

# 学习任务三 开发天气预报小程序

## 子任务 3.2 下载网络数据

### 1、任务引入：

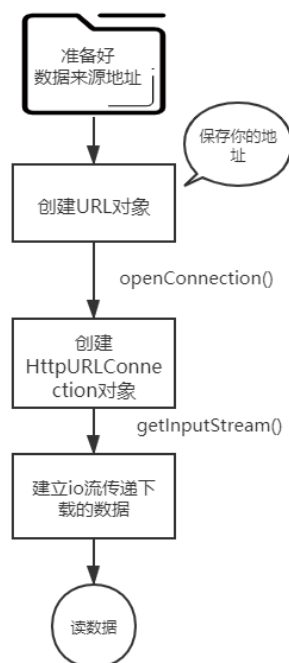
上一个任务完成的界面，需要实现根据城市，显示该城市当天的天气，那么当天的天气是从哪里来的，网络上有很多天气预报的数据，可供下载。实际上我们身边有很多这样的应用，例如我们的手机，每天可以看到当天的天气预报，就是从固定的网络资源地址下载的数据应用。

所以这个子任务，我们要学习如何利用 Java 网络访问技术，实现网页资讯的下载。

### 2、任务准备（微课学习内容）

#### 1) 理解网络数据下载的流程

Java 同样也有提供类库，支撑网络数据下载的操作，例如  
URL 类: 定义网络上的资源地址  
URLConnection 类: 建立与 Http 服务器的连接从而获取数据  
具体下载流程如下图所示：



【小练习 1】大家试一试使用这两个类如何把手机号码信息从网络上取下来。



```
package net;
import java.io.*;
import java.net.HttpURLConnection;
import java.net.URL;
public class CodeDemo {
    public static void main(String args[]) {
        String codecity=""; //接受从网络上返回的数据
        String url="https://www.baifubao.com/callback?cmd=1059&callback=phone&phone=13312997213";
        try {
            URL weburl=new URL(url); //创建一个资源地址
            HttpURLConnection conn=(HttpURLConnection)weburl.openConnection(); //创建一个和资源地址相关的连接对象
            conn.connect(); //发出连接命令
            //从这个连接通道中取出返回的数据
            BufferedReader br=new BufferedReader(new InputStreamReader(conn.getInputStream()));
            codecity=br.readLine();
            System.out.println(codecity);
        } catch (Exception e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

Markers Properties Servers Data Source Explorer Snippets Console

main (5) [Java Application] D:\vm\ eclipse\bin\javaw.exe (2021年11月25日 下午11:21:47)

https://www.baifubao.com/callback?cmd=1059&callback=phone&phone=13312997213

\*fgg\_again\*/phone({ "meta": { "result": "0", "result\_info": "", "jump\_url": "" }, "data": { "operator": "电信", "area": "广东", "area\_operator": "广东电信", "support":

## 2) 使用第三方工具简化网络数据下载

OkHttp 是一个高效的 HTTP 客户端，它有以下默认特性：

- ✓ 支持 HTTP/2，允许所有同一个主机地址的请求共享同一个 socket 连接
- ✓ 连接池减少请求延时
- ✓ 透明的 GZIP 压缩减少响应数据的大小
- ✓ 缓存响应内容，避免一些完全重复的请求

当网络出现问题的时候 OkHttp 依然坚守自己的职责，它会自动恢复一般的连接问题，如果你的服务有多个 IP 地址，当第一个 IP 请求失败时，OkHttp 会交替尝试你配置的其他 IP。OkHttp 的使用是非常简单的。它的请求/响应 API 使用构造器模式 builders 来设计，它支持阻塞式的同步请求和带回调的异步请求。

总的来说，这种网络数据的下载方式，比 Java 原生代码的性能要好的多。

我们使用 OkHttp 发送数据异步处理请求，一般分为如下三步：

- a) 下载并导入两个外部资源包：okhttp 和 okio；
- b) 构建 request 和 response 对象
- c) 发送或提取数据

网上有很多案例实现 okhttp 的两种数据请求格式 GET/POST,同步或者异步请求数据。这里我们通过下面小练习，体验一下同步 GET 下载网络数据的过程

【小练习 2】大家试一试使用 okhttp Get 同步的方式从网络上取手机号码的信息。

```
package net;
import java.io.*;
import java.net.HttpURLConnection;
import java.net.URL;
public class CodeDemo {
    public static void main(String args[]) {
        String url= "https://www.baifubao.com/callback?cmd=1059&callback=phone&phone=13312997213";
        System.out.println(url);
        OkHttpClient okHttpClient = new OkHttpClient();
        Response response;
        Request request = new Request.Builder().url(url).get().build();
        try {
            response = okHttpClient.newCall(request).execute();
            String result=response.body().string();
            System.out.println(result);
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

Markers Properties Servers Data Source Explorer Snippets Console

tain (5) [Java Application] D:\mj\ eclipse\bin\javaw.exe (2021年11月25日 下午11:21:47)

https://www.baifubao.com/callback?cmd=1059&callback=phone&phone=13312997213

\*fegg\_again\*/phone({"meta":{"result":"0","result\_info":"","jump\_url":""},"data":{"operator":"电信","area":"广东","area\_operator":"广东电信","support

### 3) 使用线程提升数据下载性能

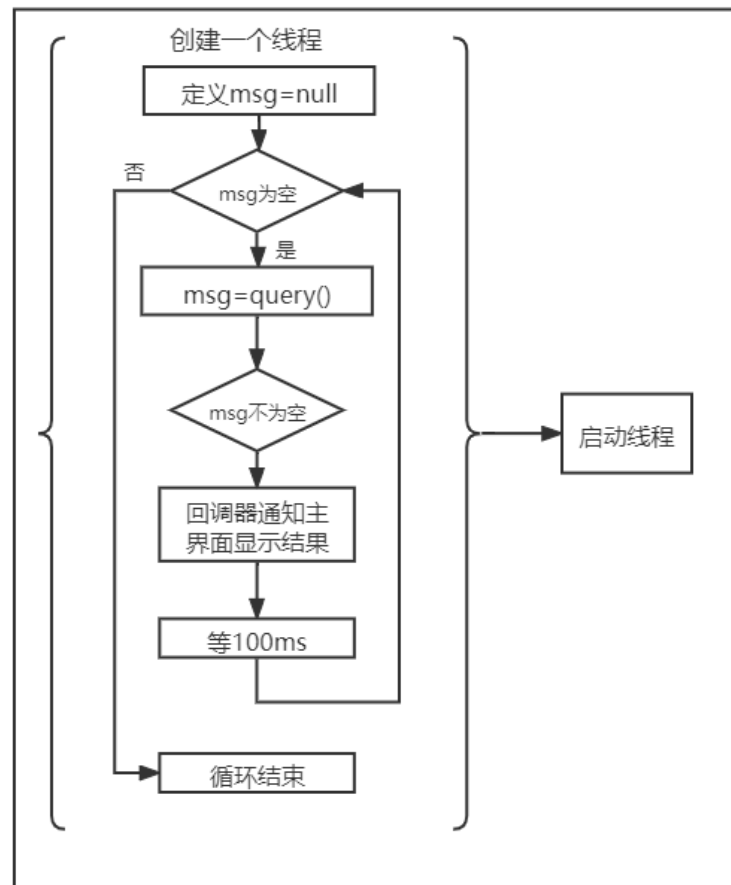
具体的改造思路如下图：



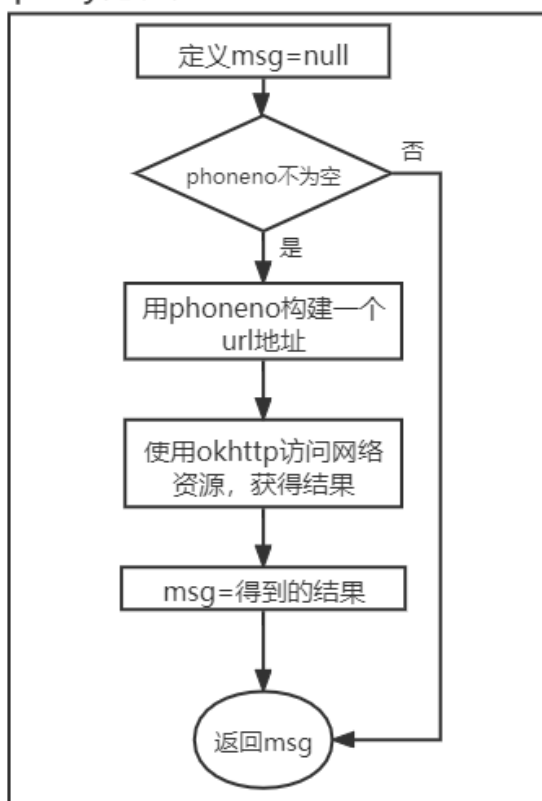
这个类图中，TestDown 类主要获得用户信息，执行查询，MobilePhone 类是核心业务逻辑类，负责启动下载数据的线程，执行下载的操作。这个类中的

三个核心方法是：start 方法、query 方法、setPhoneno 方法，他们的流程图如下所示：

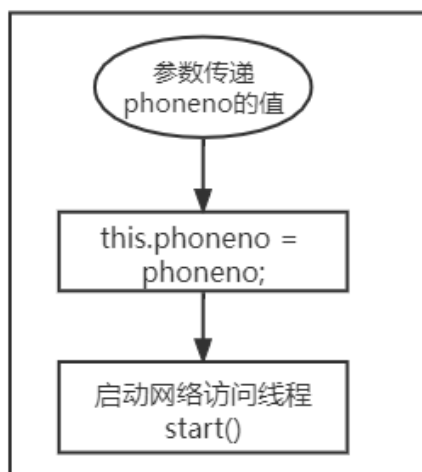
### start方法



## query方法



## setPhoneno方法



TestDown 类的参考代码如下所示:

\*\*\*\*\*

```
public class TestDown {  
    MobilePhone phone;  
    Callable call;  
    public void testdown() {  
        Scanner s=new Scanner(System.in); //得到指向键盘的输入流对象
```

```

        System.out.println("请输入你的手机号码");
        String phoneno=s.nextLine();//获得要查询的手机号码
        phone.setPhoneno(phoneno);

    }
    public interface Callable {
        public void showDate(String msg);
    }
    public TestDown() {
        call=new Callable() {

            @Override
            public void showDate(String msg) {
                System.out.println("*****以下是从网络下载的数据*****");
                System.out.println(msg);
            }
        };
        phone=new MobilePhone(call);
    }
    public static void main(String[] args) {
        new TestDown().testdown();
    }
}

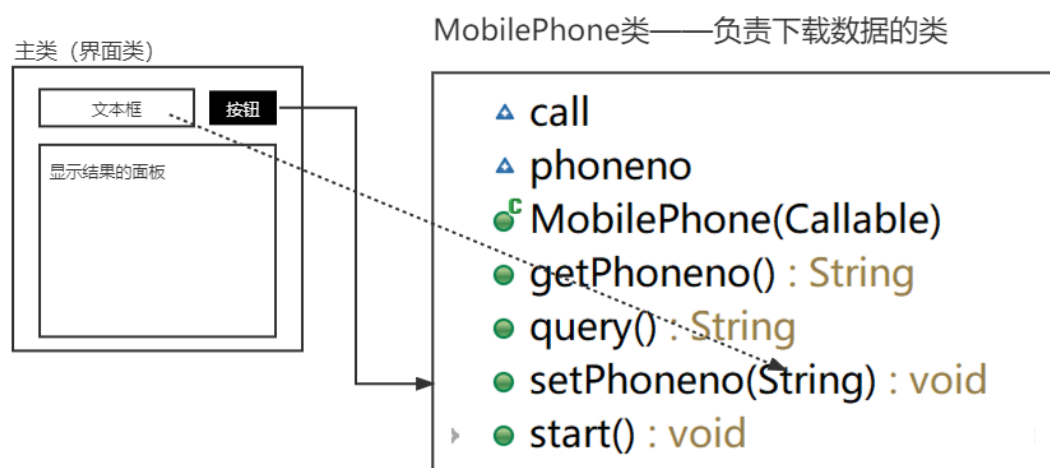
*****

```


请大家结合前面的类图，完成 MobilePhone 类的设计代码。

### 3、技能训练：

根据上面所学的内容，设计图形化界面完成手机信息的下载。



最终完成的效果如下图所示：

 查询手机号码信息

13312997213

查询

```
/*fgg_again*/phone({"meta":{"result":"0","result_info":"","jump_url":""},"data":{"operator":"电信","area":"广东","area_operator":"广东电信","support_price":{"10000":"9995","20000":"19990","3000":"2998","30000":"29985","5000":"4998","50000":"49974"},"promotion_info":null}})
```