

# 传智播客

## 《Python 程序开发案例教程》

### 教学设计

课程名称：Python 程序开发案例教程

授课年级：2019 年级

授课学期：2019 学年第一学期

教师姓名：某某老师

2019 年 09 月 09 日

课题名称	第 14 章 网络编程	计划学时	6 学时
内容分析	<p>当今社会是信息化社会，信息的传播离不开网络，随着计算机与因特网的普及和发展，网络已渗透到社会生活的各行各业，大到操作系统，小到手机应用，都与网络息息相关。</p>		
教学目标及基本要求	<ol style="list-style-type: none"> <li>1. 了解网络编程的基本概念</li> <li>2. 掌握 TCP 与 UDP 通信流程，熟练使用 socket 内置方法</li> <li>3. 掌握 TCP 并发服务器实现方式</li> <li>4. 熟悉 I/O 多线转接服务器的搭建方法</li> </ol>		
教学重点	<ol style="list-style-type: none"> <li>1. socket 套接字</li> <li>2. socket 内置方法</li> <li>3. 单进程非阻塞服务器</li> </ol>		
教学难点	<ol style="list-style-type: none"> <li>1. 基于 UDP 的网络聊天室</li> <li>2. 基于 TCP 的数据转换</li> <li>3. 多进程并发服务器</li> <li>4. 多线程并发服务器</li> <li>5. select 并发服务器</li> <li>6. epoll 并发服务器</li> </ol>		
教学方式	<p>教学采用教师课堂讲授为主，使用教学 PPT 讲解</p>		
教学过程	<p style="text-align: center;"><b>第一课时</b></p> <p style="text-align: center;">（网络概述、协议与体系结构、数据传输流程、网络架构、IP 地址和端口号）</p> <p><b>一、创设情境，引出文件什么是网络编程</b></p> <p>（1）教师通过提出需求，引出网络编程概述。</p> <p style="color: blue;">网络是网络编程的基础，网络编程的实质是两台设备中的进程通过网络机进行数据交换，即进程间的网络通信。</p>		

(2) 明确学习目标

- 要求学生了解协议与体系结构
- 要求学生了解数据传输流程
- 要求学生了解网络架构
- 要求学生了解 IP 地址和端口号

## 二、进行重点知识的讲解

(1) 教师根据课件，讲述协议与体系结构。

网络中存在多台主机，为保证主机间能顺利通讯，且通信双方可以获取到准确、有效的数据应制订一组用于数据传输的规则，这组规则就是协议。

制订协议时为网络间通信过程所划分的层次通常称为计算机网络的体系结构。

(2) 教师根据课件，讲述数据传输流程。

数据传输流程详见教材 14.1.2

(3) 教师根据课件，讲述网络架构。

网络架构分为 C/S 架构和 B/S 架构，其中 C/S 架构即客户机（client）/服务器（server）模式，这种架构需要在进行通信的两端分别架设客户机和服务器；B/S 架构是浏览器（browser）/服务器（server）架构，这是 WEB 兴起后的一种网络架构，客户机只需安装浏览器，便可与服务器进行交互。

(4) 教师根据课件，讲述 IP 地址和端口号。

IP 地址用于在网络上标记一台电脑；端口号是一台主机中进程的唯一标识，因此一个进程在向另一个进程发送数据时，要使用 ip 地址+端口号确定网络中的唯一进程。

## 三、归纳总结，布置作业/随堂练习

(1) 回顾上课前的学习目标，并对本节课的内容进行总结。

教师总结本节课需要掌握的知识点，包括什网络概述、协议与体系结构、数据传输流程、网络架构、IP 地址和端口号。

(2) 布置随堂练习，检查学生掌握情况。

根据博学谷和随堂练习资源，给学生布置随堂练习，检测学生的掌握程度，并对学生出现的问题进行解决。

(3) 使用博学谷系统下发课后作业。

## 第二课时

(**socket 网络编程基础、socket 套接字、socket 通信流程、socket 内置方法、实例 1：扫描开放端口**)

### 一、回顾上节课的内容，继续讲解本课时的知识

- (1) 教师对学生们的疑问进行统一答疑。
- (2) 回顾总结上节课内容，继续介绍本课时的内容。

上节课我们学习了网络概述、协议与体系结构、数据传输流程、网络架构、IP 地址和端口号，本节课将带领大家学习 socket 套接字、socket 通信流程、socket 内置方法、实例 1：扫描开放端口。

- (3) 明确学习目标
  - 要求学生掌握 socket 套接字
  - 要求学生掌握通信流程
  - 要求学生掌握 socket 内置方法
  - 要求学生了解实例 1：扫描开放端口

### 二、进行重点知识的讲解

- (1) 教师根据课件，讲解 socket 套接字。

socket 是进程间通信方式的一种，当使用 socket 进行通信时，进程会先生成一个 socket 文件，之后再通过 socket 文件进行数据传递。Python 有一个 socket 模块，该模块包含了网络编程的类、方法、函数等。

- (2) 教师根据课件，讲解 socket 通信流程。

根据 socket 的类型，网络通信又分为基于 TCP 协议、面向连接的通信和基于 UDP、面向无连的通信（面向连接的 socket 通信与面向非连接的 socket 通信详见图 14-7 与 14-8）。

- (3) 教师根据课件，讲解 socket 内置方法。

socket 模块中为 socket 对象定义了一些内置方法，通过这些内置方法，可以实现 socket 通信（常见的方法有 bind、listen、setblocking...详见表 14-1）。

- (4) 教师根据课件，讲解实例 1：扫描开放端口的实现过程。

教师根据教材资源实现实例 1：扫描开放端口并向学生讲解其实现过程。

### 三、归纳总结，布置作业

- (1) 回顾学习目标，总结本节课所学知识包括：socket 套接字、socket 通信

流程、socket 内置方法、实例 1：扫描开放端口。

(2) 布置随堂练习，检查学生掌握情况。

根据博学谷和随堂练习资源，给学生布置随堂练习，检测学生的掌握程度，并对学生出现的问题进行解决。

(3) 使用博学谷系统下发课后作业。

### 第三课时

#### (基于 UDP 的网络聊天室、基于 TCP 的数据转换、实例 2：TCP 文件下载、单进程非阻塞服务器)

##### 一、回顾上节课内容，继续介绍本节课的内容

(1) 教师对学生们的疑问进行统一答疑。

(2) 教师通过提问学生问题，由上一课时引出本课时要讲解的内容。

(3) 明确学习目标

- 要求学生掌握基于 UDP 的网络聊天室
- 要求学生掌握基于 TCP 的数据转换
- 要求学生了解实例 2：TCP 文件下载的实现过程
- 要求学生掌握单进程非阻塞服务器

##### 二、进行重点知识的讲解

(1) 教师根据课件，讲解基于 UDP 的网络聊天室。

教师根据教材讲解通过 UDP 实现的网络聊天室。

(2) 教师根据课件，讲解基于 TCP 的数据转换。

教师根据教材讲解通过 TCP 实现的数据转换。

(3) 教师根据课件，讲解实例 2:TCP 文件下载。

教师根据教材资源实现实例 2：TCP 文件下载并向学生讲解实现过程

(4) 教师根据课件，讲解单进程非阻塞服务器。

单进程非阻塞服务器通过解阻塞的方式实现并发操作。Python 中套接字默认以阻塞方式处理数据，若套接字调用 `accept()`、`recv()` 等方法时没有接收到数据，套接字就会阻塞等待数据递达。用户可通过套接字中提供的 `setblocking()` 方法将套接字设置为非阻塞模式，如此即便套接字中没有数据递达，套接字调用的方法也会立刻返回。

### 三、归纳总结，布置作业

- (1) 回顾学习目标，总结本节课所学知识包括：基于 UDP 的网络聊天室、基于 TCP 的数据转换、实例 2：TCP 文件下载、单进程非阻塞服务器。
- (2) 使用博学谷系统下发课后作业。

### 第四课时

#### (多进程并发服务器、多线程并发服务器、select 并发服务器、epoll 并发服务器)

#### 一、回顾上节课内容，继续介绍本课时的内容

- (1) 教师对学生们的疑问进行统一答疑。
- (2) 教师通过提问学生问题，由上一课时引出本课时要讲解的内容。
- (3) 明确学习目标
  - 要求学生掌握多进程并发服务器
  - 要求学生掌握多线程并发服务器
  - 要求学生了解 select 并发服务器
  - 要求学生了解 epoll 并发服务器

#### 二、进行重点知识的讲解

- (1) 教师根据课件，讲解多进程并发服务器。

多进程并发服务器中的主进程用于处理客户端的连接请求，当有新的客户端与服务器建立连接后，服务器会创建一个子进程，由子进程完成数据的交互工作。

- (2) 教师根据课件，讲解多线程并发服务器。

进程是系统分配资源的最小单位，每创建一个进程都会耗费一些系统资源，当进程较多时，服务器的效率会因系统内存的减少而降低，为解决这一问题，人们考虑使用耗费资源较少的线程代替进程，搭建多线程并发服务器。

- (3) 教师根据课件，讲解 select 并发服务器。

Python 中通过 select 模块实现 select 并发服务器，该模块包含了一个 select() 函数，其语法格式为 select(rlist,wlist,xlist[,timeout])，根据数据的类型，套接字分为三种状态，即可读状态、可写状态和异常状态。

- (4) 教师根据课件，讲解实 epoll 并发服务器。

	<p>epoll 服务是 linux 中常用的一种高效服务器,这种服务器采用事件通知机制,事先为要建立连接的 socket 注册事件,一旦该 socket 就绪,注册事件将被触发,socket 将被加入 epoll 的就绪套接字列表,而服务器无需主动监测所有套接字状态,只需直接获取就绪套接字列表,对其中的套接字进行处理即可。Python 中的 epoll 模式定义在 select 模块中,select 中包含了一个名为 epoll 的类,用户可先在程序中创建 epoll 对象,再通过 epoll 对象的方法实现 epoll 模式。</p> <p><b>三、归纳总结, 布置作业</b></p> <p>(1) 回顾学习目标,总结本节课所学知识包括:多进程并发服务器、多线程并发服务器、select 并发服务器、epoll 并发服务器。</p> <p>(2) 使用博学谷系统下发课后作业。</p> <p style="text-align: center;"><b>第五课时</b> <b>(上机练习)</b></p> <p>上机练习主要针对本章中需要重点掌握的知识点,以及在程序中容易出错的内容进行练习,通过上机练习可以考察同学对知识点的掌握情况,对代码的熟练程度。</p> <p><b>上机一: (练习教材示例代码以及实例 1: 扫描开放端口、实例 2: TCP: 文件下载)</b></p> <p><b>形式: 独立完成</b></p> <p><b>要求:</b></p> <p>(1) 要求学生能够熟练掌握教材中示例代码。</p> <p>(2) 要求学生能够自己实现实例 1 与实例 2 程序。</p>
<p>思考题 和习题</p>	<p>见教材第 14 章配套的习题</p>
<p>教 学 后 记</p>	