①TCP通信中，有Client与Server

②Client与Server之间的通信过程是复杂的，Java为了简化网络编程，提供了Socket(套接字)

③Socket屏蔽了传输层及以下的所有细节，让开发者可以将精力集中在应用层上

④我们使用Socket和ServerSocket两个套接字类——就是使用TCP传输协议

⑤我们通过Socket和ServerSocket来开发应用层，传输层及以下的细节，Socket帮我们弄好了，不用考虑

⑥本质上，两个应用进程之间的交互是通过报文进行的，但是在Socket的帮助下，在我们开发者看来，两个应用进程之间是通过 流(InputStream和OutputStream)来交互的

⑦所以Java网络编程，实质上是对Socket编程

在这里Socket相当于客户端，ServerSocket相当于服务端

一，ServerSocket类

1）构造方法

Public ServerSocket(int port)：创建一个ServerSocket，绑定到某个端口上

2）方法

①public Socket accept( )：

这是一个阻塞方法，ServerSocket开始监听所绑定的端口，如果一直没有Client向这个端口发送连接请求，那么程序就会阻塞在这里，直到有Client(Socket)向这个端口发送连接请求，然后accept( )会返回这个发起请求的Socket对象

②public int getLocalPort( )——获取ServerSocket绑定的端口号

二，Socket类

1）构造方法

①public Socket（String host，int port）

host——要连接的主机的IP地址

创建一个Socket对象，并**向指定主机的指定端口发送连接请求**（创建+请求）

如果host为null，那么该host默认为回送地址——本机的IP地址(127.0.0.1)

②public Socket（InetAddress host，int port）

效果如上，只是参数不同罢了

③public Socket(String host,int port, InetAddress localAddress,int localPort)

具有上面作用的同时，还指定了Socket所处的主机IP和端口

④public Socket(InetAddress host,int port，InetAddress localAddress，int localPort)

效果同③，参数不同罢了

2）普通方法

①public void connect(SocketAddress host,int timeout)

Socket向服务器发出连接请求，并指定一个超市值

②public InetAddress getInetAddress( )——返回Socket连接的 服务器的InetAddress

③public int getPort( )——返回Socket连接的 服务器的端口号

④public int getLocalPort( )——返回Socket所处的端口号

⑤public SocketAddress getRemoteSockerAddress( )

返回Socket所连接的服务器的地址，如果未连接，返回null

⑥public InputStream getInputStream( )——返回Socket的输入流

⑦public OutputStream getOutputStream( )——返回Socket的输出流

⑧public void close( )——关闭Socket

⑨public void shutdownOutput( )——关闭Socket的输出流

三，Socket与ServerSocket之间如何实现通信

!!!!!不管如何，一定是服务器端先启动，然后再启动客户端，

1）Client——>Server

客户端

①创建一个Socket，创建Socket的同时，Socket会向服务器的某个端口发出连接请求

②获取它的OutputStream

③利用OutputStream向外发送数据（可以用DataOutputStream包装一下）

④关闭OutputStream

服务器端

①创建一个ServerSocket

②调用ServerSocket的accept( )，获取请求连接的Socket对象（Client）

③获取Socket对象的InputStream，从中读取Client传来的数据（可以用DataInputStream包装）

④关闭InputStream

⑤关闭网络资源（Socket和ServerSocket）

2）Server——>Clent

情况跟上面大致相同

区别在于

服务器端——获取Socket对象的OutputStream，用于传送数据给Client

客户端——获取Socket对象的InputStream，用于获取服务器端传来的数据

四，代码实现

1）Client——>Server

2）Client与Server的双向通信

3）Client与Server的长时间通信

4）ServerSocket的多线程