAnnotation;  
  
import android.app.Activity;  
import android.content.Context;  
import android.graphics.Color;  
import android.os.Handler;  
import android.os.Looper;  
import android.os.Message;  
import android.widget.TextView;  
  
*/\*\*  
 \*  
 \*/*public class updataUI {  
  
 private TextView textView;  
  
 public updataUI(TextView textView)  
 {  
 this.textView = textView;  
 }  
  
 public void byHandlerPost(Looper looper)  
 {  
 final Handler handler = new Handler(looper);  
  
 new Thread()  
 {  
 @Override  
 public void run() {  
 try {  
 Thread.*sleep*(2000);  
 //实际上为给UI线程发送一条消息，而UI线程对应的handler接收到消息时所执行的方法为Runnable  
  **handler.post(new Runnable() {  
 @Override  
 public void run() {  
 textView.setText("by Thread handler.post runnable");  
 textView.setBackgroundColor(Color.*GRAY*);  
 }  
 });** } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
  
 }  
 }.start();  
 }  
  
 public void byHandlerSendMessage(Looper looper)  
 {  
 final Handler handler = new Handler(looper){  
 @Override  
 public void handleMessage(Message msg) {  
 textView.setText("by handler sendMessage");  
 textView.setBackgroundColor(Color.*BLUE*);  
 }  
 };  
 new Thread(){  
 @Override  
 public void run() {  
 try {  
 Thread.*sleep*(4000);  
  **handler.sendEmptyMessage(1);** } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
 }  
 }.start();  
 }  
  
 public void byRunOnUIThread(Activity activity)  
 {  
 //源码先判断是否为UI线程，后通过Handler.post(Runnable)在UI线程中执行Runnable  
  **activity.runOnUiThread(new Runnable() {  
 @Override  
 public void run() {//该方法在UI线程中执行  
 textView.setText("byRunOnUIThread");  
 textView.setBackgroundColor(Color.RED);  
 }  
 });** }  
  
 public void byViewPost()  
 {  
 //本质仍为Handler  
  **textView.post(new Runnable() {  
 @Override  
 public void run() { //该方法在UI线程中执行  
 textView.setBackgroundColor(Color.DKGRAY);  
 textView.setText("byViewPost");  
 }  
 });** }  
}

在某些情况下非UI线程也可以更新UI，……………rootimpl………..？