**1.认识Web前端**

**本节内容**

* Web前端开发简介
* 环境搭建
* JavaScript的基本语法及JavaScript对象概述
* jQuery简介
* 实现第一个jQuery程序
* 选择jQuery而不是纯JavaScript的理由

**Web前端开发简介**

* 前端开发是创建Web页面或app等前端界面呈现给用户的过程，通过**HTML，CSS及JavaScript**以及衍生出来的各种技术、框架、解决方案，来实现互联网产品的用户界面交互 。
* 移动互联网带来了大量高性能的移动终端设备以及快速的无线网络，HTML5，node.jS的广泛应用，各类框架类库层出不穷。

**Web前端编程环境搭建**

* vscode
* Microsoft在2015年4月正式宣布了 Visual Studio Code 项目：一个运行于 Mac OS X、Windows和 Linux 之上的，针对于编写现代 Web 和云应用的跨平台源代码编辑器
* 该编辑器支持多种语言和文件格式的编写，截止2019年9月，已经支持了37种语言或文件。
* Firefox
* Mozilla Firefox，中文俗称“火狐”（Fx或fx），是一个自由及开放源代码的网页浏览器，使用Gecko排版引擎，支持多种操作系统，如Windows、Mac OS X及GNU/Linux等。

**下载并安装vscode**

* https://code.visualstudio.com/

**安装VSCode插件**

* Auto Close Tag
* 自动闭合HTML/XML标签
* Auto Rename Tag
* 自动完成另一侧标签的同步修改
* Beautify
* 格式化代码
* Bracket Pair Colorizer
* 给括号加上不同的颜色，便于区分不同的区块，使用者可以定义不同括号类型和不同颜色
* ESLint
* js语法纠错，可以自定义配置，不过配置较为复杂
* eslint-disable-snippets
* 提供忽略eslint的警告、错误等提示的代码片段。
* HTML CSS Support
* 智能提示CSS类名以及id
* HTML Snippets
* 智能提示HTML标签，以及标签含义
* JavaScript(ES6) code snippets
* ES6语法智能提示
* jQuery Code Snippets
* jQuery代码智能提示
* open in browser
* 该插件支持快捷键与鼠标右键快速在浏览器中打开html文件
* Path Intellisense
* 自动提示文件路径，支持各种快速引入文件
* Live Server
* 一款在浏览器预览插件，它还可以实现随着代码的不断改变，实现实时更新
* Vetur
* Vue多功能集成插件
* Atom One Light Theme

**Firefox浏览器和JavaScript的调试方法**

* 为了方便调试，还需安装以下工具
* 下载并安装firefox浏览器
* 网址：http://firefox.com.cn/download/

**JavaScript的基本语法及JavaScript对象概述**

**简介**

* JavaScript 是 Web 的编程语言。
* 所有现代的 HTML 页面都使用 JavaScript。
* JavaScript 是一个脚本语言。
* 它是一个轻量级，但功能强大的编程语言。
* JavaScript 非常容易学。

**JavaScript的重要性**

* 很长时间以来，JavaScript在我们眼里都是编程语言中的二等公民。早先，它就像是胶水一样，它能把HTML应用与样式 粘到一块，可没有人拿它来正正规规地编写程序;
* 直到最近，一些极其重要的扭转局面的技术出现了：jQuery、JSON、Node.js和HTML5。让JavaScript成为了每一个开发人员都知道的语言。

**JavaScript 数据类型**

* 字符串（String）、数字(Number)、布尔(Boolean)、数组(Array)、对象(Object)、空（Null）、未定义（Undefined）。
* JavaScript 拥有动态类型。这意味着相同的变量可用作不同的类型：

**实例**

* var x;               // x 为 undefined  
  var x = 5;           // 现在 x 为数字  
  var x = "John";      // 现在 x 为字符串

**JavaScript 函数**

* 函数是由事件驱动的或者当它被调用时执行的可重复使用的代码块。
* function *functionname*()  
  {  
  *执行代码*  
  }

**JavaScript 事件**

* HTML 事件可以是浏览器行为，也可以是用户行为。
* 事件可以用于处理表单验证，用户输入，用户行为及浏览器动作:
* 页面加载时触发事件
* 页面关闭时触发事件
* 用户点击按钮执行动作
* 验证用户输入内容的合法性
* 等等 ...

**常见的HTML事件**

**创建 JavaScript 对象**

* 通过 JavaScript，能够定义并创建自己的对象。
* 创建新对象有两种不同的方法：
* 定义并创建对象的实例
* 使用函数来定义对象，然后创建新的对象实例

**创建直接的实例**

* person=new Object();  
  person.firstname="John";  
  person.lastname="Doe";  
  person.age=50;  
  person.eyecolor="blue";

**使用对象构造器**

* function person(firstname,lastname,age,eyecolor)  
  {  
  this.firstname=firstname;  
  this.lastname=lastname;  
  this.age=age;  
  this.eyecolor=eyecolor;  
  }

**创建 JavaScript 对象实例**

* var myFather=new person("John","Doe",50,"blue");  
  var myMother=new person("Sally","Rally",48,"green");

**把方法添加到 JavaScript 对象**

* 方法只不过是附加在对象上的函数。

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

</head>

<body>

<script>

function person(firstname,lastname,age,eyecolor){

this.firstname=firstname;

this.lastname=lastname;

this.age=age;

this.eyecolor=eyecolor;

this.changeName=changeName;

function changeName(name){

this.lastname=name;

}

}

myMother=new person("Sally","Rally",48,"green");

myMother.changeName("Doe");

document.write(myMother.lastname);

</script>

</body>

</html>

**JavaScript JSON**

* JSON 是用于存储和传输数据的格式。
* JSON 通常用于服务端向网页传递数据 。
* JSON 英文全称 **J**ava**S**cript **O**bject **N**otation
* JSON 是一种轻量级的数据交换格式。
* JSON是独立的语言
* JSON 易于理解。

**JSON 语法规则**

* 数据为 键/值 对。
* 数据由逗号分隔。
* 大括号保存对象
* 方括号保存数组

{"name":"Runoob", "url":"www.runoob.com"}

"sites":[ {"name":"Runoob", "url":"www.runoob.com"}, {"name":"Google", "url":"www.google.com"}, {"name":"Taobao", "url":"www.taobao.com"} ]

**JSON 字符串转换为 JavaScript 对象**

* 对象由花括号分隔。在括号内部，对象的属性以名称和值对的形式 (name : value) 来定义。属性由逗号分隔：
* var person={firstname:"John", lastname:"Doe", id:5566};
* 上面例子中的对象 (person) 有三个属性：firstname、lastname 以及 id。
* 空格和折行无关紧要。声明可横跨多行：
* var person={  
  firstname : "John",  
  lastname  : "Doe",  
  id        :  5566  
  };

**对象属性**

* 我们通常认为 "JavaScript 对象是键值对的容器"。
* 键值对通常写法为 **name : value** (键与值以冒号分割)。
* 键值对在 JavaScript 对象通常称为 **对象属性**。

**访问对象属性**

* 你可以通过两种方式访问对象属性:
* **实例 1**
* person.lastName;
* **实例 2**
* person["lastName"];

**对象方法**

* 对象的方法定义了一个函数，并作为对象的属性存储。
* 可以使用以下语法创建对象方法：
* *methodName : function() { code lines }*
* 可以使用以下语法访问对象方法：
* *objectName.methodName()*

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

</head>

<body>

<p>创建和使用对象方法。</p>

<p id="demo"></p>

<script>

var person = {

firstName: "John",

lastName : "Doe",

id : 5566,

fullName : function()

{

return this.firstName + " " + this.lastName;

}

};

document.getElementById("demo").innerHTML = person.fullName();

</script>

</body>

</html>

* 思考：如果不加括号person.fullName会输出什么？
* 什么是jQuery
* Jquery是一个优秀的Javascript框架。它是轻量级的js库 ，它兼容CSS3，还兼容各种浏览器（IE 6.0+, FF 1.5+, Safari 2.0+, Opera 9.0+）。使用它，可以更容易的构建有吸引力的交互式网站。
* jQuery能做什么
* 就其核心特性而言，jQuery能够满足下列需求：
* 取得文档中的元素
* 修改页面外观。
* 改变文档的内容
* 响应用户的交互操作
* 为页面添加动态效果
* 无需刷新页面从服务器信息
* 简化常见的JavaScript任务。
* jQuery为什么如此出色
* 利用CSS的优势
* 支持扩展
* 抽象浏览器不一致性
* 总是面向集合
* 将多重操作集于一行
* 开源免费
* 当前流行的 JavaScript 库有:
* Javascript框架使用分布
* 下载jQuery库文件
* jQuery框架实际上是一个JavaScript文件，因此下载jQuery文件，并在网页中引用，就完成了jQuery编程环境的搭建
* 下载网址为http://jquery.com/download/
* 网页中的引用方法
* <script src="jquery.min.js" ></script>
* jQuery选择什么版本 1.x? 2.x? 3.x
* 目前jQuery有三个大版本：  
  **1.x**：兼容ie678,使用最为广泛的，一般项目来说，使用1.x版本就可以了，最终版本：1.12.4 (2016年5月20日)  
  **2.x**：不兼容ie678，很少有人使用，如果不考虑兼容低版本的浏览器可以使用2.x，最终版本：2.2.4 (2016年5月20日)  
  **3.x**：不兼容ie678，只支持最新的浏览器。很多老的jQuery插件不支持这个版本。目前该版本是官方主要更新维护的版本。
* 编写第一个jQuery程序
* 下载：http://jquery.com提供了最新的jQuery框架下载。通常只需下载最小的jQuery包（Minified）即可。我们使用的版本为jquery-1.6.2.js（2.x不支持IE 6,7，8）
* 引用：
* <script src="jquery.js" ></script>
* 将jQuery框架文件导入后，就可以使用jQuery的选择器和各种函数功能了。

<script src="jquery.min.js"></script>

<script>

$(document).ready(function(){

alert("Hello World!");

});

</script>

$(document).ready(function(){

alert("Hello World!");

});

* 也可简化为

$(function(){

alert("Hello World!");

});

* 实现教材案例1.2
* 代码分析

1.查找诗歌文本，这是通过$()函数来完成的。通常该函数需要一个字符串参数，参数中可能包含任何CSS选择符表达式。

2加入新类，.addClass（）方法的作用是不言而喻的，它会将一个CSS类应用到我们选择的页面元素。

注意：无需迭代操作就能为所有诗歌中的节添加这个类，这是jQuery中的**隐式迭代机制**。

3.执行代码，由于Javascript代码**在浏览器初次遇到它们时就会执行**，而在浏览器处理头部时，HTML还不会呈现样式。因此，我们需要将代码延迟到DOM可用时再执行。

* 通过$(document).ready()方法，Jquery支持我们预定在DOM加载完毕后调用某个函数，而不必等待页面中的图像加载。
* 另外可能多次调用$(document).ready()并按照调用顺序执行。
* ready()方法的参数可以是一个已经定义好的函数的引用。如：
* 然而，这个方法也可以接收一个匿名函数
* 这种匿名函数的写作在jQuery中十分方便，特别适合传递那些不会被重用的函数。
* 纯JavaScript 与jQuery
* 即便是像刚才那么简单的任务，如果不用jQuery而是用纯JavaScript的话，也会非常麻烦。