今日目标：

1. NodeJS概述
2. NodeJS特点——掌握
3. NodeJS常用模块——重点
4. NodeJS搭建WEB服务器——重点
5. NodeJS连接MySQL服务器——重点

使用NodeJS代替 Apache+PHP

1.NodeJS概述

官网：https://nodejs.org/en/

NodeJS是一个ECMAScript的解释器，基于V8引擎，用于服务器端JS应用程序。

v4.4.7 LTS Long Term Support

NodeJS 和 V8的区别：

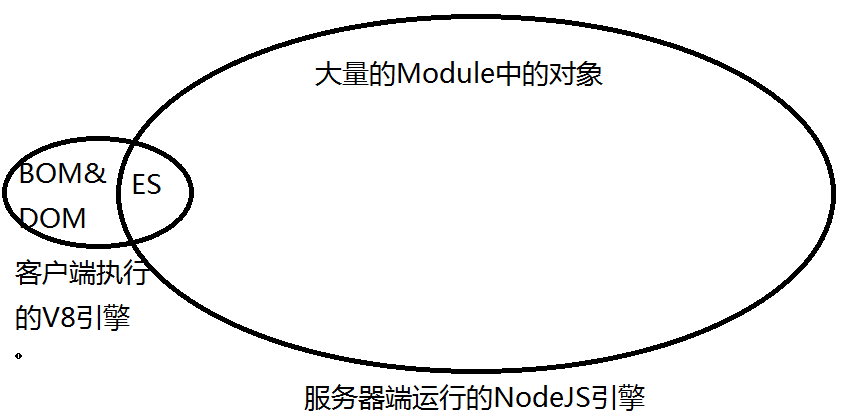
JavaScript中对象的分类：

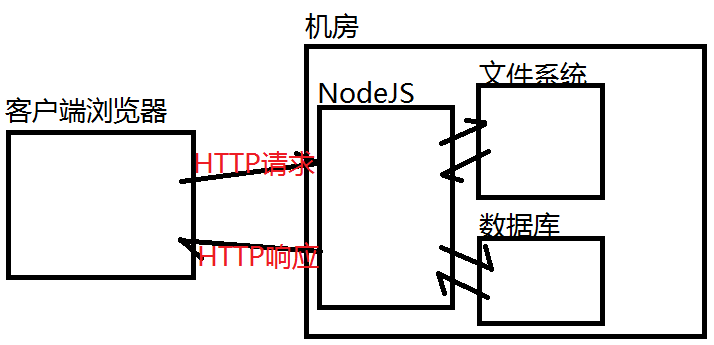
1. ES原生对象： String、Date、Math、RegExp、Array、Error、Function、Object
2. BOM&DOM对象： Window、Document、Navigator.....
3. 自定义对象： Emp、Student...

V8引擎支持上述三种对象

NodeJS中没有BOM&DOM对象，支持：(1)ES原生对象 (2)大量的第三方对象(模块)

|  |
| --- |
| NodeJS中的模块：module，一个箱子，用于封装一批的相关的对象 |





NodeJS的意义：

1. 执行效率比PHP/JSP/ASPX都要高
2. 使用一种语言统一了前后端开发——ES

2.Node.js的两种使用方法

(1)交互模式——用于测试

输入一行js语句，回车执行。

输出一个变量的值，无需使用console.log()，直接写变量名+回车。

(2)脚本模式——用于开发

把要执行的js语句编写在一个文本文件中，一次性提交给NodeJS解释器执行

c:/Users/Administrator> e: 切换的E盘

e:> cd mynode 进入指定的文件夹

e:/mynode> node xx.js 解释指定的JS文件

3.NodeJS中常用的模块

**(1)url模块——用于处理URL字符串**

var url = require('url');

url.parse( 'URL' ); 解析出一个URL中的各个部分

**(2)fs模块——用于读写文件系统中的文件内容**

var fs = require('fs');

fs.readFile('文件路径和名称', function(err, data){

//data中暂存着读取到的文件内容

})

**(3)http模块——用于接收客户端请求，给出HTTP响应——编写WEB服务器**

var http = require('http');

var server = http.createServer();

server.on('request', function(request, response){ });

server.listen(8000, function(err){ })

**(4)mysql模块——用于连接MySQL数据库服务器**

提示：NodeJS官方没有连接MySQL必需的函数或对象

第三方NodeJS函数库——NPM(Node Package Manager)

https://www.npmjs.com/

npm：是一个包管理工具，用于为NodeJS下载更多的函数模块。

下载mysql模块的方式： **npm install mysql**

第三方模块必需在当前目录下的node\_modules目录下。

var mysql = require('mysql');

var conn = mysql.createConnection({

host : '127.0.0.1',

user : 'root',

password : '',

database : 'jd'

});

conn.query('SQL...', function(err, data){

console.log('SQL执行完成');

conn.end();

})