## newman简介

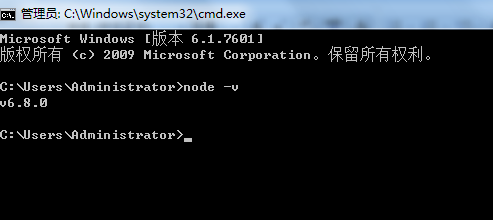
newman 是 postman 推出的一个 nodejs 库，直接来说就是 postman 的json文件可以在命令行执行的插件。Newman 可以方便地运行和测试集合，并用之构造接口自动化测试和持续集成。支持Windows、Mac、Linux平台。

#### 一、newman安装（Windows环境）

###### 1.安装node.js

先下载安装node.js。

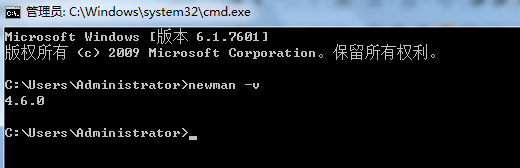
在dos窗口中输入 ：node -v 查看，如下图所示表示安装成功。



###### 2.安装newman

在dos窗口执行命令： npm -g install newman

安装完成后，输入newman -v，下图所示表示安装成功。



（npm 是 JavaScript 的包管理工具，并且是 node.js 平台的默认包管理工具。通过 npm 可以安装、共享、分发代码，管理项目依赖关系。一般安装好node.js后会默认安装好npm的，直接使用即可。）

###### 3.安装newman-reporter-html

如果想生成html格式的测试报告，还需要安装newman-reporter-html。

在dos窗口中执行：npm install -g newman-reporter-html

## 二、命令行执行脚本

newman执行脚本用法：newman run <collection> [options]

collection代表postman导出的json脚本。

options指定选项。

执行newman run -h 可查看newman的相关命令。

以下是常用的options:

-e, --environment <path> 指定环境变量文件的路径

-g, --globals <path> 指定全局变量文件的路径

--folder <path> 执行collections中folder的用例。可以多次指定运行多个文件夹

-n, --iteration-count <n> 要运行的迭代次数

-d, --iteration-data <path> 用于迭代的数据文件（json或csv），即外部文件参数

--timeout-request [n] 指定请求的超时（以毫秒为单位）

-r, --reporters [reporters] 生成报告（默认cli格式，dos控制台输出）

简单举例（在dos输出，输出到文件见三）：

1.脚本中没有环境变量和全局变量的情况时，可以直接运行json脚本。

newman run 脚本路径

脚本路径需指定到脚本名称（如d:\ test.postman\_collection.json）

2.脚本中有环境变量或者全局变量时，需要指定变量的路径。

newman run -e <环境变量路径> -g <全局变量路径> 脚本路径

3.脚本中有外部文件变量时，需要指定外部文件的路径

newman run -e <环境变量路径> -g <全局变量路径> -d <外部文件路径> 脚本路径

4.执行collections中folder的脚本。

newman run -e <环境变量路径> -g <全局变量路径> 脚本路径 -- folder 文件夹1名称 --folder 文件夹2名称 (根据需要可同时指定collection下的多个folder)

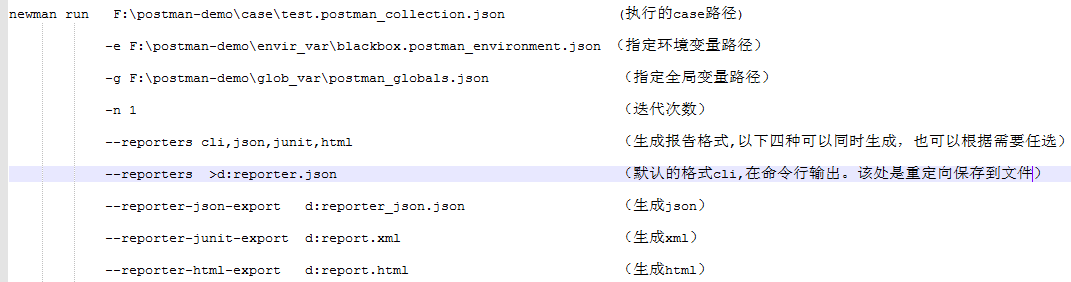
## 三、输出测试报告文件

newman 可生成cli，json，html，junit四种格式的报告，并指定路径、名称。

输出报告时使用的命令：

　　-reporters html,json,junit         指定生成html，json，xml形式的测试报告  
　　　　--reporter-json-export jsonReport.json          生成json格式的测试报告  
　　　　--reporter-junit-export xmlReport.xml            生成xml格式的测试报告  
　　　　--reporter-html-export htmlReport.html          生成html格式的测试报告

下面是对以上命令执行，同时生成多个格式报告的命令解释：



## 四、newman执行生成日志文件

为了方便定位接口执行的结果，需要提供日志信息。可在Postman接口中添加console.log()、console.info()、console.warn()、console.error()来生成不同级别的日志信息， 在Newman中执行下面命令生成日志文件。

newman run <脚本路径> -e <环境变量路径> > d:/result.log（根据需要指定）