

## 1、InetAddress

InetAddress类用于标识网络上的硬件资源，标识互联网协议(IP)地址。该类没有构造方法

```
public class InetAddress {  
    public static void main(String[] args) throws IOException {  
        //创建一个表示本地主机的InetAddress对象  
        InetAddress localAddress = InetAddress.getLocalHost();  
        //获得指定主机的InetAddress对象  
        InetAddress remoteAddress = InetAddress.getByName("www.info-soft.cn");  
        //得到IP地址的主机名。  
        System.out.println("本机的IP地址: "+localAddress.getHostName());  
        //获得字符串格式的原始IP地址  
        System.out.println("中科信软的IP地址: "+remoteAddress.getHostAddress());  
        //判断指定的时间内地址是否可以到达  
        System.out.println("3秒是否可达: "+remoteAddress.isReachable(3000));  
        System.out.println("itcast的主机名为: "+remoteAddress.getHostName());  
    }  
}
```

## 2、URL类

1)、URL(Uniform Resource Locator)统一资源定位符，表示Internet上某一资源的地址，协议名：资源名称

```
public class URLLDemo {  
    public static void main(String[] args) throws MalformedURLException {  
        URL baidu =new URL("http://www.baidu.com");  
        //? 表示参数，#表示锚点  
        URL url =new URL(baidu,"/index.html?username=tom#test");  
        //获取协议  
        System.out.println(url.getProtocol());  
        //获取主机  
        System.out.println(url.getHost());  
        //如果没有指定端口号，根据协议不同使用默认端口。此时getPort()方法的返回  
        值为 -1  
        System.out.println( url.getPort());  
        //获取文件路径  
        System.out.println( url.getPath());  
    }  
}
```

```

//文件名，包括文件路径+参数
System.out.println(url.getFile());
//相对路径，就是锚点，即#号后面的内容
System.out.println(url.getRef());
//查询字符串，即参数
System.out.println(url.getQuery());
}
}

```

## 2)、使用URL读取网页内容

通过URL对象的openStream()方法可以得到指定资源的输入流，通过流能够读取或访问网页上的资源。

```

/**
 * 使用URL读取网页内容
 */
public class URLReadDemo {
    public static void main(String[] args) throws IOException {
        URL url =new URL("http://www.baidu.com");
        //通过openStream方法获取资源的字节输入流
        InputStream is = url.openStream();
        //将字节输入流转换为字符输入流,如果不指定编码，中文可能会出现乱码
        InputStreamReader isr =new InputStreamReader(is,"UTF-8");
        BufferedReader br =new BufferedReader(isr);
        //读取数据
        String data = br.readLine();
        while(data!=null){
            //输出数据
            System.out.println(data);
            data = br.readLine();
        }
        br.close();
        isr.close();
        is.close();
    }
}

```

