

[首页](#) [ANDROID](#) [互联网](#) [杂乱无章](#) [科技资讯](#) [程序员人生](#) [程序员笑话](#) [编程技术](#) [网址导航](#)

7.6.1 Socket学习网络基础准备

分类 [Android 基础入门教程](#)

Android 基础入门教程(Q群号 : 153836263)

本节引言：

为了照顾没学过Java Socket的初学者，或者说捋一捋Android开发中涉及到的网络协议相关的概念，毕竟面试的时候，面试官来了句给我说下网络协议有几层？那么IP协议在哪层？Socket是什么鬼？分哪几种？TCP和UDP协议又在哪层？有什么区别...嗯，这...所以学习本节概念性的理论还是很有必要的！那么话不多说，开始本节内容~

1.OSI七层网络模型浅析

当然，我们不是专业搞网络工程的，只要知道有哪些层，大概是拿来干嘛的就可以了！

OSI七层网络模型(从下往上)：

物理层(Physical)：设备之间的数据通信提供传输媒体及互连设备，为数据传输提供可靠的环境。可以理解为网络传输的物理媒体部分，比如**网卡，网线，集线器，中继器，调制解调器**等！在这一层，数据还没有被组织，仅作为原始的位流或电气电压处理，这一层的单位是：**bit比特**

数据链路层(DataLink)：可以理解为数据通道，主要功能是如何在不可靠的物理线路上进行数据的可靠传递，该层作用包括：物理地址寻址，数据的成帧，流量控制，数据检错以及重发等！另外这个**数据链路指的是**：物理层要为终端设备间的数据通信提供传输媒体及其连接。媒体是长期的，连接是有生存期的。在连接生存期内，收发两端可以进行不等的一次或多次数据通信。每次通信都要经过建立通信联络和拆除通信联络两过程！这种建立起来的数据收发关系~ 该层的设备有：**网卡，网桥，网路交换机**，另外该层的单位为：**帧**

网络层(Network)：主要功能是将网络地址翻译成对应的物理地址，并决定如何将数据从发送方路由到接收方，所谓的路由与寻径：一台终端可能需要与多台终端通信，这样就产生的了把任意两台终端设备数据链接起来的问题！简单点说就是：建立网络连接和为上层提供服务！该层的设备有：**路由**！该层的单位为：**数据包**，另外IP协议就在这一层！

传输层(Transport)：向上面的应用层提供通信服务，面向通信部分的最高层，同时也是用户功能中的最低层。接收会话层数据，在必要时将数据进行分割，并将这些数据交给网络层，并且保证这些数据段有效的到达对端！所以这层的单位是：**数据段**；而这层有两个很重要的协议就是：**TCP传输控制协议与UDP用户数据报协议**，这也是本章节核心讲解的部分！

[反馈](#)

- 1.0 Android基础入门教程
 - 1.0.1 2015年最新Android基...
 - 1.1 背景相关与系统架构分析
 - 1.2 开发环境搭建
 - 1.2.1 使用Eclipse + ADT + S...
 - 1.2.2 使用Android Studio开...
 - 1.3 SDK更新不了问题解决
 - 1.4 Genymotion模拟器安装
 - 1.5.1 Git使用教程之本地仓...
 - 1.5.2 Git之使用GitHub搭建...
 - 1.6 .9(九妹)图片怎么玩
 - 1.7 界面原型设计
 - 1.8 工程相关解析(各种文件...
 - 1.9 Android程序签名打包
 - 1.11 反编译APK获取代码&...
- 2.1 View与ViewGroup的概念
 - 2.2.1 LinearLayout(线性布局)
 - 2.2.2 RelativeLayout(相对布...
 - 2.2.3 TableLayout(表格布局)
 - 2.2.4 FrameLayout(帧布局)
 - 2.2.5 GridLayout(网格布局)
 - 2.2.6 AbsoluteLayout(绝对...
- 2.3.1 TextView(文本框)详解
- 2.3.2 EditText(输入框)详解
- 2.3.3 Button(按钮)与ImageB...
- 2.3.4 ImageView(图像视图)
- 2.3.5.RadioButton(单选按钮...
- 2.3.6 开关按钮ToggleButton...
- 2.3.7 ProgressBar(进度条)
- 2.3.8 SeekBar(拖动条)
- 2.3.9 RatingBar(星级评分条)
- 2.4.1 ScrollView(滚动条)

会话层(Session)：负责在网络中的两节点之间建立、维持和终止通信。建立通信链接，保持会话过程通信链接的畅通，同步两个节点之间的对话，决定通信是否被中断以及通信中断时 决定从何处重新发送，即不同机器上的用户之间会话的建立及管理！

表示层(Presentation)：对来自应用层的命令和数据进行解释，对各种语法赋予相应的含义，并按照一定的格式传送给会话层。其主要功能是"处理用户信息的表示问题，如编码、数据格式转换和加密解密，压缩解压缩"等

应用层(Application)：OSI参考模型的最高层，为用户的应用程序提供网络服务。它在其他6层工作的基础上，负责完成网络中应用程序与网络操作系统之间的联系，建立与结束使用者之间的联系，并完成网络用户提出的各种网络服务及应用所需的监督、管理和服务等各种协议。此外，该层还负责协调各个应用程序间的工作。应用层为用户提供的服务和协议有：文件服务、目录服务、文件传输服务（FTP）、远程登录服务（Telnet）、电子邮件服务（E-mail）、打印服务、安全服务、网络管理服务、数据库服务等。

好的上面我们浅述了OSI七层网络模型，下面总结下：

OSI是一个理想的模型，一般的网络系统只涉及其中的几层，在七层模型中，每一层都提供一个特殊 的网络功能，从网络功能角度观察：

下面4层（物理层、数据链路层、网络层和传输层）主要提供数据传输和交换功能，即以节点到节点之间的通信为主

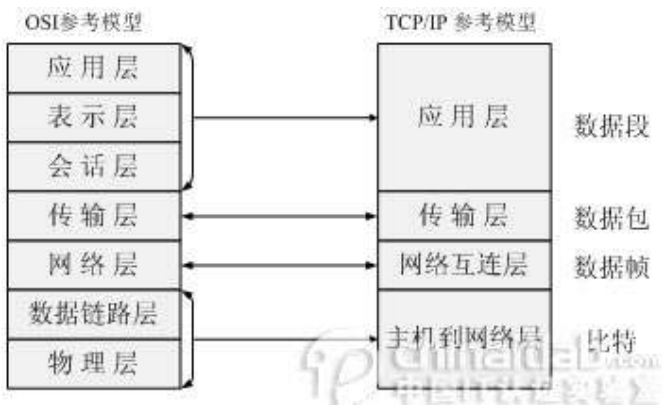
第4层作为上下两部分的桥梁，是整个网络体系结构中最关键的部分；

上3层（会话层、表示层和应用层）则以提供用户与应用程序之间的信息和数据处理功能为主。

简言之，下4层主要完成通信子网的功能，上3层主要完成资源子网的功能。

——以上内容参考自：[OSI七层模型详解](#)

2.TCP/IP四层模型



TCP/IP是一组协议的代名词，它还包括许多协议，组成了TCP/IP协议簇。TCP/IP协议簇分为四层，IP位于协议簇的第二层(对应OSI的第三层)，TCP位于协议簇的第三层 (对应OSI的第四层)。TCP/IP通讯协议采用了4层的层级结构，每一层都呼叫它的下一层所提供 的网络来完成自己的需求。这4层分别为：

应用层：应用程序间沟通的层，如简单电子邮件传输（SMTP）、文件传输协议（FTP）、网络远程访问协议（Telnet）等。

- 2.4.2 Date & Time组件(上)
- 2.4.3 Date & Time组件(下)
- 2.4.4 Adapter基础讲解
- 2.4.5 ListView简单实用
- 2.4.6 BaseAdapter优化
- 2.4.7ListView的焦点问题
- 2.4.8 ListView之checkbox错...
- 2.4.9 ListView的数据更新问题
- 2.5.0 构建一个可复用的自定...
- 2.5.1 ListView Item多布局的...
- 2.5.2 GridView(网格视图)的...
- 2.5.3 Spinner(列表选项框)...
- 2.5.4 AutoCompleteTextVie...
- 2.5.5 ExpandableListView(...
- 2.5.6 ViewPager(翻转视图)...
- 2.5.7 Toast(吐司)的基本使用
- 2.5.8 Notification(状态栏通...
- 2.5.9 AlertDialog(对话框)详解
- 2.6.0 其他几种常用对话框基...
- 2.6.1 PopupWindow(悬浮框...
- 2.6.2 菜单(Menu)
- 2.6.3 ViewPager的简单使用
- 2.6.4 DrawerLayout(官方侧...
- 3.1.1 基于监听的事件处理机制
- 3.2 基于回调的事件处理机制
- 3.3 Handler消息传递机制浅析
- 3.4 TouchListener PK OnTo...
- 3.5 监听EditText的内容变化
- 3.6 响应系统设置的事件(Co...
- 3.7 AsyncTask异步任务
- 3.8 Gestures(手势)
- 4.1.1 Activity初学乍练
- 4.1.2 Activity初窥门径
- 4.1.3 Activity登堂入室
- 4.2.1 Service初涉
- 4.2.2 Service进阶
- 4.2.3 Service精通
- 4.3.1 BroadcastReceiver牛...
- 4.3.2 BroadcastReceiver庖...
- 4.4.1 ContentProvider初探

传输层：在此层中，它提供了节点间的数据传送服务，如传输控制协议（TCP）、用户数据报协议（UDP）等，TCP和UDP给数据包加入传输数据并把它传输到下一层中，这一层负责传送数据，并且确定数据已被送达并接收。

网络互连层：负责提供基本的数据封包传送功能，让每一块数据包都能够到达目的主机（但不检查是否被正确接收），如网际协议（IP）。

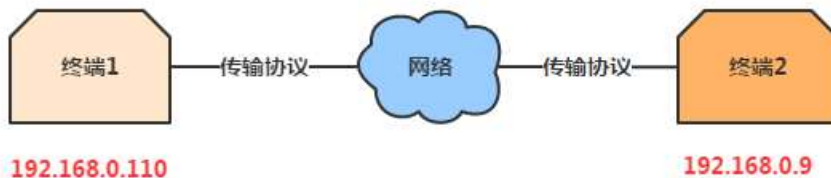
主机到网络层：对实际的网络媒体的管理，定义如何使用实际网络（如Ethernet、Serial Line等）来传送数据。

3.TCP/UDP区别讲解

好吧，前两点侃侃而谈，只是给大家普及下OSI七层模型和TCP/IP四层模型的概念，接下来要讲的是和我们Socket开发相关的一些概念名词了！

1) IP地址

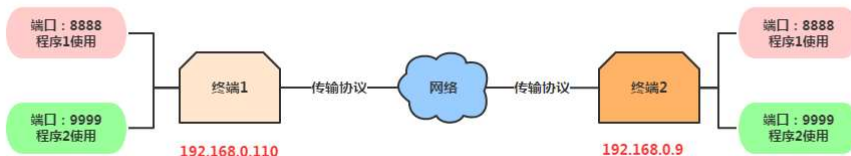
两台终端如何通过网络进行通信(IP地址)



为了实现网络中不同终端之间的通信，每个终端都必须有一个唯一的标识——IP地址

2) 端口

1. 用于区分不同的应用程序
2. 端口号的范围为0-65535，其中0-1023为系统的保留端口，我们的程序尽可能别使用这些端口！
3. IP地址和端口号组成了我们的Socket，Socket是网络运行程序间双向通信链路的终结点，是TCP和UDP的基础！
4. 常用协议使用的端口：HTTP:80，FTP：21，TELNET：23



3) TCP协议与UDP协议的比较：

TCP协议流程详解:

首先TCP/IP是一个协议簇，里面包括很多协议的。UDP只是其中的一个。之所以命名为TCP/IP协议，因为TCP,IP协议是两个很重要的协议，就用他两命名了。

4.4.2 ContentProvider再探...

4.5.1 Intent的基本使用

4.5.2 Intent之复杂数据的传递

5.1 Fragment基本概述

5.2.1 Fragment实例精讲—...

5.2.2 Fragment实例精讲—...

5.2.3 Fragment实例精讲—...

5.2.4 Fragment实例精讲—...

5.2.5 Fragment实例精讲—...

6.1 数据存储与访问之——文...

6.2 数据存储与访问之——S...

6.3.1 数据存储与访问之——...

6.3.2 数据存储与访问之——...

7.1.1 Android网络编程要学...

7.1.2 Android Http请求头与...

7.1.3 Android HTTP请求方...

7.1.4 Android HTTP请求方...

7.2.1 Android XML数据解析

7.2.2 Android JSON数据解析

7.3.1 Android 文件上传

7.3.2 Android 文件下载（1）

7.3.3 Android 文件下载（2）

7.4 Android 调用WebService

7.5.1 WebView(网页视图)基...

7.5.2 WebView和JavaScrip...

7.5.3 Android 4.4后WebVie...

7.5.4 WebView文件下载

7.5.5 WebView缓存问题

7.5.6 WebView处理网页返...

7.6.1 Socket学习网络基础准备

7.6.2 基于TCP协议的Socket...

7.6.3 基于TCP协议的Socket...

7.6.4 基于UDP协议的Socke...

8.1.1 Android中的13种Draw...

8.1.2 Android中的13种Draw...

8.1.3 Android中的13种Draw...

8.2.1 Bitmap(位图)全解析 P...

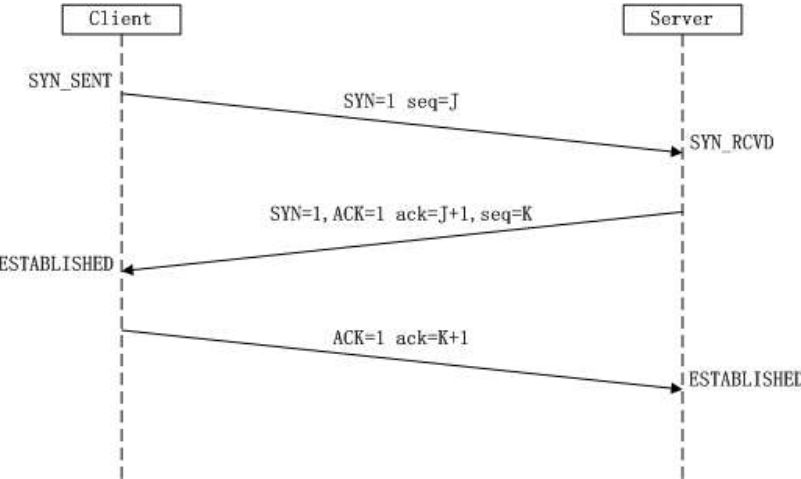
8.2.2 Bitmap引起的OOM问题

8.3.1 三个绘图工具类详解

8.3.2 绘图类实战示例

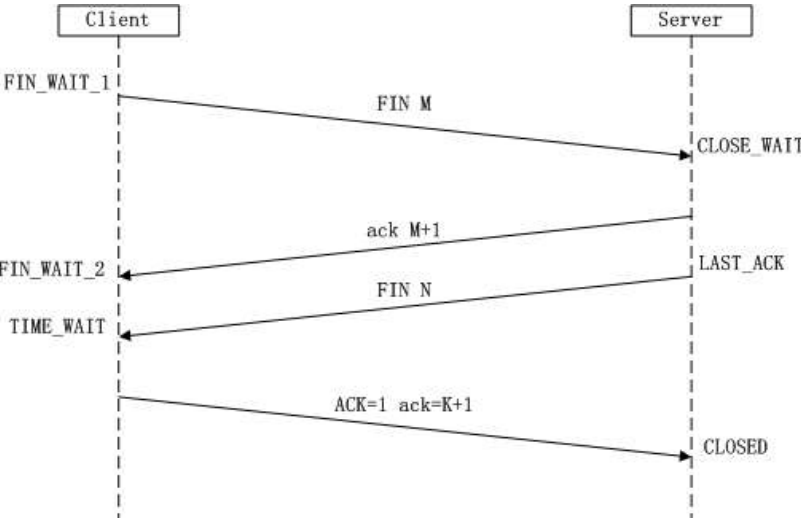
下面我们来讲解TCP协议和UDP协议的区别：

TCP (Transmission Control Protocol , 传输控制协议) 是面向连接的协议，即在收发数据钱 ，都需要与对面建立可靠的链接，这也是面试经常会问到的TCP的**三次握手**以及TCP的**四次挥手**！**三次握手**： 建立一个TCP连接时，需要客户端和服务端总共发送3个包以确认连接的建立， 在Socket编程中，这一过程由客户端执行connect来触发，具体流程图如下：



- 第一次握手**：Client将标志位SYN置为1，随机产生一个值seq=J，并将该数据包发送给Server， Client进入SYN_SENT状态，等待Server确认。
- 第二次握手**：Server收到数据包后由标志位SYN=1知道Client请求建立连接，Server将标志位 SYN和ACK都置为1，ack=J+1，随机产生一个值 seq=K，并将该数据包发送给Client以确认连接请求，Server进入 SYN_RCVD状态。
- 第三次握手**：Client收到确认后，检查ack是否为J+1，ACK是否为1，如果正确则将标志位ACK 置为1，ack=K+1，并将该数据包发送给Server，Server检查ack是否为K+1，ACK是否为1，如果正确则 连接建立成功，Client和Server进入ESTABLISHED状态，完成三次握手，随后Client与Server之间可以 开始传输数据了。

四次挥手： 终止TCP连接，就是指断开一个TCP连接时，需要客户端和服务端总共发送4个包以确认连接的断开。 在Socket编程中，这一过程由客户端或服务端任一方执行close来触发，具体流程图如下：



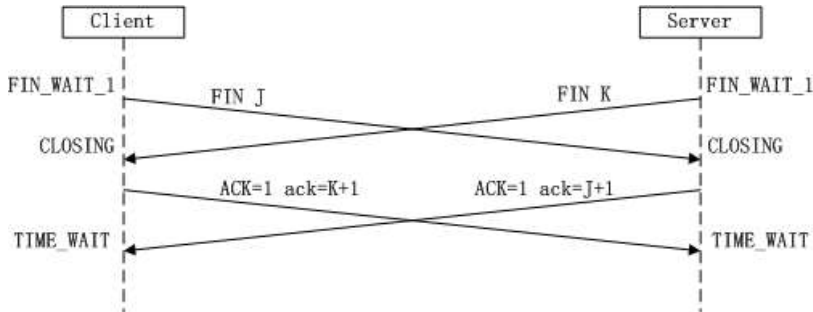
- 第一次挥手**：Client发送一个FIN，用来关闭Client到Server的数据传送，Client进入 FIN_WAIT_1状态
- 第二次挥手**：Server收到FIN后，发送一个ACK给Client，确认序号为收到序

- 8.3.3 Paint API之—— Mask...
- 8.3.4 Paint API之—— Xferm...
- 8.3.5 Paint API之—— Xferm...
- 8.3.6 Paint API之—— Xferm...
- 8.3.7 Paint API之—— Xferm...
- 8.3.8 Paint API之—— Xferm...
- 8.3.9 Paint API之—— Color...
- 8.3.10 Paint API之—— Colo...
- 8.3.11 Paint API之—— Colo...
- 8.3.12 Paint API之—— Path...
- 8.3.13 Paint API之—— Sha...
- 8.3.14 Paint几个枚举/常量值...
- 8.3.15 Paint API之——Type...
- 8.3.16 Canvas API详解(Part 1)
- 8.3.17 Canvas API详解(Part...
- 8.3.18 Canvas API详解(Part...
- 8.4.1 Android动画合集之帧...
- 8.4.2 Android动画合集之补...
- 8.4.3 Android动画合集之属...
- 8.4.4 Android动画合集之属...
- 9.1 使用SoundPool播放音...
- 9.2 MediaPlayer播放音频与...
- 9.3 使用Camera拍照
- 9.4 使用MediaRecord录音
- 10.1 TelephonyManager(电...
- 10.2 SmsManager(短信管理...
- 10.3 AudioManager(音频管...
- 10.4 Vibrator(振动器)
- 10.5 AlarmManager(闹钟服务)
- 10.6 PowerManager(电源服...
- 10.7 WindowManager(窗口...
- 10.8 LayoutInflater(布局服务)
- 10.9 WallpaperManager(壁...
- 10.10 传感器专题(1)——相...
- 10.11 传感器专题(2)——方...
- 10.12 传感器专题(3)——加...
- 10.12 传感器专题(4)——其...
- 10.14 Android GPS初涉
- 11.0 《2015最新Android基...

号+1（与SYN相同，一个FIN占用一个序号），Server进入CLOSE_WAIT状态。

第三次挥手：Server发送一个FIN，用来关闭Server到Client的数据传送，Server进入LAST_ACK 状态。

第四次挥手：Client收到FIN后，Client进入TIME_WAIT状态，接着发送一个ACK给Server，确认序号为收到序号+1，Server进入CLOSED状态，完成四次挥手。另外也可能是同事发起主动关闭的情况：



另外还可能有一个常见的问题就是：为什么建立连接是三次握手，而关闭连接却是四次挥手呢？答：因为服务端在LISTEN状态下，收到建立连接请求的SYN报文后，把ACK和SYN放在一个报文里发送给客户端。而关闭连接时，当收到对方的FIN报文时，仅仅表示对方不再发送数据了但是还能接收数据，己方也未必全部数据都发送给对方了，所以己方可以立即close，也可以发送一些数据给对方后，再发送FIN报文给对方来表示同意现在关闭连接，因此，己方ACK和FIN一般都会分开发送。

UDP协议详解：

UDP(User Datagram Protocol)用户数据报协议，非连接的协议，传输数据之前源端和终端不建立连接，当它想传送时就简单地去抓取来自应用程序的数据，并尽可能快地把它扔到网络上。在发送端，UDP传送数据的速度仅仅是受应用程序生成数据的速度、计算机的能力和传输带宽的限制；在接收端，UDP把每个消息段放在队列中，应用程序每次从队列中读一个消息段。相比TCP就是无需建立链接，结构简单，无法保证正确性，容易丢包

——上述内容部分摘自：

[TCP/IP三次握手与四次挥手](#)

[TCP和UDP的区别（转）](#)

4.Java中对于网络提供的几个关键类：

针对不同的网络通信层次，Java给我们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可以直接读取或者写入网络上的数据

Socket和ServerSocket：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

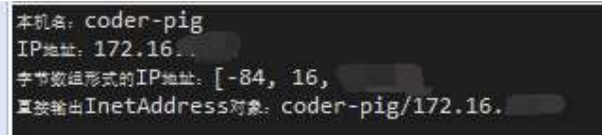
本节我们只介绍前两个类，Socket与Datagram到TCP和UDP的章节再讲解！

~InetAddress的使用例子：

示例代码：

```
public class InetAddressTest {
    public static void main(String[] args) throws Exception
    {
        //获取本机InetAddress的实例:
        InetAddress address = InetAddress.getLocalHost(
    );
        System.out.println("本机名: " + address.getHostN
ame());
        System.out.println("IP地址: " + address.getHost
Address());
        byte[] bytes = address.getAddress();
        System.out.println("字节数组形式的IP地址: " + Arr
ays.toString(bytes));
        System.out.println("直接输出InetAddress对象: " +
address);
    }
}
```

运行结果图：



~URL：这个就不用说了吧，忘了可以看会前面Http协议讲解那里~

本节小结：

本节全是概念，看起来可能够呛的是把，不过看不懂也没关系，知道七层模型每层叫 什么，大概拿来干嘛，还有TCP三次握手和四次挥手，就可以了！当然，这只是为了 应付面试~实际开发我们哪会纠结这个...直接Socket是吧~嗯，下节我们就来开始学习 Android中的Socket通信~谢谢~



VIPABC 免费请您看电影
体验外教课

立即领取



在线实例

- HTML 实例
- CSS 实例
- JavaScript 实

字符集&工具

- HTML 字符集设置
- HTML ASCII

最新更新

- 设置 SSH 通
过密...
- CSS all 属性

站点信息

- 意见反馈
- 免责声明
- 关于我们

	<p>例</p> <ul style="list-style-type: none">· Ajax 实例· jQuery 实例· XML 实例· Java 实例	<p>字符集</p> <ul style="list-style-type: none">· HTML ISO-8859-1· HTML 实体符号· HTML 拾色器	<ul style="list-style-type: none">· Px、Em 换算工具· px,pt,em换算表· px、em、rem 区别...	<ul style="list-style-type: none">· 文章归档
	<ul style="list-style-type: none">· JSON 格式化工具	<ul style="list-style-type: none">· Viewport 模板· 移动WEB前端开发...	<div><div>关注微信</div><div></div><div>Copyright © 2013-2015 菜鸟教程 runoob.com All Rights Reserved. 备案号：闽ICP备15012807号-1</div></div>	