# 实验5 网络编程

|  |  |  |  |
| --- | --- | --- | --- |
| 学号 | 202231060901 | 姓名 | 刘洋 |
| 专业 | 物联网工程 | 年级 | 2022 |

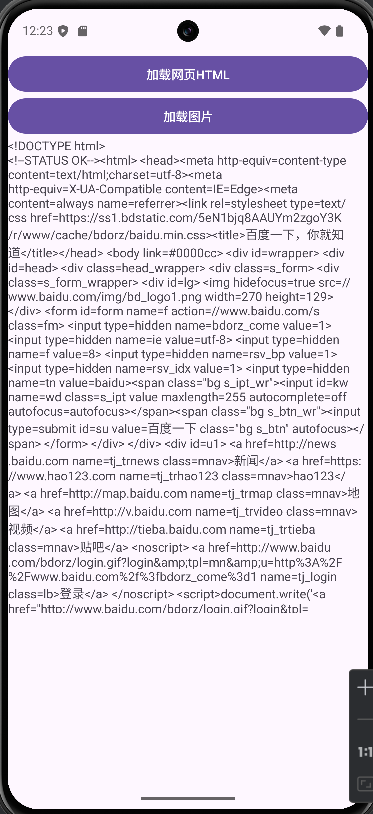
特别说明：实验报告必须命名为【学号紧跟姓名\_实验项目名】的格式，不包含括号【】。

例如【202231060901\_刘洋\_Android实验5.docx】，不按此命名不得分。

实验报告提交地址：<https://pintia.cn/>

## 简单网络访问(30分)

1. 运行效果图



1. MainActivity的Java代码(**是要文本，不是截图**)：

|  |
| --- |
| package edu.swpu.network;  import android.graphics.Bitmap;  import android.graphics.BitmapFactory;  import android.os.Bundle;  import android.widget.ImageView;  import android.widget.TextView;  import androidx.activity.EdgeToEdge;  import androidx.appcompat.app.AppCompatActivity;  import androidx.core.graphics.Insets;  import androidx.core.view.ViewCompat;  import androidx.core.view.WindowInsetsCompat;  import java.io.BufferedReader;  import java.io.InputStreamReader;  import java.net.HttpURLConnection;  import java.net.URL;  public class MainActivity extends AppCompatActivity {  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  EdgeToEdge.enable(this);  setContentView(R.layout.activity\_main);    // 创建按钮点击事件处理  findViewById(R.id.btnLoadHtml).setOnClickListener(v -> loadWebPage());  findViewById(R.id.btnLoadImage).setOnClickListener(v -> loadImage());    ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {  Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());  v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);  return insets;  });  }    private void loadWebPage() {  new Thread(() -> {  try {  URL url = new URL("https://www.baidu.com");  HttpURLConnection conn = (HttpURLConnection) url.openConnection();  BufferedReader reader = new BufferedReader(new InputStreamReader(conn.getInputStream()));  StringBuilder sb = new StringBuilder();  String line;  while ((line = reader.readLine()) != null) {  sb.append(line).append("\n");  }  reader.close();    // 在UI线程更新结果  runOnUiThread(() -> {  TextView textView = findViewById(R.id.textResult);  textView.setText(sb.toString());  });  } catch (Exception e) {  e.printStackTrace();  // 添加错误提示  runOnUiThread(() -> {  TextView textView = findViewById(R.id.textResult);  textView.setText("加载失败: " + e.getMessage());  });  }  }).start();  }    private void loadImage() {  new Thread(() -> {  try {  URL url = new URL("https://www.news.cn/local/20240619/42e3e1c931df4e81babb6bc38e8ff750/5e29ba2e128a404f8b73a5b161cf09cf.jpg");  final Bitmap bitmap = BitmapFactory.decodeStream(url.openConnection().getInputStream());    // 在UI线程更新图片  runOnUiThread(() -> {  ImageView imageView = findViewById(R.id.imageView);  imageView.setImageBitmap(bitmap);  });  } catch (Exception e) {  e.printStackTrace();  }  }).start();  }  } |

## 获取基金信息(30分)

1. 运行效果图



1. 布局文件XML代码

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <androidx.constraintlayout.widget.ConstraintLayout 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:id="@+id/main"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MainActivity">  <EditText  android:id="@+id/fundCodeInput"  android:layout\_width="200dp"  android:layout\_height="wrap\_content"  android:hint="请输入6位基金代码"  android:inputType="number"  android:maxLength="6"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toStartOf="@+id/searchButton"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  app:layout\_constraintVertical\_bias="0.3" />  <Button  android:id="@+id/searchButton"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="查看"  app:layout\_constraintBottom\_toBottomOf="@+id/fundCodeInput"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toEndOf="@+id/fundCodeInput"  app:layout\_constraintTop\_toTopOf="@+id/fundCodeInput" />  <TextView  android:id="@+id/fundInfoText"  android:layout\_width="0dp"  android:layout\_height="0dp"  android:layout\_margin="16dp"  android:scrollbars="vertical"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/fundCodeInput" />  </androidx.constraintlayout.widget.ConstraintLayout> |

1. MainActivity的代码

|  |
| --- |
| package edu.swpu.iot2022.liuyang;  import android.os.Bundle;  import android.util.Log;  import android.view.View;  import android.widget.Button;  import android.widget.EditText;  import android.widget.TextView;  import androidx.activity.EdgeToEdge;  import androidx.appcompat.app.AppCompatActivity;  import androidx.core.graphics.Insets;  import androidx.core.view.ViewCompat;  import androidx.core.view.WindowInsetsCompat;  import org.json.JSONObject;  import java.io.BufferedReader;  import java.io.InputStreamReader;  import java.net.HttpURLConnection;  import java.net.URL;  import java.util.regex.Matcher;  import java.util.regex.Pattern;  public class MainActivity extends AppCompatActivity {  private EditText fundCodeInput;  private TextView fundInfoText;  private Button searchButton;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  EdgeToEdge.enable(this);  setContentView(R.layout.activity\_main);  ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {  Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());  v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);  return insets;  });  fundCodeInput = findViewById(R.id.fundCodeInput);  fundInfoText = findViewById(R.id.fundInfoText);  searchButton = findViewById(R.id.searchButton);  searchButton.setOnClickListener(new searchBtnCallback());  }  // 添加按钮点击事件监听器  public class searchBtnCallback implements View.OnClickListener  {  @Override  public void onClick(View view) {  String fundCode = fundCodeInput.getText().toString();  fetchFund fund = new fetchFund(fundCode);  fund.fetch(new FundCallback() {  @Override  public void onSuccess(FundData fundData) {  // 格式化显示基金信息  String displayText = String.format(  "基金名称：%s\n代码：%s\n净值日期：%s\n单位净值：%s\n估算净值：%s\n估算涨跌幅：%s%%\n估值时间：%s",  fundData.name, fundData.fundcode, fundData.jzrq,  fundData.dwjz, fundData.gsz, fundData.gszzl, fundData.gztime  );  fundInfoText.setText(displayText);  }  @Override  public void onError(String error) {  fundInfoText.setText("获取数据失败：" + error);  }  });  }  }  /\*\*  \* 基金数据结构类  \*/  public static class FundData {  public String fundcode; // 基金代码  public String name; // 基金名称  public String jzrq; // 净值日期  public String dwjz; // 单位净值  public String gsz; // 估算净值  public String gszzl; // 估算涨跌幅  public String gztime; // 估值时间  }  /\*\*  \* 基金数据获���类  \* 用于从天天基金网获取基金实时数据  \*/  public class fetchFund {  // 基金代码，默认为易方达新收益混合A(001186)  private String fundCode = "001186";  // 天天基金网API接口地址，%s为基金代码占位符，%d为时间戳占位符  private final String baseUrl = "http://fundgz.1234567.com.cn/js/%s.js?rt=1463558676006";    /\*\*  \* 构造函数  \* @param fundCode 基金代码  \*/  public fetchFund(String fundCode) {  this.fundCode = fundCode;  }    /\*\*  \* 设置基金代码  \* @param fundCode 新的基金代码  \*/  public void setFundCode(String fundCode) {  this.fundCode = fundCode;  }  /\*\*  \* 使用正则表达式解析JSONP格式的基金数据  \* @param jsonpData JSONP格式的字符串  \* @return 解析后的FundData对象  \*/  private FundData parseJsonpData(String jsonpData) throws Exception {  // 使用正则表达式匹配JSONP中的JSON数据  Pattern pattern = Pattern.compile("jsonpgz\\((.+?)\\);?");  Matcher matcher = pattern.matcher(jsonpData);    if (!matcher.find()) {  throw new Exception("数据格式不正确");  }    // 获取JSON字符串  String jsonStr = matcher.group(1);  JSONObject jsonObject = new JSONObject(jsonStr);    FundData fundData = new FundData();  fundData.fundcode = jsonObject.getString("fundcode");  fundData.name = jsonObject.getString("name");  fundData.jzrq = jsonObject.getString("jzrq");  fundData.dwjz = jsonObject.getString("dwjz");  fundData.gsz = jsonObject.getString("gsz");  fundData.gszzl = jsonObject.getString("gszzl");  fundData.gztime = jsonObject.getString("gztime");    return fundData;  }  /\*\*  \* 获取基金数据  \* @param callback 回调接口，用于处理请求结果  \*/  public void fetch(FundCallback callback) {  // 创建新线程执行网络请求，避免阻塞主线程  new Thread(() -> {  try {  // 构建请求URL，添加当前时间戳防止缓存  URL url = new URL(String.format(baseUrl, fundCode));  Log.d("fund","url : "+url);  HttpURLConnection connection = (HttpURLConnection) url.openConnection();  connection.setRequestMethod("GET");    int responseCode = connection.getResponseCode();  if (responseCode == HttpURLConnection.HTTP\_OK) {  // 读取响应数据  BufferedReader reader = new BufferedReader(  new InputStreamReader(connection.getInputStream()));  StringBuilder response = new StringBuilder();  String line;    while ((line = reader.readLine()) != null) {  response.append(line);  }  reader.close();    // 解析数据  FundData fundData = parseJsonpData(response.toString());    // 在主线程中更新UI  runOnUiThread(() -> {  callback.onSuccess(fundData);  });  } else {  // 请求失败，返回错误信息  callback.onError("请求失败: " + responseCode);  }  connection.disconnect();  } catch (Exception e) {  // 发生异常时返回错误信息  callback.onError(e.getMessage());  }  }).start();  }  }  // 修改回调接口  interface FundCallback {  void onSuccess(FundData fundData);  void onError(String error);  }  } |

## 网络图片浏览器(30分)

你构建的项目的相关代码

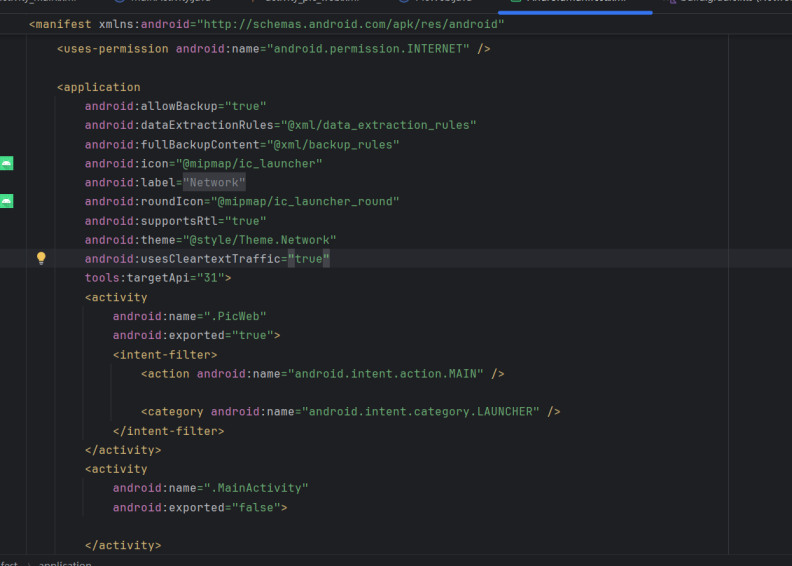
1. MainActivity的代码(是要文本，不是截图)

|  |
| --- |
| package edu.swpu.iot2022.liuyang;  import android.graphics.Bitmap;  import android.graphics.BitmapFactory;  import android.os.Bundle;  import android.widget.Button;  import android.widget.EditText;  import android.widget.ImageView;  import android.widget.Toast;  import androidx.activity.EdgeToEdge;  import androidx.appcompat.app.AppCompatActivity;  import java.io.InputStream;  import java.net.HttpURLConnection;  import java.net.URL;  public class PicWeb extends AppCompatActivity {  private ImageView imageView;  private EditText urlEditText;  private Button loadButton;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  EdgeToEdge.enable(this);  setContentView(R.layout.activity\_pic\_web);    // 初始化视图  imageView = findViewById(R.id.imageView);  urlEditText = findViewById(R.id.urlEditText);  loadButton = findViewById(R.id.loadButton);  loadButton.setOnClickListener(v -> {  String imageUrl = urlEditText.getText().toString();  if (!imageUrl.isEmpty()) {  loadImage(imageUrl);  }  });  }  private void loadImage(String imageUrl) {  new Thread(() -> {  final Bitmap bitmap = downloadImage(imageUrl);  // 在主线程更新UI  runOnUiThread(() -> {  if (bitmap != null) {  imageView.setImageBitmap(bitmap);  } else {  Toast.makeText(PicWeb.this, "加载图片失败", Toast.LENGTH\_SHORT).show();  }  });  }).start();  }  private Bitmap downloadImage(String imageUrl) {  try {  URL url = new URL(imageUrl);  HttpURLConnection connection = (HttpURLConnection) url.openConnection();  connection.setDoInput(true);  connection.connect();  InputStream input = connection.getInputStream();  return BitmapFactory.decodeStream(input);  } catch (Exception e) {  e.printStackTrace();  return null;  }  }  } |

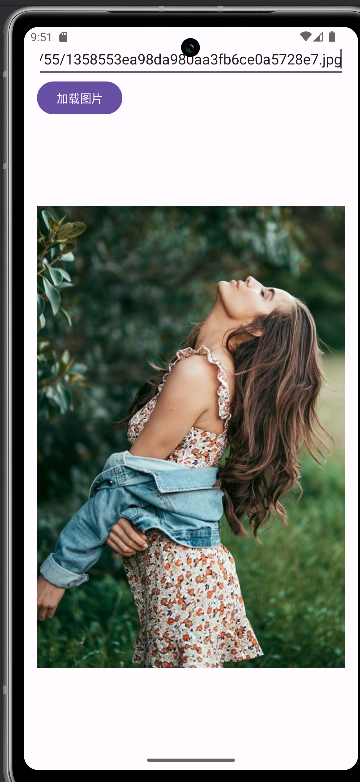
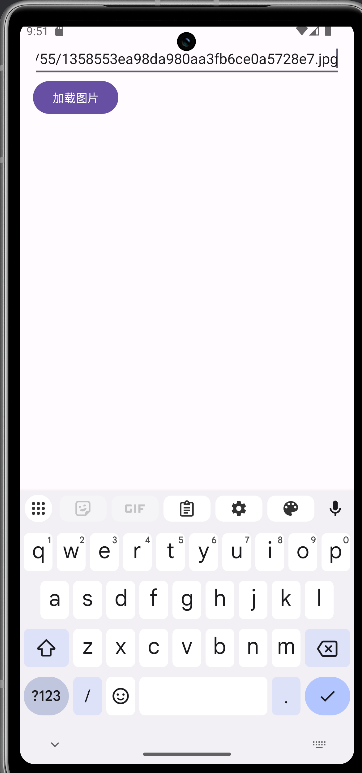
1. 布局文件的XML代码(是要文本，不是截图)

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:id="@+id/main"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:padding="16dp">  <EditText  android:id="@+id/urlEditText"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:hint="输入图片URL"  android:inputType="textUri" />  <Button  android:id="@+id/loadButton"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="加载图片" />  <ImageView  android:id="@+id/imageView"  android:layout\_width="match\_parent"  android:layout\_height="0dp"  android:layout\_weight="1"  android:scaleType="fitCenter" />  </LinearLayout> |

1. AndroidManifest.xml文件的截图，此处是截图



1. 运行效果截图，多图



## 实验心得体会(10分)

要求不少于100汉字，必须是自己原创，不得雷同。

|  |
| --- |
| 在实现图片加载功能时，我遇到了几个典型问题：首先是在主线程中直接进行网络请求导致应用崩溃，通过学习了解到Android禁止在主线程进行网络操作，需要创建新线程处理；其次是图片加载后无法更新UI的问题，这让我理解了Android的线程机制，必须通过runOnUiThread在主线程更新界面。 |