Java NIO学习，学习的记录过程。

## 前言

NIO就是new IO是对与传统IO的替换，这意味着NIO会对传统的Java IO与传统IO的API都会进行替换，主要的组件有：

* 通道与缓冲器：在标准的IOAPI中，你是使用的字节流或者字符流；在NIO中，你需要使用的是通道与缓冲器；数据总是从一个通道读入到缓冲器，或从一个缓冲期写入到通道中；
* 非阻塞IO：NIO是一种非阻塞IO，例如，一个线程可以请求从通道读数据到缓冲器，当读取时，线程请求后就可以去做其他事情了，一旦数据被读完，线程还可以返回来处理它们；
* 选择器：选择器是一个对象，可以监视多个通道发出的事件；比如打开连接、数据到达等；

## 概述

NIO的核心的API有Channel、Buffer、Selector；Channel有点像流；主要流的实现有：FileChannel、DatagramChannel、SocketChannel、ServerSocketChannel；

NIO的BUFFER实现有：ByteBuffer、CharBuffer等；

Selector允许单个线程处理多个Channel。

## Channel

NIO通道与流类似；Channel是双向的，流是单向的，Channel可以异步的读写；读写数据首先要有一个Buffer。

FileChannel：读写文件数据，DatagramChannel：通过UDP读写数据；SocketChannel：通过TCP读写数据；ServerSocketChannel：监听新进来的TCP连接；像Web服务器那样。每一个新进来的连接都会创建一个SocketChannel。

## Buffer

缓冲区是一块可以写入数据，可以读取数据的内存，这块内存被包装成NIO对象，并提供了一组方法。Buffer读写数据的步骤如下：

1.写入数据到Buffer；

2.调用flip()方法；

3.从Buffer中读取数据；

4.调用clear()或者compact()方法；

需要记住capacity、position、limit这3个值在Buffer不同模式下的含义。还有调用flip()方法后，各个值的含义。rewind()方法将position设为0，可以从新读取Buffer里面的值；clear()与compact()方法晴空Buffer区；mark()与reset()设定一个标记并且后面可以重置回标记；equals()与compareTo()比较元素。

分散（scatter），分散（scatter）注意看下定义。

还需要记住的FileChannel的transferFrom()方法与transferTo()方法。

Buffer的类型主要有：ByteBuffer、MappedByteBuffer、CharBuffer、DoubleBuffer、FloatBuffer、IntBuffer、LongBuffer、ShortBuffer。