1. 网络基础知识

1-1 网络基础简介

·端口号范围为0~65535，其中0~1023位系统所保留

·http：80 ftp：21 telnet：23

·针对网络通信的不同层器，Java提供的网络功能有四大类：

·InetAddress：用于标识网络上的硬件资源

·URL： 统一资源定位符 通过URL可以直接读取或写入网络上的数据

·Sockets： 使用TCP协议实现网络通信的Socket相关的类

·Datagram： 使用UDP协议，将数据保存在数据报中，通过网络进行通信

1. Java中网络相关API的应用

2-1 Java中InetAddress的引用

·InetAddress类用于标识网络上的硬件资源，标识互联网协议（IP）地址



2-2 Java中URL的应用

·URL（Uniform Resource Locator）同意资源定位符，表示Internet上某一资源的地址

·URL由两部分组成：协议名称和资源名称，中间用冒号隔开



·在java.net包中，提供了URL类来表示URL。

·URL imooc = new URL("http://www.imooc.com");

URL url = new URL(imooc,"/index.html?username=tom#test");//?后面表示参数#后面表示锚点

System.out.println("协议："+url.getProtocol());

System.out.println("主机："+url.getHost());

//如果未指定端口号，则使用默认的端口号，此时getPort（）方法返回值为-1

System.out.println("端口："+url.getPort());

System.out.println("文件路径："+url.getPath());

System.out.println("文件名"+url.getFile());

System.out.println("相对路径："+url.getRef());//相对路径即锚点

System.out.println("查询字符串:"+url.getQuery());

·通过URL对象的openStream()方法可以得到指定资源的输入流。

·通过输入流可以读取访问网络上的数据

1. 通过Socket实现TCP编程

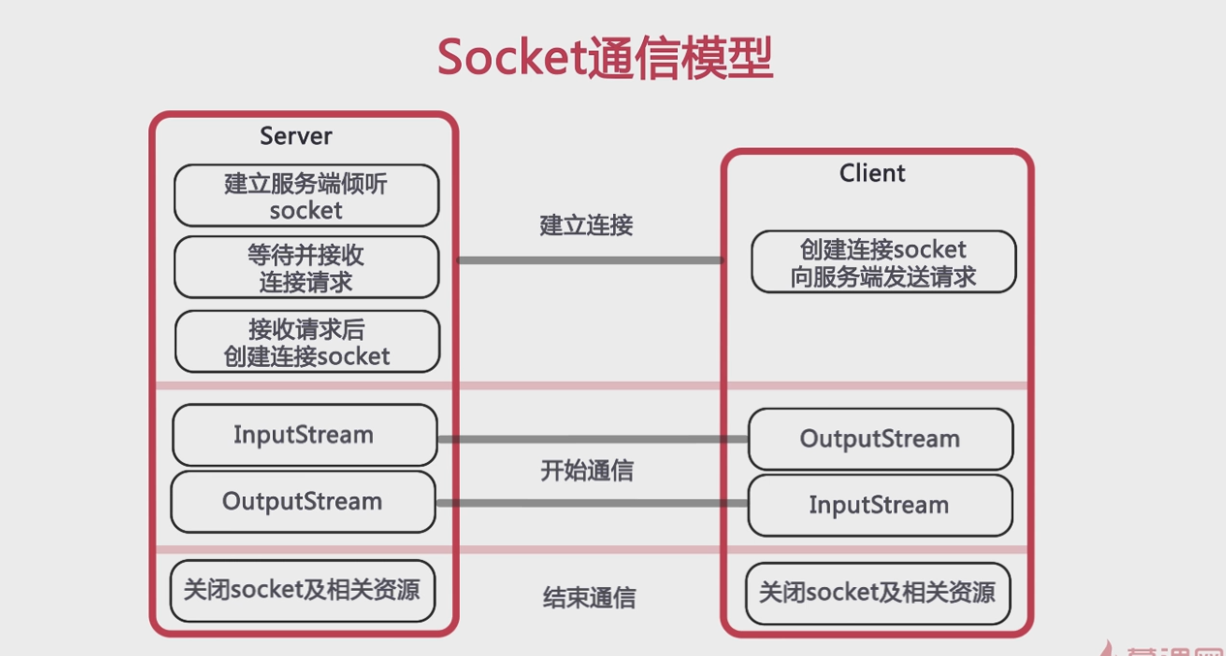
3-1 Socket简介

·TCP协议是面向连接、可靠地、有序的，以字节流的方式发送数据

·基于TCP协议实现网络通信的类

·客户端的Socket类

·服务器端的ServerSocket类



Socket通信实现步骤

·创建ServerSocket和Socket

·打开连接到Socket的输入/输出流

·按照协议对Socket进行读/写操作

·关闭输入输出流、关闭Socket

3-2 编程实现基于TCP的Socket通信之服务器端

·创建ServerSocket对象，绑定监听窗口

·通过accept()方法监听客户端请求

·连接建立后，通过输入流读取客户端发送的请求信息

·通过输出流向客户端发送响应信息

·关闭相关资源

3-3 编程实现基于TCP的Socket通信之客户端

·创建Socket对象，指明需要连接的服务器的地址和端口号

·连接建立后，通过输出流向服务器端发送请求信息

·通过输入流获取服务器响应的信息

·关闭相关资源

3-4 完善用户登录之服务器响应客户端

3-5 使用多线程实现多客户端的通信

·多线程服务器

应用多线程来实现服务器与多客户端之间的通信

基本步骤

·服务器端创建ServerSocket，循环调用accept()等待客户端连接

·客户端传建一个Socket并请求和服务器端连接

·服务器端接收客户端请求，创建Socket与该客户建立专线连接

·建立连接的两个Socket在一个单独的线程上对话

·服务器端继续等待新的连接

1. 通过Socket实现UDP编程

4-1 DatagramPacket

·UDP编程

UDP协议（用户数据保协议）是无连接、不可靠的、无序的，速度较快

UDP协议以数据报作为数据传输的载体

进行数据传输时，首先需要将传输的数据定义成数据报（Datagram），在数据报中指明 数据所要达到的 Socket（主机地址和端口号），然后再将数据报发送出去

·相关操作类

DatagramPacket：表示数据报包

DatagramSocket：进行端到端通信的类

4-2 编程实现基于UDP的Socket通信之服务器端

·服务器端实现步骤

·创建DatagramSocket，指定端口号

·创建DatagramPacket

·接收客户端发送的数据信息

·读取数据

4-3 编程实现基于UDP的Socket通信之客户端

·客户端实现步骤

·定义发送信息

·创建DatagramPacket，包含将要发送的信息

·创建DatagramSocket

·发送数据

1. Socket总结

5-1 Socket总结

·多线程的优先级

未设置优先级可能会导致运行时速度非常慢，可降低优先级

·是否关闭输入流和输出流

对于同一个socket，如果关闭了输出流，则与该输出流关联的socket也会被关闭，所以一般不用关闭流，直接关闭socket即可

·使用TCP通信传输对象

1. 综合练习

6-1 综合练习---问题描述

6-2 综合练习---实现分析