

Java - 10

web socket

这一板块我们主要学习网络编程。

简单挑战——现代网络基础

在java程序中，我们利用通道（channel）链接外面的世界，会建立客户端（client）通道，还会建立服务器（server）通道。我们会创建客户端和服务端，并让他们相互通信。

要让客户端正常工作，需要了解三个方面：

1. 如何建立客户端与服务器的初始连接。
2. 如何从服务器接受消息。
3. 如何向服务器发送消息

建立连接

要和另一个机器对话，我们需要一个对象表示这两个机器之间的一个网络连接。什么是连接呢？连接就是两个机器之间的一个关系，两个机器彼此知道对方并且知道如何向对方通信。

在java中，我们并不需要考虑底层细节，也就是不必考虑“网络栈”的底层实现，而是从高层考虑。要建立一个连接，需要知道服务器的两个信息：它在哪里（IP地址），它在哪个端口运行（端口号）。这里提供教程来了解一下什么是IP和TCP：[TCP/IP 介绍](#)

```
// 通过IP地址和TCP端口号创建一个对象并打开通道
InetSocketAddress serverAddress = new InetSocketAddress("196.164.1.103", 5000);
//这里是举例
SocketChannel socketChannel = SocketChannel.open(serverAddress);
```

接收信息

要在一个远程连接上通信，可以使用常规的I/O流，这得益于Java的特性——绝大多数I/O工作并不关心你的高层链接流到底链接哪里，你可以以操作文件的方式来操作网络Channel。

```
//建立一个与服务器的链接
SocketAddress serverAddr = new InetSocketAddress("127.0.0.1", 5000);
SocketChannel socketChannel = SocketChannel.open(serverAddr);
//从这个链接创建并获取一个Reader
Reader reader = Channels.newReader(socketChannel,
StandardCharsets.UTF_8); //reader是底层字节流和高层字符流的桥梁
//创建或获取一个BufferedReader并读取
BufferedReader bufferedReader = new BufferedReader(reader); //将BufferedReader串联到reader
String message = bufferedReader.readLine();
```

发送信息

在发送信息时，我们使用PrintWriter。这里的操作和接收信息基本类似。

```
//建立一个与服务器的链接
SocketAddress serverAddr = new InetSocketAddress("127.0.0.1", 5000);
SocketChannel socketChannel = SocketChannel.open(serverAddr);
//从这个链接创建并获取一个writer
Writer writer = Channels.newWriter(socketChannel, StandardCharsets.UTF_8);
//创建一个PrintWriter并打印一些内容
PrintWriter printWriter = new PrintWriter(writer);
writer.println("message to send");
writer.print("another message");
```

当然我们这里采用的是**Channel**的方式来进行网络编程，但其实我们还有许多的建立远程连接和读写远程服务器的不同方法，可以了解一下什么是**URL**以及如何运用**URL**：[Java URL](#)

任务：

改写上面的读写文件代码，采用**Socket**编程的方式进行远程连接的建立并读写远程服务器，你不需要实际运行该代码，因为你还没有学习如何建立一个属于自己的服务器。别担心，我们很快就会学习到的。。。吗？

tip: 你可能需要了解一下Socket的使用方法：[Java 实例 - ServerSocket 和 Socket 通信实例](#)

要求：将实现的代码推送到GitHub仓库上，此处提交链接：

出题人QQ：1727448271@qq.com

进阶挑战——？ ？ ？

你可能已经看到了胜利的曙光，现在我们还有进阶挑战吗？相信做到这里的同学们都是有能力的，按下这个完成按钮吧，你们已经做的足够棒了。也许考虑歇一歇，好好地睡一觉，毕竟你们已经完成了，不是吗？
