**全栈专业课程大纲**

2018.4修订

[第1阶段课程大纲 6](#_Toc23445)

**[第1单元](#_Toc3077)** [HTML基础及常用标签 6](#_Toc3077)

**[第2单元](#_Toc19198)** [表单 7](#_Toc19198)

**[第3单元](#_Toc5671)** [CSS样式表及选择器 8](#_Toc5671)

**[第4单元](#_Toc29090)** [文字、段落属性、背景属性、列表属性及CSS盒模型 9](#_Toc29090)

**[第5单元](#_Toc26816)** [浮动及其清除 11](#_Toc26816)

**[第6单元](#_Toc25959)** [display与overflow属性的应用及定位 11](#_Toc25959)

**[第7单元](#_Toc3425)** [HTML5新特性及CSS3选择器 12](#_Toc3425)

**[第8单元](#_Toc12246)** [项目一 13](#_Toc12246)

**[第9单元](#_Toc23744)** [CSS3变形与过渡 13](#_Toc23744)

**[第10单元](#_Toc11989)** [CSS3动画效果 14](#_Toc11989)

**[第11单元](#_Toc11531)** [CSS3弹性盒模型 14](#_Toc11531)

**[第12单元](#_Toc6196)** [响应式网页设计 15](#_Toc6196)

**[第13单元](#_Toc7112)** [移动端项目案例 16](#_Toc7112)

**[第14单元](#_Toc7823)** [Less及Sass 16](#_Toc7823)

**[第15单元](#_Toc6053)** [综合项目案例 16](#_Toc6053)

**[第16单元](#_Toc27218)** [SVN的搭建及用法 16](#_Toc27218)

**[第17单元](#_Toc12780)** [Github 16](#_Toc12780)

**[第18单元](#_Toc28652)** [综合项目案例 16](#_Toc28652)

[第2阶段课程大纲 16](#_Toc15478)

**[第1单元](#_Toc13416)** [js的概述盒基础语法 17](#_Toc13416)

**[第2单元](#_Toc8273)** [流程控制语句 18](#_Toc8273)

**[第3单元](#_Toc23563)** [函数 19](#_Toc23563)

**[第4单元](#_Toc31347)** [数组的使用 19](#_Toc31347)

**[第5单元](#_Toc31399)** [string对象 20](#_Toc31399)

**[第6单元](#_Toc14132)** [Math与日期对象 20](#_Toc14132)

**[第7单元](#_Toc7107)** [bom 21](#_Toc7107)

**[第8单元](#_Toc27234)** [dom1 22](#_Toc27234)

**[第9单元](#_Toc26828)** [Dom(3) 23](#_Toc26828)

**[第10单元](#_Toc20781)** [事件 24](#_Toc20781)

**[第11单元](#_Toc17885)** [动画1 26](#_Toc17885)

**[第12单元](#_Toc5183)** [正则表达式 26](#_Toc5183)

**[第13单元](#_Toc16884)** [JQ1 29](#_Toc16884)

**[第14单元](#_Toc21429)** [JQ3 30](#_Toc21429)

**[第15单元](#_Toc23937)** [JQ3 31](#_Toc23937)

**[第16单元](#_Toc5087)** [面向对象1 31](#_Toc5087)

**[第17单元](#_Toc690)** [面向对象3 33](#_Toc690)

**[第18单元](#_Toc6375)** [数据存储 34](#_Toc6375)

[第3阶段课程大纲 34](#_Toc9455)

**[第1单元](#_Toc28883)** [RequireJS 35](#_Toc28883)

**[第2单元](#_Toc26975)** [Node.js的简介 35](#_Toc26975)

**[第3单元](#_Toc9859)** [Node.js 36](#_Toc9859)

**[第4单元](#_Toc2160)** [Express 36](#_Toc2160)

**[第5单元](#_Toc11663)** [原生Ajax 37](#_Toc11663)

**[第6单元](#_Toc22705)** [jQuery中的Ajax和跨域 38](#_Toc22705)

**[第7单元](#_Toc22072)** [MongoDB 38](#_Toc22072)

**[第8单元](#_Toc13707)** [MongoDB 案例练习 39](#_Toc13707)

**[第9单元](#_Toc20940)** [Express项目 39](#_Toc20940)

**[第10单元](#_Toc18575)** [Angular -01 40](#_Toc18575)

**[第11单元](#_Toc21283)** [Angular -02 41](#_Toc21283)

**[第12单元](#_Toc12923)** [Angular -03 41](#_Toc12923)

**[第13单元](#_Toc29738)** [Angular -04 项目练习 42](#_Toc29738)

**[第14单元](#_Toc14628)** [Angular -05 项目练习 42](#_Toc14628)

**[第15单元](#_Toc4243)** [ES6 42](#_Toc4243)

**[第16单元](#_Toc2529)** [ES6 43](#_Toc2529)

**[第17单元](#_Toc1818)** [ES6 项目练习 43](#_Toc1818)

**[第18单元](#_Toc23603)** [gulp，webpack 43](#_Toc23603)

[第4阶段课程大纲 44](#_Toc7695)

**[第1单元](#_Toc24247)** [Vue.js(一) 45](#_Toc24247)

**[第2单元](#_Toc27899)** [Vue.js(一) 45](#_Toc27899)

**[第3单元](#_Toc26794)** [事件处理器与表单 46](#_Toc26794)

**[第4单元](#_Toc31898)** [组件与过渡 46](#_Toc31898)

**[第5单元](#_Toc31877)** [自定义指令与路由 47](#_Toc31877)

**[第6单元](#_Toc6019)** [Vue.js实战(一) 47](#_Toc6019)

**[第7单元](#_Toc19577)** [第七天 Vue.js 项目实战 (二) 48](#_Toc19577)

**[第8单元](#_Toc22837)** [Vue.js 项目实战 (三) 48](#_Toc22837)

**[第9单元](#_Toc14295)** [Vue.js 项目实战 (四) 49](#_Toc14295)

**[第10单元](#_Toc13921)** [Vue.js 项目实战 (五) 49](#_Toc13921)

**[第11单元](#_Toc12452)** [微信小程序(一) 50](#_Toc12452)

**[第12单元](#_Toc20646)** [微信小程序(二) 50](#_Toc20646)

**[第13单元](#_Toc13038)** [微信小程序(三) 51](#_Toc13038)

**[第14单元](#_Toc14624)** [微信小程序实战(一) 51](#_Toc14624)

**[第15单元](#_Toc919)** [微信小程序实战(二) 52](#_Toc919)

**[第16单元](#_Toc8312)** [微信小程序实战(三) 52](#_Toc8312)

**[第17单元](#_Toc23222)** [项目答辩 53](#_Toc23222)

**[第18单元](#_Toc10319)** [简历制作 53](#_Toc10319)

[第5阶段课程大纲 53](#_Toc32187)

**[第1单元](#_Toc26947)** [MongoDB数据库安装 54](#_Toc26947)

**[第2单元](#_Toc12280)** [MongoDB 54](#_Toc12280)

**[第3单元](#_Toc6623)** [vue项目1 54](#_Toc6623)

**[第4单元](#_Toc21574)** [vue项目2 55](#_Toc21574)

**[第5单元](#_Toc26157)** [vue项目3 55](#_Toc26157)

**[第6单元](#_Toc12497)** [vue项目4 56](#_Toc12497)

**[第7单元](#_Toc17612)** [vue项目5 57](#_Toc17612)

**[第8单元](#_Toc3671)** [vue自选项目一 57](#_Toc3671)

**[第9单元](#_Toc15793)** [vue自选项目二 57](#_Toc15793)

**[第10单元](#_Toc30307)** [vue自选项目三 57](#_Toc30307)

**[第11单元](#_Toc30720)** [vue自选项目四 57](#_Toc30720)

**[第12单元](#_Toc12354)** [vue自选项目五 58](#_Toc12354)

**[第13单元](#_Toc20383)** [简历制作一 58](#_Toc20383)

**[第14单元](#_Toc23242)** [简历制作二 58](#_Toc23242)

**[第15单元](#_Toc30836)** [面试技巧一 58](#_Toc30836)

**[第16单元](#_Toc12584)** [模拟面试一 58](#_Toc12584)

**[第17单元](#_Toc9316)** [模拟面试二 58](#_Toc9316)

**[第18单元](#_Toc15185)** [模拟面试三 58](#_Toc15185)

# 第1阶段课程大纲

## HTML基础及常用标签

### HTML简介

### HTML标签的属性和属性值

### HTML元素的分类

#### 1.3.1 行内元素（内联元素）

#### 1.3.2 块元素

#### 1.3.3 行内块元素

### HTML元素的嵌套原则

#### 块元素的三级分类

#### 元素嵌套原则

### HTML基本元素

#### 标题标记以及属性

#### 段落标记以及对齐方式属性

#### 图像标记以及属性

#### 路径(绝对相对、相对路径)

#### 水平线标记以及属性

#### HTML里面常用的特殊符号（空格等）

### 无序列表

#### 无序列表标签

#### 无序列表type属性

### 有序列表ol

#### 有序列表标签

#### 有序列表type属性和reversed(倒序)属性

### 自定义列表 dl

#### 自定义列表标签（定义项、描述项）

### 超链接的语法

#### 超链接的标签

#### 超链接href属性

#### 超链接target属性

#### 超链接对象（文本、图像、空连接、锚点链接）

### 表格的基本结构

#### table标记

#### thead、tbody、tfoot、caption

#### tr标记

#### td标记

#### th标记

### 表格标记table的属性

#### border/width/height属性

#### cellspacing/cellpadding属性

#### background/bgcolor属性

#### algin属性和valgin属性

### 单元格的合并与拆分

#### 水平合并单元格colspan

#### 垂直合并单元格rowspan

### 使用表格布局网页

## 表单

### 表单元素及其属性

#### action 属性

#### method属性

#### target属性

### input控件

#### 文本框

#### 密码框

#### 单选框

#### 复选框

#### 文件域

#### 提交按钮

#### 普通按钮

#### 重置按钮

#### 图像域

### 下拉菜单

#### 下拉列表框的结构

#### selected属性

### 文本域及其属性

#### cols属性

#### rows属性

### 案例

## CSS样式表及选择器

### CSS的基本概念及语法

### HTML样式表

#### 行内样式

#### 内嵌样式

#### 外部链接样式

#### 外部导入样式

#### 样式表的优先级

### CSS选择器

#### 全局选择器

#### 标签选择器

#### 类选择器

#### ID选择器

#### 关系选择器（包含选择器、子选择器、相邻选择器）

#### a元素伪类选择器

#### 属性选择器

#### 通用兄弟选择器

#### 相邻兄弟选择器

#### 选择器的权值

## 文字、段落属性、背景属性、列表属性及CSS盒模型

### 文字属性

#### color字体颜色

#### font-size字号

#### font-family字体

#### font-weight字体加粗

### 段落属性

#### 文字修饰（text-decoration）

#### 水平对齐方式（text-align）

#### 文本缩进（text-indent）

#### 文本行高 (line-height)

### 背景属性

#### background-color背景颜色

#### background-image背景图像

#### background-repeat背景重复

#### background-position背景图像位置

#### background-attachment背景附件

#### background背景复合属性

### 列表属性

#### list-style-type

#### list-style-image

#### list-style-position

### CSS盒模型

### 设置尺寸

#### 设置宽度width

#### 设置高度height

### 边框属性

#### 边框复合属性border

#### 上边框border-top

#### 右边框border-right

#### 下边框border-bottom

#### 左边框border-left

#### 常用边框样式：solid 实线、dashed 虚线、dotted点线

### 内填充padding

#### 上内填充padding-top

#### 右内填充padding-right

#### 下内填充padding-bottom

#### 左内填充padding-left

#### 掌握WEB标准中padding的使用规范

#### padding对元素宽高的影响

### 外边距margin

#### 上外边距margin-top

#### 右外边距丁margin-right

#### 下外边距margin-bottom

#### 左外边距margin-left

#### 掌握WEB标准中margin的使用规范

## 浮动及其清除

### 什么是网页布局

### 浮动float属性

#### left左浮动

#### right右浮动

### 清除浮动clear属性

### 浮动属性的应用

#### Div的浮动（实现布局）

#### 列表的浮动（实现导航）

### 使用盒模型布局搭建布局色块

### 案例

## display与overflow属性的应用及定位

### dislay:inline的应用

### display:block的应用

### display:none的应用

### display:inline-block的应用

### CSS样式中overflow的运用规范

### CSS样式中visibility的运用规范

### visibility:hidden与display:none的区别

### 掌握定位属性的四个值：

#### 默认值：static

#### 绝对定位：absolute

#### 相对定位：relative

#### 固定定位：fixed

### 掌握定位属性方向值运用规范

#### 上部：top的运用规范

#### 下部：bottom的运用规范

#### 左部：left的运用规范

#### 右部：right的运用规范

### 掌握z-index空间位置的运用规范

### 案例

## HTML5新特性及CSS3选择器

### HTML5 概述

### HTML5 新增特性概况介绍

#### 新增结构化语义标签

#### 新增表单相关内容

#### 完全支持CSS3

#### 新增多媒体播放：音频audio、视频video

#### 新增绘图：canvas

#### 新增本地存储（离线存储、web存储、localstorage、localsession）

#### 新增拖拽功能API

#### 支持实现响应式布局

### HTML5新标签

#### 新增结构化标签

#### 新增非结构化标签

### 新增表单新特性

#### 新增表单元素

#### 新增input元素新类型

#### 新增表单元素新属性

### CSS3选择器

#### 基本选择器

#### 属性选择器

#### 伪类选择器

#### 动态伪类    【爱恨原则：love/hate】

#### 元素状态伪类【跟form表单相关的伪类】

#### 否定伪类

#### 结构伪类【nth伪类】

#### 伪元素选择器

### CSS3边框、背景、渐变、文字特效

### CSS3雪碧图

#### 雪碧图的好处

#### 雪碧图的用法

## 项目一

### 利用HTML5新标签及CSS3新增选择器制作页面

## CSS3变形与过渡

### CSS3变形效果

#### transform

#### transform-origin

#### translate变形

#### rotate变形

#### scale变形

#### skew变形

#### matrix变形

### 过渡效果

#### 过渡简介

#### transition-property

#### transition-duration

#### transition-timing-function

#### transition-delay

### CSS3变形及过渡案例

### 3D 变形 （了解）

#### transform-style

#### perspective

## CSS3动画效果

### 动画简介

#### animation 属性详解

#### 自定义关键帧动画

### 第三方动画库：animate.css

## CSS3弹性盒模型

### 弹性盒模型概述

#### 弹性盒容器

#### 弹性子元素

### 弹性盒容器的六大属性

#### flex-direction

#### flex-wrap

#### flex-flow

#### justify-content

#### align-items

#### align-content

### 弹性项目的六大属性

#### order

#### flex-grow

#### flex-shrink

#### flex-basis

#### flex

#### align-self

### 双飞翼布局

### 圣杯布局

## 响应式网页设计

### 目前常用的网页布局

### 响应式网页设计（RWD）

#### 响应式网页设计的概念

#### 响应式网页设计的优势和不足

#### 响应式网页设计的使用场景

#### 响应式网页设计的具体做法

### 媒介查询

#### 媒介查询的概念

#### 媒介查询的意义

#### 媒介查询的具体做法

## 移动端项目案例

## Less及Sass

## 综合项目案例

## SVN的搭建及用法

## Github

## 综合项目案例

# 第2阶段课程大纲

## js的概述盒基础语法

### Javascript概述

### JavaScript组成

#### 1.3.1 ECMAScript

#### 1.3.3 文档对象模型(DOM)

#### 1.3.3 浏览器对象模型（BOM）

### JavaScript引入

### JavaScript语句

### 变量

#### 1.5.1 变量的概念

#### 1.5.3 变量的声明

### ypeof操作符

### 数字（Number）类型

#### 1.7.1 整型

#### 1.7.3 浮点型

#### 1.7.3 数值范围

#### 1.7.4 NaN

### 文本（String）类型

#### 1.8.1 字符串直接量

#### 1.8.3 转义字符

#### 1.8.3 字符串的使用

### 布尔（Boolean）类型

### Undefined类型

### Null类型

### 对象（Object）类型

### 数据类型转换

#### 1.13.1 各种数据类型转换为字符串

#### 1.13.3 各种数据类型转换为数值

#### 1.13.3 各种数据类型转换为布尔值

### 表达式的概念

### 运算符

#### 1.15.1 算术运算符

#### 1.15.3 关系运算符

#### 1.15.3 逻辑运算符

#### 1.15.4 赋值运算符

#### 1.15.5 条件运算符

#### 1.15.6 运算符优先级

## 流程控制语句

### if语句

### switch语句

### for语句

### 跳转语句

#### 3.4.1 break语句

#### 3.4.3 continue语句

### while语句

### do-while语句

### 案例

## 函数

### 函数定义

### 函数调用

### 函数参数

#### 3.5.1 形参

#### 3.5.3 实参

### 函数的返回值

### 局部变量和全局变量（了解）

## 数组的使用

### 定义数组

### 数组的属性

### 遍历数组

### arguments对象

### 二维数组(了解)

### 数组的方法

#### 4.6.1 join()

#### 4.6.3 reverse()

#### 4.6.3 sort()

#### 4.6.4 concat()

#### 4.6.5 slice()

#### 4.6.6 splice()

#### 4.6.7 push()和pop()

#### 4.6.8 unshift()和shift()

#### 4.6.9 toString()

### 数组的运用

#### 4.7.1数组的去重

## string对象

### String对象

### tring对象的属性

### String对象的方法

#### 5.3.1 字符方法5.3.3 字符串操作方法5.3.3 字符串位置方法

## Math与日期对象

### Math对象

### Math对象的属性

### Math对象的方法

#### 6.3.1 min()和max()方法

#### 6.3.3 ceil()

#### 6.3.3 floor()

#### 6.3.4 round()

#### 6.3.5 random()

#### 6.3.6 abs()

#### 6.3.7 pow()

#### 6.3.8 sqrt()

### Date对象的方法

#### 6.4.1 toLocaleString()

#### 6.4.3 toString()

#### 6.4.3 valueOf()

#### 6.4.4 getTime()

#### 6.4.5 getFullYear()

#### 6.4.6 getMonth()

#### 6.4.7 getDate()

#### 6.4.8 getDay()

#### 6.4.9 getHours()

#### 6.4.10 getMinutes()

#### 6.4.10 getSeconds()

#### 6.4.13 getMilliseconds()

### 获取对象

#### 6.5.1 getElementById()

#### 6.5.3 getElementsByTagName()

#### 6.5.3 getElementsByName()

## bom

### window对象的讲解

### 全局作用域

### 窗口关系及框架

### 窗口位置和大小

### 打开窗口

### 间歇调用和超时调用

### 系统对话框

### location对象

### navigator对象

### screen对象

### history对象

## dom1

### DOM概念:

### 节点和节点层次关系

#### 8.3.1 节点层次：DOM树形结构

#### 8.3.3 节点类型：文档中的内容，包括标签、文本、注释、标签属性等

#### 8.3.3 节点属性：

#### 8.3.8 节点关系

#### 8.3.5 常用属性

### 文档写入

#### 8.3.1 document.write():

#### 8.3.3 document.writeIn():

### 查找元素

#### 8.4.1 getElementById()

#### 8.4.3 getElementsByTagName()

#### 8.4.3 getElementsByName()

#### 8.4.4 获取元素中的属性值

#### 8.4.5 设置元素中的属性值

### 标签元素的属性

#### 8.5.1获取元素节点的标签名

#### 8.5.3 获取、设置元素节点里的内容（内容可以包含html标签）

#### 8.5.3 outerHTML/innerText，非w3c的DOM标准

#### 8.5.4 innerHTML和outerHTML的区别

#### 8.5.5 innerText

#### 8.5.6 style.CSS

### 节点遍历

#### 8.6.1 childNodes

#### 8.6.3 firstChild

#### 8.6.3 lastChild

#### 8.6.4 ownerDocument

#### 8.6.5 parentNode

#### 8.6.6 previousSibling

#### 8.6.7 nextSibling

## Dom(3)

### 标签节点的操作方法

#### 9.1.1 document.creatElement()（最常用）

#### 9.1.3 document.creatTextNode()

#### 9.1.3 append Child()、

#### 9.1.4 insertBefore(插入的新节点，原节点)

#### 9.1.5 body.insertBefore(header, h1);

#### 9.1.6 replaceChild(新节点，老节点)

#### 9.7.7 div.replaceChild(span,p)

#### 9.7.8 removeChild()

#### 9.7.9 div.removeChild(p)

#### 9.7.10 remove()

#### 9.7.10 div.remove()

### 案例

#### 9.3.1 点击按钮创建按li

#### 9.3.3 点击按钮反向创建li

#### 9.3.3 点击按钮创建li并且可以删除节点

#### 9.3.4 利用for循操作dom

### 克隆节点

### DOM常见属性

#### 9.4.1 offsetWidth：获取元素 度(包含border)

#### 9.4.3 offsetHeight：获取元素的高度(包含border)

#### 9.4.3 clientWidth：获取元素的宽度( 不包含border)

#### 9.4.4 clientHeight：获取元素的高度 (不包含border)

#### 9.4.5 offsetTop：获取定 位元素距离父级上边的距离

#### 9.4.6 offsetLeft：获取 定位元素距离父级左边的距离

### 滚动偏移动(淘宝返回顶部)

#### 9.5.1 滚动条事件

#### 9.5.3 onscroll

#### 9.5.3使用DOM操作实现图- 片懒加载

#### 9.5.4图片

#### 9.5.5图片预加载

### 作业

#### 9.6.1留言板

## 事件

### 什么是事件

#### 10.1.1 window.onload

#### 10.1.3 window.onscroll

### 事件对象

#### 10.3.1 获取对象

#### 10.3.3 事件对象的兼容写法

### 鼠标事件mouse

#### 10.3.1 onclick:点击事件,按下并抬起

#### 10.3.3 onmousedown:鼠标按下

#### 10.3.3 onmouseup:鼠标抬起

#### 10.3.4 onmousemove:鼠标移动

#### 10.3.5 ondbclick:双击事件

#### 10.3.6 oncontextmenu:右键菜单

### 键盘事件key

#### 10.4.1 onkeydown: 键盘按下

#### 10.4.3 onkeyup:键盘抬起

#### 10.4.3 可通过keycode属性获取唯一标识码

#### 10.4.4 实例

### 表单事件

#### 10.5.1 onfocus:获取焦点

#### 10.5.3 onblur:失去焦点

#### 10.5.3 onsubmit:提交事件

#### 10.5.4 ev.preventDefault():阻止默认事件。不兼容IE6~8

#### 10.5.5 return false :阻止默认事件，没有兼容性问题

### on的方式

#### 10.6.1 给同一个元素绑定两次事件,使用on的方式，后面会覆盖前面的

#### 10.6.3 addEventListener()

#### 10.6.3.1 不会发生覆盖的情况

#### 10.6.3.3 不兼容IE6~8

#### 10.6.3.3 removeEventListener()

#### 10.6.3 回调函数

#### 10.6.3.1 把函数当成参数传入

### 事件冒泡

#### 10.7.1 一个标签元素处理完事件之后，它的父元素、父元素的父元素也都会处理事件

#### 10.7.2 取消事件冒泡

#### 10.7.3 addEventListener() 冒泡的方式

#### 10.7.4 event.stopPropagation();取消事件传递

#### 10.7.5 addEventListener(false) 冒泡的方式

#### 10.7.6 addEventListener(true) 下沉的方式

### 阻止事件传递

#### 10.8.1 ev.cancelBubble = true或者ev.stopPropagation()

#### 10.8.3 实例:使用事件绑定的方式实现拖拽

### 鼠标滚轮事件

#### 10.9.1 非火狐: mousewheel

#### 10.9.3 火狐:DOMMouseScroll

#### 10.9.3 案例:使用滚轮改变图片的大小

### 实例

#### 10.10.1 拖拽效果

#### 10.10.3 onmousedown

#### 10.10.3 onmousemove

#### 10.10.4 onmouseup

#### 10.10.5 注意：事件加给document

## 动画1

### 小球移动

### 碰壁反弹

### 无间断滚动

### 轮播图

### 模态窗口

## 正则表达式

### 什么是正则？

#### 13.1.1 Regular Expression，一套匹配规则，匹配想要的东西：把人类所认可的一种规则使用计算机能读懂的语言表达出来，正则正能操作字符串，是一种正常人类很难读懂的语法格式。

### 正则表达式对象的创建

#### 13.3.1 语法

#### 13.3.3 方法

#### 13.3.3.1 test():获取到匹配的字符返回true，否则返回false

#### 13.3.3.3 exec():获取到匹配的字符并返回，如果没有则返回null

### 字符串的方法

#### 13.3.1 str.search(/haha/)

#### 13.3.3 str.replace(/haha/, “hehe”)

#### 13.3.3 str.match(/\d/g):找到匹配的字符并返回

### 正则语法规则

#### 13.4.1 modifiers（修饰词）

#### 13.4.3 i 匹配时忽略大小写

#### 13.4.3 g 执行时全局匹配:查找所有匹配项，而非查找到一个即停止

#### 13.4.3 匹配模式

#### 13.4.3.1 小括号:(abc)

#### 13.4.3.2 中括号:[13345]

#### 13.4.3.3 [0-9]:包括所有数字

#### 13.4.3.4 [abc]:包括a、b、c其中的一个

#### 13.4.3.5 [a-z]:包括所有的小写字母

#### 13.4.3.6 [A-Z]:包括所有的大写字母

#### 13.4.3.7 [A-z]:包括所有的字母

#### 13.4.3.8 [0-9a-z]:包含所有的数字和小写字母

#### 13.4.3.9 [^ab]: 除了字符ab

#### 13.4.3 元字符

#### 13.4.3.1 点:代表任何的字符,注意:如果需要加.的时候，需要进行转义.

#### 13.4.3.3 \w：任何字母、数字、下划线

#### 13.4.3.3 \W：任何非字母、数字、下划线

#### 13.4.3.4 \d：任何数字

#### 13.4.3.5 \D：任何非数字

#### 13.4.3.6 \s：空白字符

#### 13.4.3.7 \S：非空白字符

#### 13.4.3.8 \b：查找单词边界

#### 13.4.3.9 \B：查找非单词边界

#### 13.4.3.10 了解

#### \0：查找NULL字符

#### \n：查找换行符

#### \f： 查找分页符

#### \r：查找回车符

#### \t：查找制表符

#### \v：查找垂直制表符

#### 13.4.4 量词

#### 13.4.4.1 ^n：匹配开头为n的字符串

#### 13.4.4.3 n$：匹配结尾为n的字符串

#### 13.4.4.3 n+：匹配任何包含至少一个n的字符串

#### 13.4.4.5 n{x}： 匹配任何包含x个n的字符串

#### 13.4.4.6 n{x, y}：匹配包含x到y个n的字符串

#### 13.4.4.7 了解

#### n\*：匹配任何包含0个或者多个n的字符串

#### n?：匹配任何包含0个或1个n的字符串

#### ?=n：匹配任何其后紧接指定字符串n的字符串

#### ?!n：匹配任何其后没有紧接字符串n的字符串

### 实例

#### 13.5.1 判断全部数字

#### 13.5.3 判断文件后缀为PDF

#### 13.5.3 判断手机号

#### 13.5.4 判断邮箱

#### 13.5.5 判断IP地址

#### 13.6转义字符

#### 13.7判断密码强度

#### 13.8 检测网址

#### 13.9预定义模式

#### 中文匹配

#### 匹配1~355之间的值

#### 检测ip地址

#### 工具网站：[https://regexper.com](https://regexper.com/)

## JQ1

### jQuery入门

#### 13.1.1什么是 jQuery

#### 13.1.3学习 jQuery 的条件

#### 13.1.3 jQuery 的版本

#### 13.1.4 jQuery 的功能和优势

#### 13.1.5下载及运行 jQuery

### 基础核心

#### 13.3.1代码风格

#### 13.3.3加载模式

#### 13.3.3对象互换

#### 13.3.4多个库之间的冲突

### 常规选择器

#### 13.3.1简单选择器

#### 13.3.3进阶选择器

#### 13.3.3高级选择器

### 过滤选择器

#### 13.4.1基本过滤器

#### 13.4.3内容过滤器

#### 13.4.3可见性过滤器

#### 13.4.4子元素过滤器

### 表单选择器

#### 13.5.1常规选择器

#### 13.5.3表单选择器

#### 13.5.3表单过滤器

## JQ3

### 基础 DOM 和 CSS 操作

#### 14.1.1 DOM 简介

#### 14.1.3设置元素及内容

#### 14.1.3元属性操作

#### 14.1.4元素样式操作

#### 14.1.5 CSS 方法

### DOM 节点操作

#### 14.2.1创建节点

#### 14.2.3插入节点

#### 14.2.3包裹节点

#### 14.2.4节点操作

### 基础事件

#### 14.3.1绑定事件

#### 14.3.3简写事件

#### 14.3.3复合事件

### 事件对象

#### 14.4.1事件对象

#### 14.4.2冒泡和默认行为

### 高级事件

#### 14.5.1模拟操作

#### 14.5.3命名空间

#### 14.5.3事件委托

#### 14.5.4 on、off 和 one

### 案例：

#### 14.6.1数据操作

## JQ3

### 动画效果

#### 15.1.1显示、隐藏

#### 15.1.2滑动、卷动

#### 15.1.3淡入、淡出

#### 15.1.4自定义动画

#### 15.1.5列队动画方法

#### 15.1.6动画相关方法

#### 15.1.7动画全局属性

### 插件开发

#### 15.2.1插件概述

#### 15.2.2自定义插件

## 面向对象1

### 面向对象的思想；继承、封装、多态。

### Function函数

#### 16.3.1基本介绍

#### 16.3.3函数的声明方式

#### 16.3.3作为值的函数

#### 16.3.4 arguments

#### 16.3.5 apply()和call()方法的应用

#### 16.3.6返回值

#### 16.3.7递归

### 对象与数组

#### 16.3.1 Object介绍

#### 16.3.3 Object类型

#### 16.3.3数组与对象的差异

#### 16.3.4哈希表、关联型数组

### 访问对象属性

#### 16.4.1访问对象属性

#### 16.4.3访问对象方法

#### 16.4.3修改属性与方法

#### 16.4.4删除对象属性与方法

#### 16.4.5使用this值

### 构造函数

#### 16.5.1构造器函数介绍

#### 16.5.3构造器函数的好处

#### 16.5.3 instanceof操作符

#### 16.5.4返回对象的函数

### 回调函数

### 返回函数的函数

### 能重写自己的函数

## 面向对象3

### 变量、作用域及内存

#### 17.1.1基本类型值与引用类型值

#### 17.1.3传递参数

#### 17.1.3检测类型

#### 17.1.4执行环境与作用域

#### 17.1.5块级作用域

#### 17.1.6 var关键字

#### 17.1.7变量查询

#### 17.1.8内存生命周期

#### 17.1.9变量内存及什么是内存泄露

### 基本包装类型

#### 17.3.1基本包装类型概述

#### 17.3.3 Boolean 类型

#### 17.3.3 Number 类型

#### 17.3.4 String 类型

### 创建对象

#### 17.3.1工厂模式

#### 17.3.3构造函数

#### 17.3.3构造函数的规范

#### 17.3.3.1构造函数与工厂模式的不同

#### 17.3.3.3构造函数执行流程

#### 17.3.3.3构造函数this的使用

#### 17.3.3.4构造函数与普通函数的区别

### 原型

#### 17.4.1原型概述

#### 17.4.3\_\_proto\_\_属性

#### 17.4.3原型模式的执行流程

#### 17.4.4 hasOwnProperty()函数、in 操作符

#### 17.4.5 constructor属性

#### 17.4.6原型链

#### 17.4.7原型声明顺序与重写原型

#### 17.4.8组合构造函数+原型模式：

### 继承

#### 15.1.1原型链继承

#### 15.1.3冒充继承

#### 15.1.3组合继承

### 匿名函数和闭包

#### 15.3.1匿名函数

#### 15.3.3闭包优缺点

#### 15.3.3关于this对象

#### 15.3.4模仿块级作用域

#### 15.3.5静态私有变量

#### 15.3.6模块模式

## 数据存储

### cookie

### lostorage,

### sessionstorage

# 第3阶段课程大纲

## RequireJS

### RequireJS简介

### 基本API

#### 1.2.1 define,require

#### 1.2.2 加载配置文件

#### 1.2.3 define的使用

#### 1.2.4 html引入并使用

#### 1.2.5 css引入并使用

### 加载非规范模块

## Node.js的简介

### Node.js简介

#### 2.1.1 什么是Node

#### 2.1.2 Node的用途

#### 2.1.3 Node的特点

#### 2.1.4 Node的历史

#### 2.1.5 Node的核心模块

### Node的安装

#### 2.2.1官网<http://nodejs.org>

#### 2.2.2 Node的安装

#### 2.2.3 Node的基本命令

### Node的fs模块

#### 2.3.1读取文件(同步读取|异步读取)

#### 2.3.2 Node向文件写内容(同步写入|异步写入)

#### 2.3.3读写操作的综合运用：文件的拷贝

#### 2.3.4大文件如何读写(流的概念和使用)

## Node.js

### Node.js 模块

#### 3.1.1 CommonJS

#### 3.1.2 nodejs中的模块化

### Node.js 包和NPM

#### 3.2.1 nodejs包的概念

#### 3.2.2 npm常用命令

### Node.js 常用模块

#### 3.3.1 http模块

#### 3.3.2 url模块

#### 3.3.3 querystring模块

#### 3.3.4 path模块

## Express

4.1 Express基础

#### 4.1.1 什么是Express

#### 4.1.2 官网http://www.expressjs.com.cn

#### 4.1.3 安装express

#### 4.1.4 express的Helloworld

#### 4.1.5 express的脚手架

#### 4.1.6 express的基本路由

#### 4.1.7静态文件

### Express核心

#### 4.2.1路由

#### 4.2.2中间件

#### 4.2.3模板引擎

#### 4.2.4错误处理

### 案例

#### 4.3.1定义接口服务

#### 4.3.2 表单上传

## 原生Ajax

### Ajax概述

#### 5.1.1 什么是ajax

#### 5.1.2 ajax优点

### XMLHttpRequest

#### 5.2.1 创建xhr对象

#### 5.2.2 xhr对象创建的兼容问题

### GET余POST

#### 5.3.1 http协议

#### 5.3.2 get方式请求express接口

#### 5.3.3 post请求express接口

### 封装Ajax

## jQuery中的Ajax和跨域

### Json

#### 6.1.1 JSON 语法

#### 6.1.2 解析和序列化

### JQ中的AJAX

#### 6.2.1 $.load()方法

#### 6.2.2 $.get()和$.post()

#### 6.2.3 $.getScript()和$.getJSON()

#### 6.2.4 $.ajax()方法

#### 6.2.5 表单序列化

### 跨域访问

#### 6.3.1 同源策略

#### 6.3.2 jsonp跨域

#### 6.3.3 CORS跨域

## MongoDB

### MongoDB概念解析

### 数据库基本操作

#### 7.2.1 连接数据库

#### 7.2.2 创建数据库

#### 7.2.3 创建数据库

#### 7.2.4 删除数据库

#### 7.2.5 插入文档

#### 7.2.6 更新文档

#### 7.2.6 删除文档

#### 7.2.7 查询文档

#### 7.2.8 条件操作符

#### 7.2.9 limit和skip

#### 7.2.10 排序

### Nodejs连接数据库

#### 7.3.1 nodejs连接数据库

#### 7.3.2 操作数据库

## MongoDB 案例练习

### 使用Express和MongoDB 实现登陆注册

8.2 实现用户添加和删除功能

## Express项目

### 实现用户登陆功能

### 实现聊天功能

## Angular -01

### AngularJS的入门

#### 10.1.1 AngularJS简介

#### 10.1.2 AngularJS的简介

#### 10.1.3 AngularJS的历史

#### 10.1.4 如何使用AngularJS

#### 10.1.5 AngularJS的优点

#### 10.1.6 MVC设计模式

#### 10.1.7 MVC程序

### 模块

#### 10.2.1 什么是模块

#### 10.2.2 Angular中的模块

### 控制器与作用域

#### 10.3.1 什么是控制器

#### 10.3.2 创建自己的控制器

#### 10.3.3 实例化控制器对象ng-controller

#### 10.3.4 作用域与控制器的关系

### 表达式

#### 10.4.1书写表达式

#### 10.4.2 常用的表达式

### 数据双向绑定的原理

#### 10.5.1 $watch、

#### 10.5.2 $apply

### 过滤器

#### 10.6.1 什么是过滤器

#### 10.6.2 内置过滤器

#### 10.6.3 自定义过滤器

## Angular -02

### 基本指令

#### 11.1.1常用的指令

#### 11.1.2自定义指令

### 服务

#### 11.2.1 什么是服务

#### 11.2.2 常见的内置服务

#### 11.2.2.1 $http服务

#### 11.2.2.2 $location服务

#### 11.2.2.3 $interval服务

#### 11.2.2.4 $timeout服务

### 依赖注入

## Angular -03

12.1 SPA

12.1.1 什么是SPA

12.2 ngRouter

#### 12.2.1 ngRouter指令

#### 12.2.2 创建路由

12.3 ui.route路由

#### 12.3.1 如何使用UI-Router

#### 12.3.2 UI-Router案例

12.4项目框架搭建

## Angular -04 项目练习

13.1 首页

#### 13.1.1 完成首页面开发

#### 13.1.2 使用路由跳转详情页

13.2 详情页

#### 13.2.1 完成详情页编写

#### 13.2.1 实现添加购物车功能

## Angular -05 项目练习

14.1 购物车模块

#### 14.1.1 实现购物车页面编写

#### 14.1.2 实现购物车添加删除功能

## ES6

15.1 ES6简介

#### 15.1.1 ECMAScript和JavaScript

#### 15.1.2 Babel

15.2 let和const

15.3 变量的解构和赋值

15.4 字符串的扩展

15.5 数值的扩展

15.6 函数的扩展

16.7 数组的扩展

16.8 对象的扩展

## ES6

16.1 Set和Map数据结构

16.2 Promise

16.3 async

16.4 Class

16.5 模块化

## ES6 项目练习

17.1 项目练习

17.1.1 使用es6语法进行项目练习

## gulp，webpack

18.1 gulp概述

18.2 gulp的安装

18.3 gulp的使用

#### 18.3.1基本的配置

#### 18.3.2高级的配置

#### 18.3.3 常用的插件

18.4 gulp搭建项目实战

18.5 webpack 讲解

#### 18.5.1 概述

#### 18.5.2 安装

18.5.3 基本使用

# 第4阶段课程大纲

## Vue.js(一)

### 简介

#### 1.1.1 什么是Vue

#### 1.1.2 第一个Vue案例

### 安装

#### 1.2.1 独立引入

#### 1.2.2 NPM方法

### 目录结构

#### 1.3.1 目录解析

### 模板语法

#### 1.4.1 插值

#### 1.4.2 文本,html,属性,表达式

#### 1.4.3 指令,参数,修饰符,用户输入

#### 1.4.4 过滤器,缩写

## Vue.js(一)

### 条件判断v-if

### 条件判断v-else

### 条件判断v-else-if

### 条件判断v-show

#### 2.4.1v-show与v-if的区别

### 循环语句

#### 2.5.1 v-for

#### 2.5.2 v-for迭代对象

#### 2.5.3 v-for迭代整数

### 计算属性

#### 2.6.1 computed

#### 2.6.2 computed vs methods

#### 2.6.3 computed setter

### 样式绑定

#### 2.7.1 class 属性绑定

#### 2.7.2 数组语法

#### 2.7.3 Vue.js style(内联样式)

## 事件处理器与表单

### 事件处理器

### 事件修饰符

### 按键修饰符

### Vue.js

### 事件修饰符

### 输入框

### 复选框

### 单选按钮

### select列表

## 组件与过渡

### 组件

### 全局组件

### 局部组件

### prop

### 动态prop

### 自定义事件

### 单文件组件

### 过渡效果

### 过渡状态

## 自定义指令与路由

### 自定义指令v-focus

### 钩子函数

### 生命周期

### 路由

#### 5.4.1 路由简介

#### 5.4.2 安装

#### 5.4.3 路由实例

#### 5.4.4 路由传值

### vue-cli路由

### 生产环境部署

## Vue.js实战(一)

### 演示项目

### 选择项目

### 分析项目需求

### 项目分工(小组成员选取负责的模块)

### 项目描述(项目周期，项目名称，项目简介，项目所用的技术点，以及项目的产品运营模式和盈利模式)

### 项目框架搭建(以及项目目录结构)

### 6.7 bug跟踪及解决方案表

### 记录小组所遇到的问题和解决方案，以及为解决的问题

### 6.8每日项目跟踪表

### 组长每天记录组员的项目完成进度，以及所需剩余周期，核对项目计划表

## 第七天 Vue.js 项目实战 (二)

### 7.1 检查项目进度表

### 根据学生完成情况，解决学生的共性问题

### 7.2 bug跟踪及解决方案表

### (记录小组所遇到的问题和解决方案，以及为解决的问题)

### 7.3 每日项目跟踪表

### (组长每天记录组员的项目完成进度，以及所需剩余周期，核对项目计划表)

### 7.4 学员总结

## Vue.js 项目实战 (三)

### 8.1检查项目进度表

### (根据学生完成情况，解决学生的共性问题)

### 8.2 bug跟踪及解决方案表

### (记录小组所遇到的问题和解决方案，以及为解决的问题)

### 8.3每日项目跟踪表(组长每天记录组员的项目完成进度，以及所需剩余周期，核对项目计划表)

### 8.4学员总结

## Vue.js 项目实战 (四)

### 9.1 检查项目进度表

#### (根据学生完成情况，解决学生的共性问题)

### 9.2 bug跟踪及解决方案表

#### (记录小组所遇到的问题和解决方案，以及为解决的问题)

### 9.3每日项目跟踪表

### (组长每天记录组员的项目完成进度，以及所需剩余周期，核对项目计划表)

### 9.4学员总结

## Vue.js 项目实战 (五)

### 10.1 检查项目进度表

#### 根据学生完成情况，解决学生的共性问题)

### 10.2 bug跟踪及解决方案表

#### (记录小组所遇到的问题和解决方案，以及为解决的问题)

### 10.3 每日项目跟踪表

### (组长每天记录组员的项目完成进度，以及所需剩余周期，核对项目计划表)

### 10.4 学员总结

## 微信小程序(一)

### 11.1微信小程序起步

11.1.1 微信小程序简介

11.1.2 实例Hello World!

### 11.2 代码构成

11.2.1 JSON配置

11.2.2 WXML模板

11.2.3 WXSS样式

11.2.4 JS逻辑交互

## 微信小程序(二)

### 12.1 工具简介与框架

12.1.1 工具简介

12.1.2 目录结构

### 12.2 配置

12.2.1 app.json 配置项列表

12.2.2 pages

12.2.3 window

12.2.4 tabBar

### 12.3 逻辑层

### 12.4 视图层

## 微信小程序(三)

### 13.1 视图容器

#### 13.1.1 view

#### 13.1.2 scroll-view

#### 13.1.3 swiper

#### 13.1.4 movabile-view

#### 13.1.5 cover-view

### 13.2 基础内容

#### 13.2.1 icon

#### 13.2.2 text

#### 13.2.3 rich-text

#### 13.2.4 progress

### 13.3 表单组件

#### 13.3.1 button, checkbox,form,input

#### 13.3.2 label,picker,picker-view,radio

#### 13.3.3 slider,switch,textarea

### 13.4导航

#### 13.4.1 navigator

## 微信小程序实战(一)

### 14.1演示项目

### 14.2选择项目

### 14.3分析项目需求

### 14.4项目分工

### (小组成员选取负责的模块)

### 14.5项目描述

### (项目周期，项目名称，项目简介，项目所用的技术点，以及项目的产品运营模式和盈利模式)

### 14.6项目框架搭建(以及项目目录结构)

#### 14.7 bug跟踪及解决方案表

#### (记录小组所遇到的问题和解决方案，以及为解决的问题)

#### 14.8 每日项目跟踪表

#### (组长每天记录组员的项目完成进度，以及所需剩余周期，核对项目计划表)

## 微信小程序实战(二)

### 15.1 检查项目进度表

#### (根据学生完成情况，解决学生的共性问题)

### 15.2. bug跟踪及解决方案表

#### (记录小组所遇到的问题和解决方案，以及为解决的问题)

### 15.3每日项目跟踪表

#### (组长每天记录组员的项目完成进度，以及所需剩余周期，核对项目计划表)

### 15.4学员总结

## 微信小程序实战(三)

### 16.1检查项目进度表

#### (根据学生完成情况，解决学生的共性问题)

### 16.2 bug跟踪及解决方案表

#### (记录小组所遇到的问题和解决方案，以及为解决的问题)

### 16.3每日项目跟踪表

#### (组长每天记录组员的项目完成进度，以及所需剩余周期，核对项目计划表)

### 16.4学员总结

## 项目答辩

## 简历制作

# 第5阶段课程大纲

## MongoDB数据库安装

### 完成数据库的安装

### 数据库的环境配置

### 启动数据库

### 关闭数据库

### mongodb增删改查

## MongoDB

### 使用node关联数据库

### 创建集合

### 增删查改

### express+mongodb完成接口搭建

## vue项目1

### 使用Vue.js搭建项目框架

### 使用路由组件构建底部的tabbar

### 整理数据接口

### 开始布局

#### 3.4.1使用插件或者原生js写轮播图

#### 3.4.2搭建轮播图下方的按钮

#### 3.4.3布局首页的相关模块

### 项目分工(小组成员选取负责的模块)

### 3.6项目描述

#### (项目周期，项目名称，项目简介，项目所用的技术点，以及项目的产品运营模式和盈利模式)

### 3.7项目框架搭建

#### (git,gulp,less/sass, 以及项目目录结构)

### 3.8 bug跟踪及解决方案表

#### (记录小组所遇到的问题和解决方案，以及为解决的问题)

### 3.9每日项目跟踪表

#### (组长每天记录组员的项目完成进度，以及所需剩余周期，核对项目计划表)

### 3.10 学员今日成果（项目进度10%）

#### 3.10.1.搭建框架使用vue构造路由。

#### 3.10.2.完成首页布局。

## vue项目2

### 4.1首页的内容区域的构建

#### 4.1.1完成首页的剩余模块

### 4.2完成分类页面的页面构建

#### 4.2.1实现左侧商品分类栏

#### 4.2.2实现分类商品的详情页

#### 4.2.3实现商品的排序功能

#### 4.2.4预留点击购买按钮和点击商品进入详情页的接口

### 4.3 学员今日成果（项目进度40%）

#### 4.3.1完成二级路由下的子页

#### 4.3.2实现基本页面的布局以及调取数据渲染整个页面。

## vue项目3

### 5.1详情页

#### 5.1.1详情页的显示逻辑

#### 5.1.2构建详情页的样式

#### 5.1.3编写页面传值逻辑,实现点击不同的商品,出现的详情页内容不同.

#### 5.1.4实现详情页的商品详情

#### 5.1.5实现收藏和购买的逻辑功能

### 5.2购物城

#### 5.2.1实现购物车的页面搭建

#### 5.2.2实现购物城的地址和收藏列表模块

#### 5.2.3实现商品的价格随数量变化的模块

#### 5.2.4实现其他页面和购物车页面间的数据传递

#### 5.2.5实现阶段付款页面

### 5.3 学员今日成果（项目进度60%）

#### 5.3.1完成二级路由下的页面布局及数据调用。

## vue项目4

### 6.1 注册和登录模块

#### 6.1.1编写接口,实现登录注册

#### 6.1.2实现我的页面的布局

#### 6.1.3实现我的页面的底部轮播

#### 6.1.4实现订单页的模块

### 6.2 商品收藏

#### 6.2.1实现商品收藏的页面布局

#### 6.2.2实现其他页面和收藏页面的数据交互

#### 6.2.3实现店铺收藏模块

#### 6.2.4实现收获地址管理模块

### 6.3 学员今日成果（项目进度80%）

#### 6.3.1完成页面传值实现购物车的操作。

#### 6.3.2完成登录注册页面的操作。

## vue项目5

### 7.1 页面优化

#### 7.1.1增加购物车徽标模块

#### 7.1.2增加加入购物车的动画模块

#### 7.1.3增加下拉刷新模块

#### 7.1.4增加数据的缓存模块

#### 7.1.5增加占位图的模块

#### 7.1.6增加页面跳转的loding图模块

### 7.2 功能添加

#### 7.2.1实现首页的搜索功能

#### 7.2.2实现商品详情的分享功能

#### 7.2.3实现页面的右边返回功能

#### 7.2.4实现定位的地图模块

#### 7.2.5实现用户的头像上传模块

### 7.3 学员今日成果（项目进度100%）

#### 7.3.1完成所有设计页面

#### 7.3.2验证bug并解决bug

## vue自选项目一

## vue自选项目二

## vue自选项目三

## vue自选项目四

## vue自选项目五

## 简历制作一

## 简历制作二

## 面试技巧一

## 模拟面试一

## 模拟面试二

## 模拟面试三