1：官方文档

<https://github.com/dominictarr/stream-combiner>

2：官方文档说明

作用：把一个pipeline转成一个single stream.

Combine函数返回一个stream，这个stream是从第一个stream,直达最后一个stream。

duplex.js

duplex (writeStream, readStream)

把一个可读流和一个可写流转成一个可读可写流。

stream-combiner.js

Combine (stream1,...,streamN)

返回一个single流，第一个流是可写的，最后一个流是可读的。

The `stream-combiner` module creates a pipeline from a list of streams,

returning a single stream that exposes the first stream as the writable side and

the last stream as the readable side

var stream = combine(a, b, c, d);

will internally do `a.pipe(b).pipe(c).pipe(d)` but the `stream` returned by

`combine()` has its writable side hooked into `a` and its readable side hooked

into `d`.

through2.js

作用：  
备注：

疑问：为什么我不使用Node的核心模块”stream

在Node10之前的对于stream，stream core只提供了一个pipe().如果要写自定义的stream是十分的困难的，因为要进行状态管理。

在Node10之后，开发人员只需要对于一个抽象方法进行实现就好了，其他的就不用管了。

但是我在发布一个npm包的时候？  
我想兼容Node10以下的版本，但是我想使用NODE12提供的方法，怎么破？

参考文献：

<http://r.va.gg/2014/06/why-i-dont-use-nodes-core-stream-module.html>