今天学习的内容

1:复习上周的内容

2:作业:海葵/英特尔首页/学子商城商品列表

3:今天的目标

3.1:node.js 基本概念/模块

3.2:阿里面试题:用户在浏览器中输入 www.taobao.com 直到看到

网页之间发生了什么?

(1)操作系统(window/linux/macos)访问网络上DNS服务器，把域名转换IP地址

(2)浏览器通过ip向淘宝服务器发送HTTP请求

(3)淘宝服务器接收请求，解析请求，查找资源,

可能访数据库,构建响应消息，发送浏览器中

(4)浏览器接收并解析响应消息

(5)浏览器缓存接收响应主体，解析和渲染响应内容

3.3静态网页和动态网页

静态网页:网页内容任何人在任何时间访问都是不变的

HTML/CSS/JS/Flash/视频/音频/....

动态网页:网页内容不同人在不同时间访问可能是不同

JSP/PHP/ASP/Node.JS

JSP = html+java 功能强大适合大型或超大型项目{卡车}

2003=1个月[php+mysql];5w;50w-->mysql->oracle

500w->php->java[..]->1000w->服务器->小型机

oracle+小型机

集群+java+淘宝(mysql)-> 1亿/1day

PHP = 简单易用;适合互联网项目[论坛] {家用轿车}

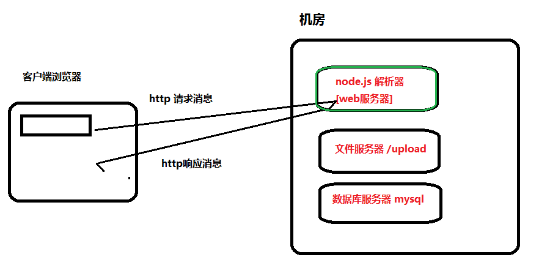
ASP = 微软产品,适全 windows  $$$$ {家用轿车} 300w

数据库 sql server 20w +开发工具+服务器

Node.js = 性能好,适合服务端IO密集型项目，不适合CPU

密集型项目 首页荐node.js {911}

|  |  |
| --- | --- |
| CPU密集型项目 | IO密集型 |
| 滴滴打车;高德导航;天气预报;石油预测分布 | I 输入 O 输出 数据获取，数据输出 |



3.3Node.JS

Node.js不是JS。是一种运行于**服务器端开发平台**，它的开发语言使用 js ,历史上第一个可以通吃前后台语言.

官方网站: www.nodejs.org

淘宝镜像: https://npm.taobao.org/mirrors/node

版本:0.12(16年初) 4.x(16年中)6.x(16年底)

node-v8.10.0-x64.msi windows下64位

node-v8.10.0-x86.msi windows 下32位

node-v8.10.0.pkg 苹果电脑

3.4Node.JS 两种运行模块

(1)交互模式---一般用于临时测试

node 回车

(2)脚本模式--正式项目中使用方式

把要把的要执行语句编写一个文本文件(后缀任意，没有可行)

一次性提交node解释器执行

node x.js 回车

提示:只要安装完node.js 重启一下webstorm，可以自动发现

node.exe 解释程序,新建项目编号一定utf-8

File->Settings->Editor->File Encoding->

Global Encoding->UTF-8

Project Encoding->UTF-8

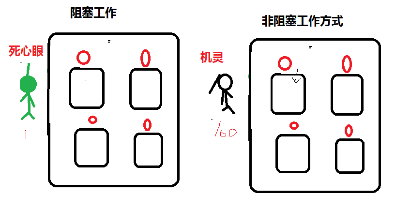
练习:创建02.js 打印九九乘法表, 以脚本模式来运行

3.5 Node.JS 如何自学一门语言-- (node.js)

(1)了解背景

--百度百科 node.js

[Node.js 使用事件驱动， 非阻塞I/O 模型]



(2)搭建开发环境:编写Hello World

(3)数据类型-----重点

|  |
| --- |
| 前端JS数据类型  (1)原始类型  string/number/boolean/undefined/null  (2)引用类型  ES对象类型:Date/RegExp/Object/Error/...  DOM/BOM:window/document/location/event/....  用户自定义:{} |
| 后端Node.js 数据类型  (1) 原始类型  string/number/boolean/undefined/null  (2) 引用类型  ES对象类型:Date/RegExp/Object/Error/...  用户自定义:{}  (3)Node.js 第三方对象 |

(4)变量和常量

var age = 20;

const PI = 3.14;

(5)运算符

算术运算符

比较运算符

逻辑运算符

三目运算符

位运算符

赋值运算符

特效运算符 instanceof typeof .

(6)逻辑结构

循环结构:for(;;);do{}while();while(){};for(in);for(..of..)

练习:03.js 声明一个保存5人学生成绩的数组,

使用三种for循环依次打印出每个成绩值

选择结构:if...else; switch...case

练习04.js 创建一个变量 var path = "/index"

使用二种选择结构,判断path的值为哪种(/index;/search;/login)

调用不同的执行函数.

(7)通用小程序 九九乘法表 100以质数 数组排序

练习:05.js 打印100以所有质数

(8)函数和对象

(9)常用的组件，第三方工具，框架

(10)实际小项目

|  |
| --- |
| Modal 模态框  Model 模型  Module 模块 |

3.5:Node.JS 中特有概念--模块

一个web项目中可以分为多个不同"模块", 用户模块，订单模块 ，支付模块...

Node.js按照不同功能，将函数，对象保存在不同文件或者目录

这些文件或目录称 "模块"

Node.js中每个模块都是一个独立构造函数,解析器为每个.js

添加下代码.

(function(exports,require,module,\_\_dirname,\_\_filename){

exports:{} 用于声明向外部导出自己成员

require:fn 用于导入其它模块，创建指定模块对象

module:{} 指代当前模块对象!

\_\_dirname 二个下划线 当前模块所有属性目录绝对路径

\_\_filename 当前模块绝对路径

//自己编写代码

});

语法:

exports.属性名=值; #导出指定数据

var m = require("./模块名称"); #导入指定模块并创建对象

练习:08.js 创建三个变量(uname/age/ctime)一个函数(fnRun)导出

09\_app.js 引入08.js 模块并且调用三个变量和一个函数

|  |
| --- |
| Node.js 模块中exports和module.exports的区别是什么?  二者都可以用于向外界导出自己的内部成员,但  module变量指代当前模块对象，真导出是 module.exports  node.js 底层 exports = module.exports  所以:若只是给exports添加新成员，则等价于module.exports添加新成员，但若修改了exports指向，则不会产生实质作用. |
| exports.age = 10; 等同于 module.exports.age = 10;  成功导出成员 |
| exports = {uname:"tom"} 导出失败  **module.exports** = {uname:"jerry"} 成功导出对象 |

练习:创建一个文件模块 10\_circle.js 其中定义常量PI，声明一个

方法 getSize(r) ,根据传入圆形的半径，返回其面积,

以及 getPerimiter(r); 返回其周长，再创建一个应用程序

11\_app.js 引入 10\_circle.js 模块，调用其公开二个成员方法

作业1：今写NODE.JS提供一个模块:URL

创建模块文件 myURL,向外导出一个方法resolve(url)

该方法接收的参数形式如下:

http://www.jd.com:443/ad/index?uname=tom&pno=3

返回对象:形如

{

protocol:"http",

servername:"www.jd.com",

port:443,

pathname:"/ad/index",

query:{uname:"tom",pno:3}

}