

# 24 网络编程

JAVAAEE高级

## 一：网络编程描述

- 1、Java提供的网络类库，可以实现无痛的网络连接联网的底层细节被隐藏在 Java 的本机安装系统里，由 JVM 进行控制
- 2、并且 Java 实现了一个跨平台的网络库，**程序员面对的是一个统一的网络编程环境**

### 1、计算机网络

- 1、把分布在不同地理区域的计算机与专门的外部设备用通信线路互连成一个规模大、功能强的网络系统，从而使众多的计算机可以方便地互相传递信息、共享硬件、软件、数据信息等资源

### 2、网络编程

- 1、就是用来实现网络互连的不同计算机上运行的程序间可以进行数据交换
- 2、说白了就是直接或间接地通过网络协议与其它计算机进行通讯

## 二：网络通信协议

- 1、一定的规则（有两套参考模型）

### 1、OSI参考模型

1、模型过于理想化，未能在因特网上进行广泛推广

## 2、TCP/IP参考模型

2、事实上的国际标准

OSI参考模型	TCP/IP参考模型	TCP/IP参考模型 各层对应协议
应用层	应用层	HTTP、ftp、 telnet、DNS...
表示层		
会话层		
传输层	传输层	TCP、UDP、...
网络层	网络层	IP、ICMP、ARP...
数据链路层	物理+数据链路层	Link
物理层		

## 三：网络编程三要素

### 1、IP概述

- 1、每个设备在网络中的唯一标识(类似身份证号码唯一)
- 2、每台网络终端在网络中都有一个独立的地址，我们在网络中传输数据就是使用这个地址
- 3、ipconfig：查看本机IP192.168.12.42

### 2、端口号概述

- 1、每个程序在设备上的唯一标识

- 2、每个网络程序都需要绑定一个端口号
- 3、传输数据的时候除了确定发到哪台机器上，还要明确发到哪个程序
- 4、端口号范围从0-65535
- 5、编写网络应用就需要绑定一个端口号，尽量使用1024以上的，1024以下的基本上都被系统程序占用了
- 6、常用默认的端口
  - mysql: 3306
  - oracle: 1521
  - sqlserver:1433
  - tomcat: 8080
  - http:80

### 3、网络通信协议

- 1、计算机网络中实现通信必须有一些约定，即通信协议，对速率、传输代码、代码结构、传输控制步骤、出错控制等制定标准
- 2、说白了就是为计算机网络中进行数据交换而建立的规则、标准或约定的集合

#### ①：通信协议分层的思想

- 1、由于结点之间联系很复杂，在制定协议时，把复杂成份分解成一些简单的成份，再将它们复合起来
- 2、最常用的复合方式是层次方式，即同层间可以通信、上一层可以调用下一层，而与再下一层不发生关系
- 3、各层互不影响，利于系统的开发和扩展

#### ②：传输层协议中有两个非常重要的协议

##### I：UDP

- 1、面向无连接，数据不安全，速度快。不区分客户端与服务端

- 2、将数据、源、目的封装成数据包，不需要建立连接
- 3、发送数据结束时无需释放资源从而速度快

## II : TCP

- 1、面向连接（三次握手），数据安全，速度略低
- 2、TCP协议进行通信的两个应用进程：客户端、服务端
- 3、三次握手方式:  
客户端先向服务端发起请求, 服务端响应请求, 传输数据
- 4、在连接中可进行大数据量的传输，传输完毕，需释放已建立的连接，效率低

## 四 : InetAddress类

- 1、InetAddress类没有提供公共的构造器
- 2、而是提供了如下两个静态方法来获取InetAddress实例
- 3、如获取IP和域名

```
1.      @Test
2.      public void getVoid() throws UnknownHostException{
3.          InetAddress byName = InetAddress.getByName("www.baidu.com");
4.          System.out.println("byName:"+byName);
5.          System.out.println("对象所包的域名:"+byName.getHostName());
6.          System.out.println("对象所包含的IP:"+byName.getHostAddress());
7.
8.          InetAddress HostbyName = InetAddress.getLocalHost();
9.          System.out.println("获取本机的域名和IP地址:"+HostbyName);
10.     }
```