# 高级课阶段第1次笔试题

**笔试目的**：锻炼大家的笔试能力，不会让你在笔试中出现提笔忘字的情况出现。同时也是对自己学习成果的一次检验，掌握本阶段最核心最精华的知识点。

1. 问题：console中除了console.log以外，有哪些常用方法

本题目的：

考察学员对Node.js核心库中的console掌握情况

答出1点2分，5个及满分10分

答案：

（1）console.info

（2）console.error

（3）console.warn

（4）console.time

（5）console.timeEnd

（6）console.assert

（7）console.trace

1. 问题：为什么要用node?

本题目的：

1. 让学员了解什么是Node.js 3分
2. Node.js 的特点/特性是什么 3分
3. 对每条特性展开讲解 4分

答案：

1. Node.js是构建于Chorme V8引擎基础之上的 JavaScript 运行时环境

2. 特性: 轻量高效，事件驱动、非阻塞I/O，强大的包管理工具(npm)

3. 非阻塞IO,可以适应分块传输数据，较慢的网络环境，尤其擅长高并发访问;

轻量体现在node本身既是代码，又是服务器，前后端使用统一语言;

可扩展体现在可以轻松应对多实例，多服务器架构，同时有海量的第三方应用组件

1. 问题：Node.js中包含哪些全局对象

本题目的：

考察学员对Global的掌握

答出1点2分，5点满分10分

答案：

\_\_dirname 文件夹名

\_\_filename 文件名

require 引入

exports 导出

module 模块

global 全局对象

1. 问题：node 包含哪些核心模块

本题目的：考察学员对Core-Modules 的了解程度

写出1个 模块 – 解释 得 2分， 写出5个及以上10分

如: 写出 fs - 文件系统 2分，只写出模块名/只写出解释，得1分

答案：

fs 文件系统

http 网络

module 模块

events 事件

path 路径

console 控制台输出模块

querystring 解析

buffer 二进制流

assert 断言

1. 问题：代码实现一个简单的http服务器

本题目的：http是node的实现服务器的基础，必须掌握服务器创建

引入模块 3分

创建服务器4分

监听 3分

答案：

var http = require('http');

http.createServer(function(req, res) {

res.end(‘Hello, Node.js’);

}).listen(3000);

1. 问题： node有哪些定时功能

本题目的：

考察学员对Node.js定时功能的掌握情况

答案：

setTimeout/clearTimeout, 3分

setInterval/clearInterval, 3分

setImmediate/clearImmediate, 3分

process.nextTick 1分

1. 问题：如何通过node.js读取json配置文件

本题目的：

考察学员对require以及文件系统的理解。

答案：

主要有两种方式:

第一种是利用node内置的require('data.json')机制，直接得到js对象; 2分

第二种是读入文件入内容，然后用JSON.parse(content)转换成js对象．2分

二者的区别是require机制情况下，如果多个模块都加载了同一个json文件，那么其中一个改变了js对象，其它跟着改变，这是由node模块的缓存机制造成的，只有一个js模块对象; 3分

第二种方式则可以随意改变加载后的js变量，而且各模块互不影响，因为他们都是独立的，是多个js对象 3分

1. 问题：读写一个文件有多少种方法

本题目的：

考察学员对fs的掌握情况

答案：

总体来说有四种:

1) POSIX式底层读写 3分

2) 流式读写 1分

3) 同步文件读写 3分

4) 异步文件读写 3分

1. 问题：express项目的目录大致是什么样子的？

本题目的：

熟悉 express 框架的组成

以下答案 1点2分，写出5点以上满分10分

答案：

app.js,

package.json,

/bin/www,

/public,

/routes,

/views.

1. 问题：express response有哪些常用方法？

本题目的：

考察学员对express的掌握

以下答案 1点2分，写出5点以上满分10分

答案：

（1）res.end() 结束response

（2）res.json() 返回json

（3）res.jsonp() 返回jsonp

（4）res.redirect() 重定向请求

（5）res.render() 渲染模板

（6）res.send() 返回多种形式数据

（7）res.sendFile 返回文件

（8）res.sendStatus() 返回状态