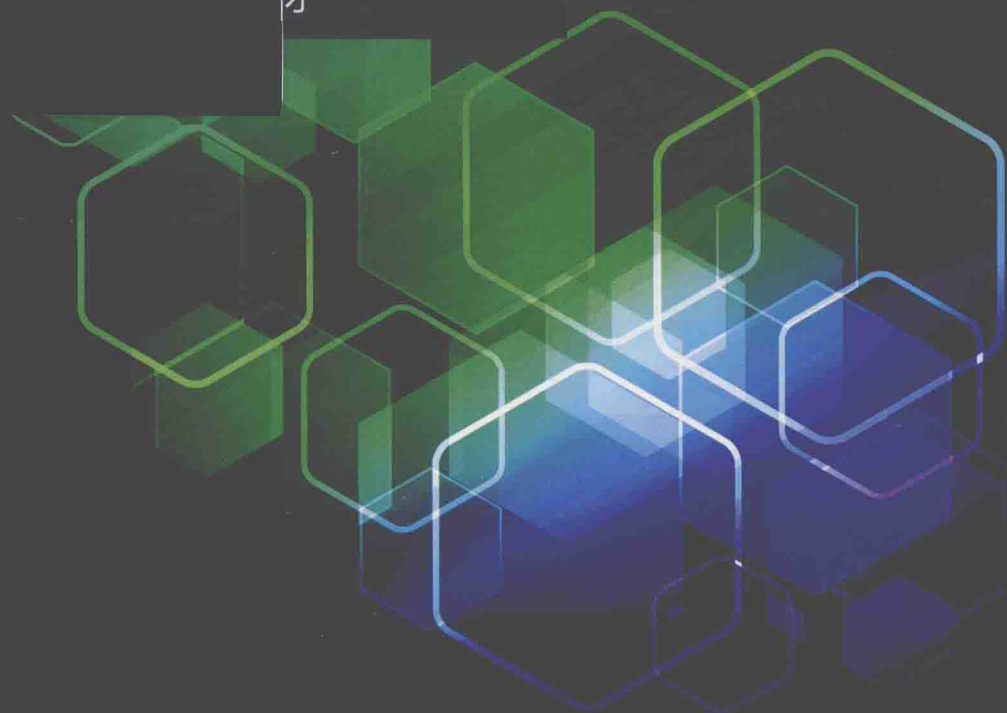


企业级卓越人才培养解决方案

材



# Node JS<sup>®</sup>

## 项目实战

天津滨海迅腾科技集团有限公司 主编

南开大学出版社

卓越人才培养解决方案“十三五”规划教材

# Node.js 项目实战

天津滨海迅腾科技集团有限公司 主编

南开大学出版社

天津

图书在版编目 (CIP) 数据

Node.js 项目实战 / 天津滨海迅腾科技集团有限公司  
主编. — 天津: 南开大学出版社, 2018.8  
ISBN 978-7-310-05643-9

I. ①N… II. ①天… III. ①JAVA语言—程序设计  
IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2018) 第 187136 号

主 编 牛永钢 康 健 陈桂芳  
副主编: 周青政 雷长虹 黄祥书 田 钰 陈观爱

**版权所有 侵权必究**

南开大学出版社出版发行

出版人: 刘运峰

地址: 天津市南开区卫津路 94 号 邮政编码: 300071

营销部电话: (022)23508339 23500755

营销部传真: (022)23508542 邮购部电话: (022)23502200

天津午阳印刷有限公司印刷

全国各地新华书店经销

2018 年 8 月第 1 版 2018 年 8 月第 1 次印刷

260×185 毫米 16 开本 19.5 印张 450 千字

定价: 68.00 元

如遇图书印装质量问题, 请与本社营销部联系调换, 电话: (022)23507125



# 企业级卓越人才培养解决方案“十三五”规划教材 编写委员会

- 指导专家**：
- |     |                  |
|-----|------------------|
| 周凤华 | 教育部职业技术教育中心研究所   |
| 李 伟 | 中国科学院计算技术研究所     |
| 张齐勋 | 北京大学             |
| 朱耀庭 | 南开大学             |
| 潘海生 | 天津大学             |
| 董永峰 | 河北工业大学           |
| 邓 蓓 | 天津中德应用技术大学       |
| 许世杰 | 中国职业技术教育网        |
| 郭红旗 | 天津软件行业协会         |
| 周 鹏 | 天津市工业和信息化委员会教育中心 |
| 邵荣强 | 天津滨海迅腾科技集团有限公司   |
- 主任委员**：王新强 天津中德应用技术大学
- 副主任委员**：
- |     |              |
|-----|--------------|
| 张景强 | 天津职业大学       |
| 宋国庆 | 天津电子信息职业技术学院 |
| 闫 坤 | 天津机电职业技术学院   |
| 刘 胜 | 天津城市职业学院     |
| 郭社军 | 河北交通职业技术学院   |
| 刘少坤 | 河北工业职业技术学院   |
| 麻士琦 | 衡水职业技术学院     |
| 尹立云 | 宣化科技职业学院     |
| 王 江 | 唐山职业技术学院     |
| 廉新宇 | 唐山工业职业技术学院   |
| 张 捷 | 唐山科技职业技术学院   |
| 杜树宇 | 山东铝业职业学院     |
| 张 晖 | 山东药品食品职业学院   |
| 梁菊红 | 山东轻工职业学院     |
| 赵红军 | 山东工业职业学院     |
| 祝瑞玲 | 山东传媒职业学院     |

王建国 烟台黄金职业学院  
陈章侠 德州职业技术学院  
郑开阳 枣庄职业学院  
张洪忠 临沂职业学院  
常中华 青岛职业技术学院  
刘月红 晋中职业技术学院  
赵 娟 山西旅游职业学院  
陈 炯 山西职业技术学院  
陈怀玉 山西经贸职业学院  
范文涵 山西财贸职业技术学院  
任利成 山西轻工职业技术学院  
郭长庚 许昌职业技术学院  
李庶泉 周口职业技术学院  
许国强 湖南有色金属职业技术学院  
孙 刚 南京信息职业技术学院  
夏东盛 陕西工业职业技术学院  
张雅珍 陕西工商职业学院  
王国强 甘肃交通职业技术学院  
周仲文 四川广播电视大学  
杨志超 四川华新现代职业学院  
董新民 安徽国际商务职业学院  
谭维奇 安庆职业技术学院  
张 燕 南开大学出版社

# 企业级卓越人才培养解决方案简介

企业级卓越人才培养解决方案(以下简称“解决方案”)是面向我国职业教育量身定制的应用型、技术技能人才培养解决方案。以教育部—滨海迅腾科技集团产学研合作协同育人项目为依托,依靠集团研发实力,联合国内职业教育领域相关政策研究机构、行业、企业、职业院校共同研究与实践的科研成果。本解决方案坚持“创新校企融合协同育人,推进校企合作模式改革”的宗旨,消化吸收德国“双元制”应用型人才培养模式,深入践行基于工作过程“项目化”及“系统化”的教学方法,设立工程实践创新培养的企业化培养解决方案。在服务国家战略:京津冀教育协同发展、中国制造 2025(工业信息化)等领域培养不同层次的技术技能人才,为推进我国实现教育现代化发挥积极作用。

该解决方案由“初、中、高”三个培养阶段构成,包含技术技能培养体系(人才培养方案、专业教程、课程标准、标准课程包、企业项目包、考评体系、认证体系、社会服务及师资培训)、教学管理体系、就业管理体系、创新创业体系等;采用校企融合、产学研融合、师资融合的“三融合”模式,在高校内共建大数据(AI)学院、互联网学院、软件学院、电子商务学院、设计学院、智慧物流学院、智能制造学院等;并以“卓越工程师培养计划”项目的形式推行,将企业人才需求标准、工作流程、研发规范、考评体系、企业管理体系引进课堂,充分发挥校企双方优势,推动校企、校际合作,促进区域优质资源共建共享,实现卓越人才培养目标,达到企业人才招录的标准。本解决方案已在全国几十所高校开始实施,目前已形成企业、高校、学生三方共赢的格局。

天津滨海迅腾科技集团有限公司创建于 2004 年,是以 IT 产业为主导的高科技企业集团。集团业务范围已覆盖信息化集成、软件研发、职业教育、电子商务、互联网服务、生物科技、健康产业、日化产业等。集团以科技产业为背景,与高校共同开展“三融合”的校企合作混合所有制项目。多年来,集团打造了以博士、硕士、企业一线工程师为主导的科研及教学团队,培养了大批互联网行业应用型技术人才。集团先后荣获天津市“五一”劳动奖状先进集体、天津市政府授予“AAA”级劳动关系和谐企业、天津市“文明单位”“工人先锋号”“青年文明号”“功勋企业”“科技小巨人企业”“高科技型领军企业”等近百项荣誉。集团将以“中国梦,腾之梦”为指导思想,在 2020 年实现与 100 所以上高校合作,形成教育科技生态圈格局,成为产学研协同育人的领军企业。2025 年形成教育、科技、现代服务业等多领域 100% 生态链,实现教育科技行业“中国龙”目标。

# 前 言

Node.js 是一个为开发人员提供开发服务器端应用的平台,且其基于 Chrome V8 引擎,具有单线程模式、非阻塞 I/O、轻量高效以及事件驱动等优点,可以方便地搭建响应速度快、易于扩展的网络应用。

本书以项目实战开发为基础,以 Node.js 原生模块和典型案例为主线,详细介绍 Node.js 开发的基础知识和相应案例实践,让读者全面、深入、透彻地了解 Node.js 开发的主要技术,并且能够和各种主流框架整合使用,提高实际开发水平和项目实战能力。

本书主要有八个项目,即 TF 物业系统客户端界面、TF 物业系统用户管理界面、TF 物业系统商品管理界面、TF 物业系统数据库表的建立、服务端用户管理功能、服务端商品管理功能、服务端缴费管理功能、客户端与服务端交互,循序渐进地讲述 Node.js 项目开发步骤及流程,通过本书的学习,读者可以更加熟练地使用 Node.js 与各种主流框架整合开发,了解项目开发的流程及最终的交互。

本书涵盖的主要内容有 Node.js 应用、REPL、timer 模块、events 模块、process、child\_process 模块、fs 模块、Path 模块、url 模块、MongoDB 数据库、HTTP 服务、Express、测试、部署发布等,内容丰富、实例典型、实用性强。并且设有学习目标、学习路径、任务描述、任务技能、任务实施、任务总结、英语角以及任务习题,结构条理清晰、内容详细,非常适合希望通过编码实例学习 Node.js 开发的人员阅读。

本书由牛永钢、康健、陈桂芳任主编,由周青政、雷长虹、黄祥书、田钰、陈观爱等共同任副主编,牛永钢负责统稿,康健、陈桂芳负责全面内容的规划,周青政、雷长虹、黄祥书、田钰、陈观爱负责整体内容编排。具体分工如下:项目一至项目三由周青政、雷长虹编写,康健负责全面规划;项目四至项目五由黄祥书编写,康健负责全面规划,项目六至项目八由田钰、陈观爱共同编写,陈桂芳负责全面规划。

本书理论条理清晰、实例操作讲解细致,实现了理论与实践的结合,操作步骤后有相对应的效果图,便于读者直观、清晰地看到操作效果,牢记书中的操作步骤,使读者在 Node.js 的学习过程中能够更加顺利。

天津滨海迅腾科技集团有限公司  
技术研发部

# 目 录

项目一 TF 物业系统客户端界面 .....	1
学习目标 .....	1
学习路径 .....	1
任务描述 .....	2
任务技能 .....	2
技能点 1 Node.js 概述 .....	2
技能点 2 Node.js 应用 .....	4
技能点 3 使用 WebStorm 调试 Node.js .....	6
任务实施 .....	12
任务总结 .....	27
英语角 .....	27
任务习题 .....	28
项目二 TF 物业系统用户管理界面 .....	29
学习目标 .....	29
学习路径 .....	29
任务描述 .....	30
任务技能 .....	30
技能点 1 REPL .....	30
技能点 2 console 模块 .....	33
技能点 3 timer 模块 .....	34
技能点 4 模块化 .....	38
任务实施 .....	41
任务总结 .....	60
英语角 .....	60
任务习题 .....	61
项目三 TF 物业系统商品管理界面 .....	63
学习目标 .....	63
学习路径 .....	63
任务描述 .....	64
任务技能 .....	64
技能点 1 Buffer .....	64



技能点 2 util 模块 .....	70
技能点 3 events 模块 .....	75
任务实施 .....	84
任务总结 .....	97
英语角 .....	97
任务习题 .....	97
<b>项目四 TF 物业系统数据库表的建立 .....</b>	<b>99</b>
学习目标 .....	99
学习路径 .....	99
任务描述 .....	100
任务技能 .....	100
技能点 1 process .....	100
技能点 2 child_process 模块 .....	107
技能点 3 函数 .....	113
任务实施 .....	115
任务总结 .....	122
英语角 .....	123
任务习题 .....	123
<b>项目五 服务端用户管理功能 .....</b>	<b>125</b>
学习目标 .....	125
学习路径 .....	125
任务描述 .....	126
任务技能 .....	126
技能点 1 fs 模块 .....	126
技能点 2 Stream(数据流) .....	140
技能点 3 Path 模块 .....	144
技能点 4 url 模块 .....	147
任务实施 .....	150
任务总结 .....	175
英语角 .....	175
任务习题 .....	176
<b>项目六 服务端商品管理功能 .....</b>	<b>177</b>
学习目标 .....	177
学习路径 .....	177
任务描述 .....	178
任务技能 .....	178
技能点 1 MongoDB 数据库 .....	178

技能点 2 MySQL 数据库 .....	186
技能点 3 HTTP 服务 .....	193
任务实施 .....	198
任务总结 .....	230
英语角 .....	230
任务习题 .....	231
<b>项目七 服务端缴费管理功能</b> .....	<b>233</b>
学习目标 .....	233
学习路径 .....	233
任务描述 .....	234
任务技能 .....	235
技能点 1 Express 框架 .....	235
技能点 2 数据库使用 .....	242
技能点 3 静态资源 .....	245
任务实施 .....	246
任务总结 .....	268
英语角 .....	268
任务习题 .....	269
<b>项目八 客户端与服务端交互</b> .....	<b>270</b>
学习目标 .....	270
学习路径 .....	270
任务描述 .....	271
任务技能 .....	271
技能点 1 测试 .....	271
技能点 2 部署发布 .....	279
任务实施 .....	280
任务总结 .....	299
英语角 .....	299
任务习题 .....	299

# 项目一 TF 物业系统客户端界面



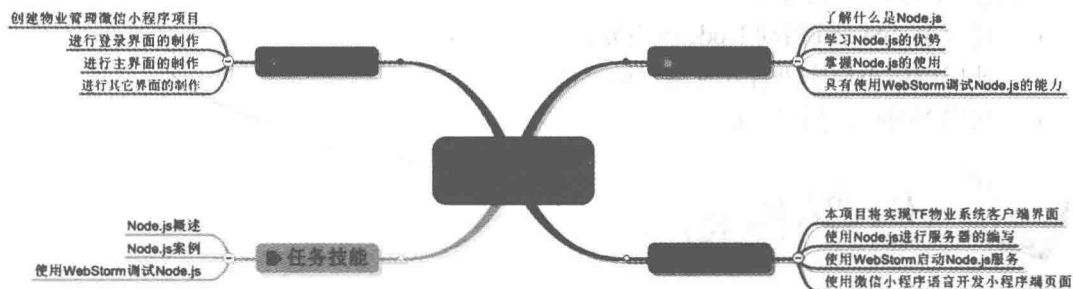
## 学习目标

通过 TF 物业系统客户端界面的实现,了解 Node.js 的特点,学习 Node.js 的应用场景,掌握使用 Node.js 搭建服务器平台,具有使用 WebStorm 调试 Node.js 的能力。在任务实现过程中:

- 了解什么是 Node.js。
- 学习 Node.js 的优势。
- 掌握 Node.js 的使用。
- 具有使用 WebStorm 调试 Node.js 的能力。

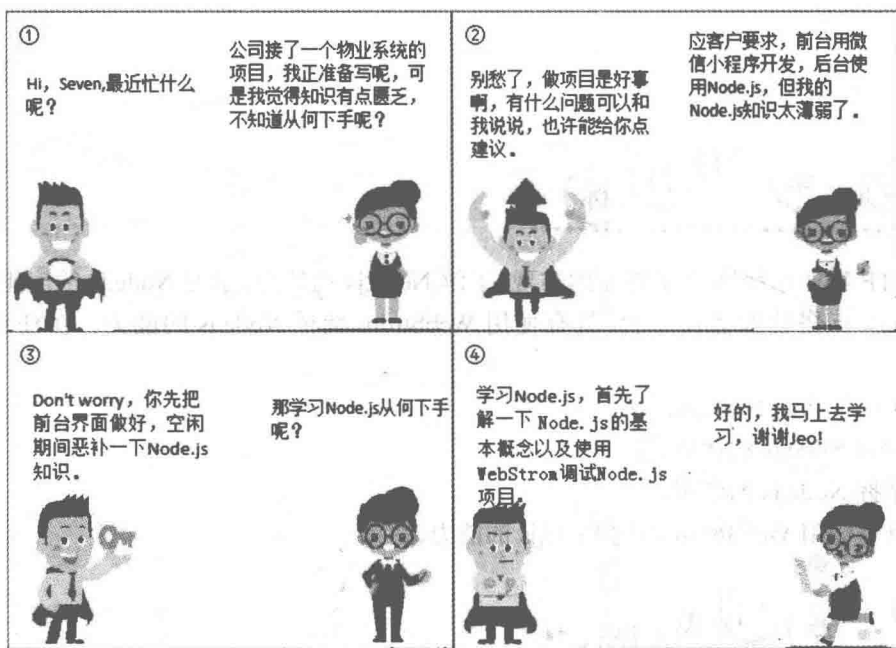


## 学习路径



## 任务描述

### 【情境导入】



### 【功能描述】

本项目将实现 TF 物业系统客户端界面。

- 使用 WebStorm 启动 Node.js 服务。
- 使用微信小程序语言开发客户端界面。
- 使用 Node.js 进行交互。

## 任务技能

### 技能点 1 Node.js 概述

#### 1 Node.js 简介

Node.js 由 Ryan Dahl 于 2009 年 5 月发布,是基于 Chrome JavaScript 运行时建立的平台,其实质是对 Chrome V8 引擎(执行 JavaScript 的速度非常快,性能非常好)进行了封装,用于方

便地搭建响应速度快、易于扩展的网络应用。Node.js 对一些特殊用例进行优化,提供替代的 API,使得 V8 在非浏览器环境下运行得更好。其应用是由 JavaScript 语言开发,因此,具有 JavaScript 的多种特点。Node.js 具有如下特点。

- 一个命令行工具。
- 利用 V8 引擎。
- 拥有子进程,在同一时间可以做多个事情。
- 是基于事件的,类似 AJAX 的工作都可以在服务器端完成。
- 浏览器和服务器端之间共享代码。
- 方便与数据库连接。

## 2 Node.js 的优势

Node.js 是一个服务器端运行的 JavaScript 脚本语言,大部分的 API 与客户端 JavaScript 保持一致,且在单线程模式下工作。单线程模式是 Node.js 的一大优点,不仅如此,Node.js 还具有其他优点。Node.js 主要优点如下。

### (1) 单线程模式

单线程指当遇到需要加载数据库等请求时,它会将其放入“队列”中执行,待下一轮事件循环时再判断能否执行它的回调函数,与多线程编程不同,可以忽略状态的同步问题,没有死锁的存在,也没有线程上下文交换所带来的性能上的开销。

### (2) 非阻塞 I/O

Node.js 的非阻塞 I/O 处理对系统资源耗用低,性能高,且具有出众的负载能力,非常适合用作依赖其他 I/O 资源的中间层服务。例如,用户发起一个读取文件描述符操作时,函数立即返回,不作任何等待,进程继续执行。

### (3) 轻量高效

Node.js 轻量高效是数据密集型分布式部署环境下实时应用系统的完美解决方案。

### (4) 事件驱动、异步编程

事件驱动主要是通过事件或状态的变化来进行应用程序的流程控制,一般通过事件监听完成,一旦事件被检测到,则调用相应的回调函数(回调函数是常用的解决异步的方法)。Node.js 的匿名函数和闭包特性非常适合事件驱动、异步编程。

## 3 为什么学习 Node.js

### (1) 学习 Node.js 的理由

Node.js 具有 NPM 包管理系统,能解决 Node.js 代码部署上的大多数问题。使用“npm install 包/库名”可以安装必要的包/库,也可以下载并安装其他人编写的命令程序。学习 Node.js 的理由如下。

- 相比于其他开发语言,更容易配置。
- 基于 JavaScript 运行,可在服务器和客户端使用相同的语言,甚至可在它们之间共享一些代码。
- 单线程事件驱动系统,即使面对大量的请求,也可快速一次处理。
- 通过 NPM 可访问的软件包不断增加,包括客户端和服务端的库/模块,以及用于

Web 开发的命令行工具。

- 适合原型设计、敏捷开发和快速产品迭代。
- 适合具有大量并发连接的应用程序,并且每个请求只需要很少的 CPU 周期。

## (2) Node.js 的应用场景

Node.js 应用广泛,非常适合搭建静态资源服务器、制作聊天应用程序等,还可以应用到不同的场景上,如:

- 高度事件驱动的应用程序和严重的 I/O 限制。
- 处理大量与其他系统连接的应用程序。
- 即时应用程序。
- 高流量、可扩展的应用。
- 建立网络应用程序。



### 快来扫一扫!

当你了解了什么是 Node.js 之后,你是否想要再了解下 Node.js 是如何发展到今天这一地步的,扫描右方二维码,看看 Node.js 的发展史!



## 技能点 2 Node.js 应用

使用 Node.js 搭建服务器平台的步骤如下:

第一步:创建 HelloNode.js 文件。

第二步:引入 http 模块。

进入 HelloNode.js 文件,使用 require 指令加载 http 模块,并将实例化的 http 赋值给定义的变量 http,代码如下所示。

```
var http = require("http");
```

第三步:创建服务器。

使用 http.createServer() 方法创建服务器,并使用 listen() 方法监听 3000 端口。通过 request、response 参数来接收和响应数据。代码如下所示。

```
var http = require("http");
```

```
http.createServer(function(request, response){  
  response.writeHead(200, {'Content-Type': 'text/plain'});  
  response.end("hello Node.js");  
}).listen(3000);  
// 命令窗口提示内容  
console.log(' http://127.0.0.1:3000/');
```

第四步: 打开命令窗口并切换到 HelloNode.js 文件所在目录下, 输入“node HelloNode.js”命令启动服务。效果如图 1.1 所示。



图 1.1 启动服务

第五步: 打开浏览器访问 <http://127.0.0.1:3000/>, 出现内容为“hello Node.js”的网页。效果如图 1.2 所示。



图 1.2 访问浏览器

## 技能点 3 使用 WebStorm 调试 Node.js

除了在命令窗口运行 Node.js 项目外, WebStorm 根据 Node.js 的一系列特性在新版本发布中集成了 Node.js 运行环境, 以方便进行 Node.js 的编译和调试。使用 WebStorm 调试 Node.js 的步骤如下:

第一步: 安装 Node.js。

第二步: 下载 WebStorm 代码编辑器, 下载网址 <https://www.jetbrains.com/webstorm/>, 网址效果如图 1.3 所示。下载后进行安装。



图 1.3 WebStorm 下载网址

第三步: 安装后打开 WebStorm, 点击菜单中的“File”, 选择“Settings”, 进入设置界面, 效果如图 1.4、1.5 所示。



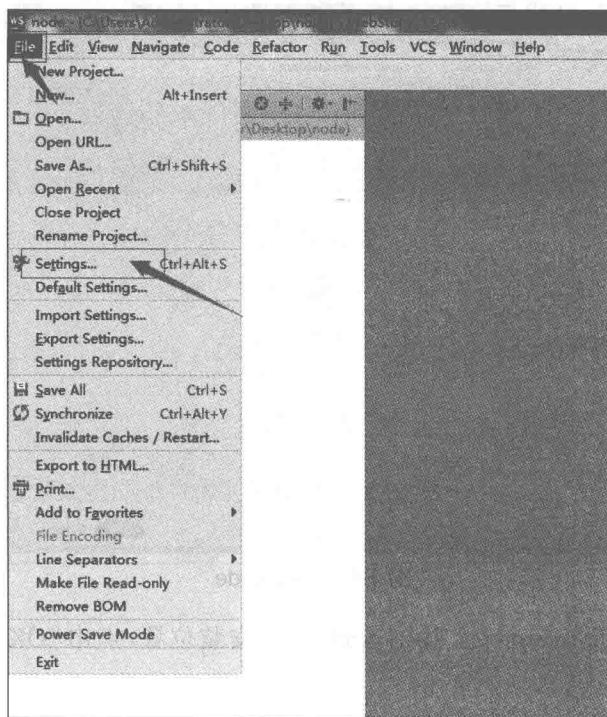


图 1.4 进入设置界面

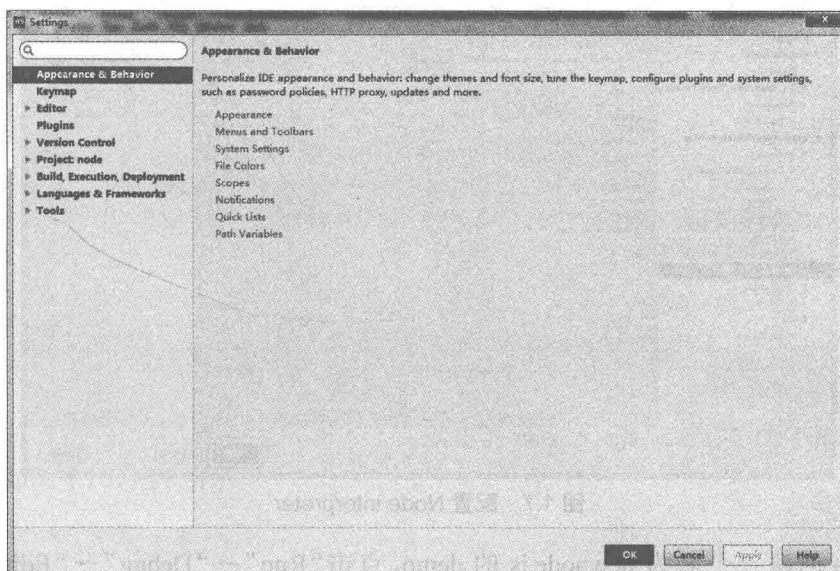


图 1.5 进入设置界面

第四步: 在设置界面输入“Node”进行查找, 选择“Node.js and NPM”界面, 效果如图 1.6 所示。