# 实验15：Node.js流应用（Stream）

实验背景

利用已掌握的知识，完成实例。如图所示：

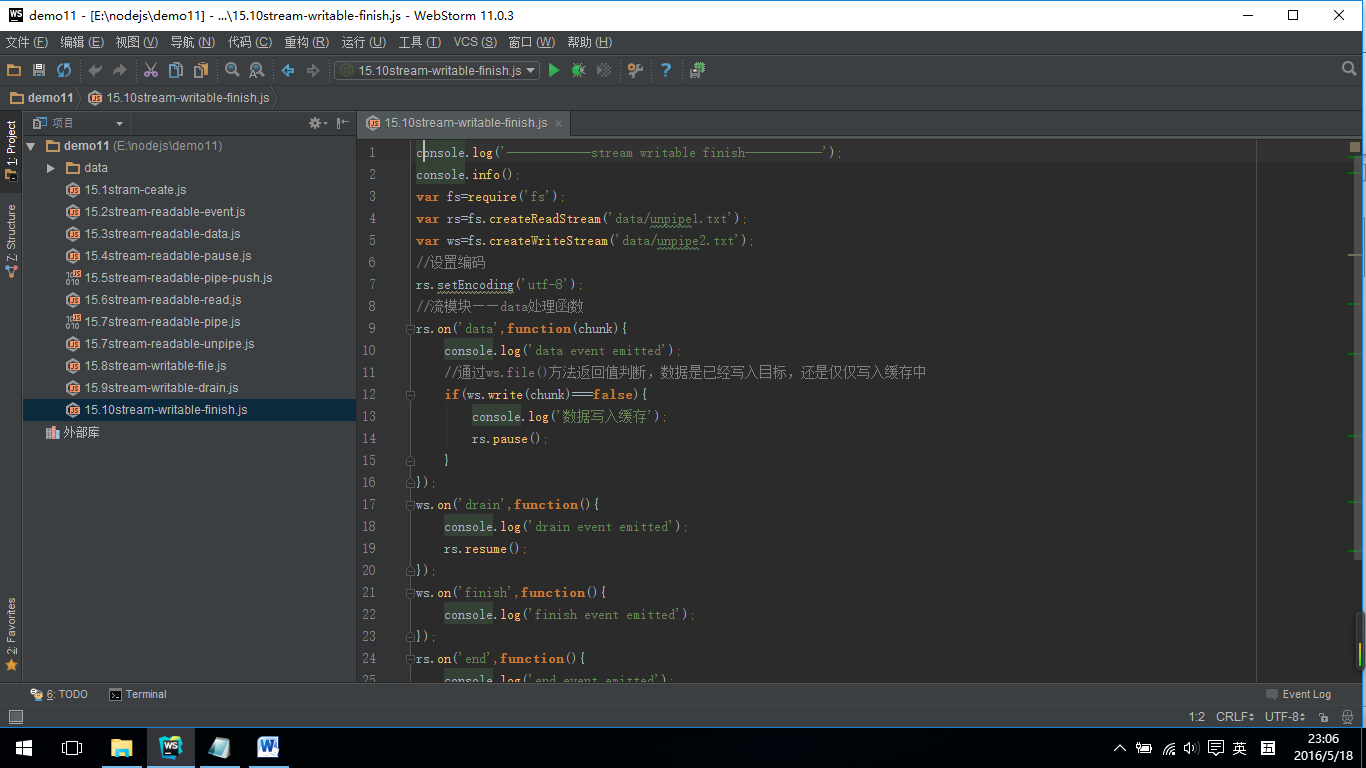


图1-1

实验目的

1、掌握Node.js Stream(流)

2、掌握管道流

3、掌握链式流

实验分析

本次实验要完成的有三个方面的知识。第一：完成实例《15.1》，第二：完成实例《15.7》，第三：完成实例《15.9》。

实验步骤

1. 实例《15.1》

console.log('------------stream create read stream------------');

console.info();

var fs=require('fs');

var rs=fs.createReadStream('data/basic.txt');

//设置编码

rs.setEncoding('utf-8');

//流模块——readable处理函数

rs.on('readable',function(){

console.log('readable event emitted.');

console.info();

});

//流模块——data处理函数

rs.on('data',function(chunk){

console.log('data event emitted.');

console.info();

});

//流模块——error处理函数

rs.on('error',function(){

console.log('error event emitted.');

console.info();

});

//流模块——end处理函数

rs.on('end',function(){

console.log('end event emitted.');

console.info();

});

//流模块——end处理函数

rs.on('close',function(){

console.log('close event emitted.');

console.info();

});

1. 实例《15.7》

console.log('------------stream readable pipe-----------');  
console.info();  
**var** fs=require('fs');  
**var** readable=fs.createReadStream('data/basic.txt');  
**var** writable=fs.createWriteStream('data/write.txt');  
readable.pipe(writable);

步骤3 实例《15.9》

console.log('------------stream writable file-----------');

console.info();

var fs=require('fs');

var rs=fs.createReadStream('data/unpipe1.txt');

var ws=fs.createWriteStream('data/unpipe2.txt');

//设置编码

rs.setEncoding('utf-8');

//流模块——data处理函数

rs.on('data',function(chunk){

console.log('data event emitted');

//通过ws.file()方法返回值判断，数据是已经写入目标，还是仅仅写入缓存中

if(ws.write(chunk)===false){

console.log('数据写入缓存');

rs.pause();

}

});

ws.on('drain',function(){

console.log('drain event emitted');

rs.resume();

});

rs.on('end',function(){

console.log('end event emitted');

ws.end();

});

});扩展练习

1.思考并实现，完成案例《实例15.10》并实现测试

实验思考

1.思考并实现，独立设计一个小应用。