第一题：谈谈Socket阻塞和非阻塞的理解

【答案】

套接字对象实际上是放在操作系统内核中。这个套接字对象内部有两个重要的缓冲结构，一个是读缓冲(read buffer)，一个是写缓冲(write buffer)，它们都是有限大小的数组结构。当我们对客户端的socket写入字节数组，是将字节数组拷贝到内核区套接字对象的write buffer中，内核网络模块会有单独的线程负责不停地将write buffer的数据拷贝到网卡硬件，网卡硬件再将数据送到网线，经过一些列路由器交换机，最终送达服务器的网卡硬件中。同样，服务器内核的网络模块也会有单独的线程不停地将收到的数据拷贝到套接字的read buffer中等待用户层来读取。最终服务器的用户进程通过socket引用的read方法将read buffer中的数据拷贝到用户程序内存中进行反序列化成请求对象进行处理。

write buffer空间都是有限的，所以如果应用程序往套接字里写的太快，这个空间是会满的。一旦满了，写操作就会阻塞，直到这个空间有足够的位置腾出来。不过有了NIO(非阻塞IO)，写操作也可以不阻塞，能写多少是多少，通过返回值来确定到底写进去多少，那些没有写进去的内容用户程序会缓存起来，后续会继续重试写入。

同样read buffer的内容可能会是空的。这样套接字的读操作(一般是读一个定长的字节数组)也会阻塞，直到read buffer中有了足够的内容(填充满字节数组)才会返回。有了NIO，就可以有多少读多少，无须阻塞了。读不够的，后续会继续尝试读取。