**西 安 邮 电 大 学**

**（计算机学院）**

《Android应用开发》课内实验报告

# 实验四：多线程应用

**专业名称： 计算机科学与技术**

**班 级： 计科1608**

**学 号： 05168066**

**学生姓名： 祁泽宇**

**指导教师： 张庆生**

**起止时间： 2019年5月20日**

1. 课程设计目的
2. 掌握多线程的基本概念
3. 掌握Handler类的基本用法
4. 掌握AsyncTask类的基本用法
5. 掌握Timer类的基本用法
6. 掌握多线程的综合应用
7. 实验环境

Jdk:1,8,0\_40

IDE:Android Studio

1. 实验内容

启动应用先显示一个图片“正在获取网络图片”，打开新线程从网络下载图片，下载完毕替换“正在获取网络图片”；单击“查看更多”按钮，在打开新界面中使用多线程从本地资源中随机选择图片进行切换显示。

1. 实验过程及分析
2. 创建Android项目，修改布局文件，并补全代码

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity"  
 android:orientation="vertical">  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="产品介绍"  
 android:textStyle="bold"  
 android:textSize="25dp"  
 android:gravity="center"  
 android:id="@+id/itemlist"  
 />  
 <ImageView  
 android:id="@+id/iv"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:src="@drawable/ing" />  
 <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="查看更多"  
 android:onClick="add"/>  
  
  
  
</LinearLayout>

1. 主Activity类MainActivity.java 代码所示如下

package com.example.class6;  
  
import android.content.Intent;  
import android.graphics.Bitmap;  
import android.graphics.BitmapFactory;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.ImageView;  
  
import java.io.InputStream;  
import java.net.URL;  
import java.net.URLConnection;  
  
public class MainActivity extends AppCompatActivity {  
  
 private ImageView iv = null;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 iv = (ImageView) findViewById(R.id.iv);  
 //开启一个线程从网络加载图片  
 new Thread(new Runnable() {  
 @Override  
 public void run() {  
 //可修改为其他网络的图片地址  
 final Bitmap bm =getPic("https://i.loli.net/2019/05/20/5ce21829b949c78220.jpg");  
 try{  
 Thread.sleep(2000); //睡眠两秒  
 }  
 catch ( Exception e)  
 {  
 e.printStackTrace();  
 }  
 //再开始一个线程设置从网络下载图片  
 iv.post(new Runnable() {  
 @Override  
 public void run() {  
 //请在此位置设置下载的图片显示  
 iv.setImageBitmap(bm);  
 }  
 });  
 }  
 }).start();  
 }  
 private Bitmap getPic(String path)  
 {  
 Bitmap bm = null;  
 try{  
 URL url = new URL(path);  
 URLConnection conn = url.openConnection();  
 conn.connect();  
 InputStream is = conn.getInputStream();  
 bm = BitmapFactory.decodeStream(is);  
  
  
 }catch (Exception e)  
 {  
 e.printStackTrace();  
 }  
 return bm;  
 }  
 //启动第二个界面  
 public void add(View v)  
 {  
 Intent intent = new Intent();  
 intent.setClass(this,MyThread.class);  
 startActivity(intent);  
 }  
}

1. 修改布局文件activity\_thread.xml,并补全代码

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
 <ImageView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:id="@+id/iv1"  
 android:adjustViewBounds="true"/>  
  
  
</LinearLayout>

1. 修改Activity类文件MyThread.java，代码如下所示

package com.example.class6;  
import android.annotation.SuppressLint;  
import android.os.Bundle;  
import android.os.Handler;  
import android.os.Message;  
import android.support.v7.app.AppCompatActivity;  
import android.widget.ImageView;  
  
import java.util.Random;  
  
  
public class MyThread extends AppCompatActivity implements Runnable{  
 private ImageView iv = null;  
 private Handler handler = null;  
 private int[] path = {R.drawable.ing,R.drawable.ic\_launcher\_background};  
 private Thread t;  
 @SuppressLint("handlerLeak")  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_thread);  
 iv = (ImageView) findViewById(R.id.iv1);  
 t = new Thread( this);  
 t.start();  
 handler = new Handler(){  
 @Override  
 public void handleMessage(Message msg) {  
 if(msg.what == 0x111)  
 {  
 iv.setImageResource(path[msg.arg1]);  
 super.handleMessage(msg);  
 }  
  
  
 }  
 };  
  
 }  
 @Override  
 public void run(){  
 int index = 0;  
 while(!Thread.currentThread().isInterrupted()){  
 index = new Random().nextInt(path.length);  
 Message msg = handler.obtainMessage();  
 msg.arg1 = index;  
 msg.what = 0x111;  
 handler.sendMessage(msg);  
 try{  
 Thread.sleep(2000);  
 }catch (Exception e)  
 {  
 e.printStackTrace();  
 }  
 }  
 }  
 @Override  
 protected void onDestroy() {  
 super.onDestroy();  
 t.interrupt();  
 //请在此关闭线程  
 }  
  
  
}

1. 在AndroidManifest.xml文件中增加网络许可，代码如下

<uses-permission android:name="android.permission.INTERNET"/>

1. 实验结果

图1 下载图片前 图2 下载图片后

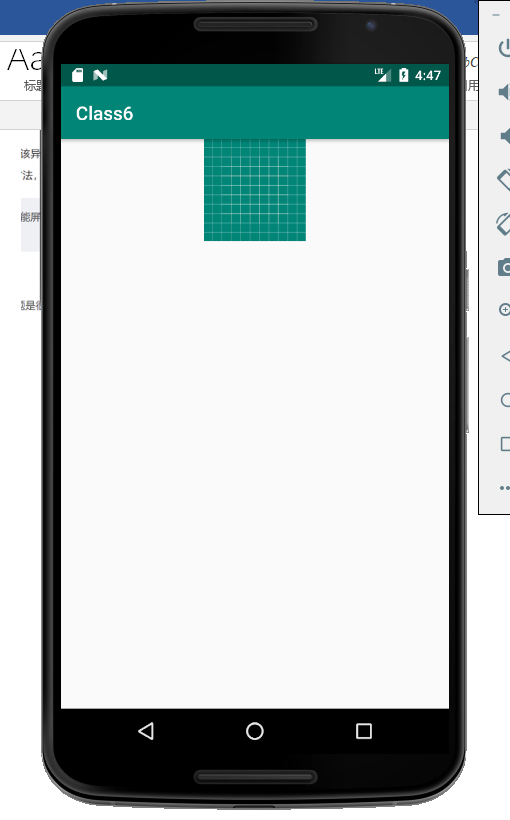
 

图3 切换图片1 图4 切换图片2

1. 总结

本次实验我们简单的尝试了多线程应用，多线程常用于游戏，多媒体程序，有网络连接的程序、进行耗时计算等的程序中，Android使用Thread类创建线程，并提供用于发送消息和处理消息的类Handler。在后台工作中比较简单，只需要向UI线程传递一些简单数据的环境下，Android还提供了AsyncTask类，此外，Timer定时器也是实现多线程程序的方式。这次实验并没有涉及到这两个类，但可以添加来达到同样的效果，因此我准备在课余时间继续学习。更好的掌握Thread类的用法。