# 2017年9月

## 20

### POST请求

### [URL编码](http://www.cnblogs.com/season-huang/p/3439277.html)

**如果你需要编码整个URL，然后需要使用这个URL，那么用encodeURI。**

**当你需要编码URL中的参数的时候，那么encodeURIComponent是最好方法。**

### [JSON对象和JSON字符串](http://www.haorooms.com/post/js_jsons_h)及相互转化

|  |  |
| --- | --- |
| JSON对象: | var str2 = { "name": "haorooms", "sex": "man" }; |
| JSON字符串: | var str1 = '{ "name": "haorooms", "sex": "man" }'; |
| JSON字符串转换为JSON对象 | var obj = str.parseJSON(); //由JSON字符串转换为JSON对象  var obj = JSON.parse(str); //由JSON字符串转换为JSON对象 |
| 将JSON对象转化为JSON字符串 | var last=obj.toJSONString(); //将JSON对象转化为JSON字符  var last=JSON.stringify(obj); //将JSON对象转化为JSON字符 |

### [跨域](https://itoss.me/2016/12/31/%E8%AF%B4%E8%AF%B4%E8%B7%A8%E5%9F%9F%E9%82%A3%E4%BA%9B%E4%BA%8B%E5%84%BF/)

## 22

### Callback理解

|  |  |
| --- | --- |
| 简单理解： | 把方法callback传给函数foo，在foo执行完以后再执行callback。 |
| 示例 | |
| **var** *foo* = **function** (callback) {  var a = “This is foo”;  alert(“I wrote var a!”);//step 1: I wrote var a!  callback(a); }; **function** *init*() {  //此处foo中的参数是一个函数，即callback,这里的word为foo传过来的a  *foo*(**function** (word) {  alert(a); //step 2: This is foo  }  } | |

## 25

### Window.top

遇到具体问题时再详细学习。

先把一个基本的[链接](http://www.cnblogs.com/haogj/archive/2012/03/28/2422485.html)放在这里。

## 26

### 伪类

### setTimeout()

|  |  |
| --- | --- |
| 语法 | 用于在指定的毫秒数后调用函数或计算表达式。 |
| 参数 | 描述 |
| code | 必需。要调用的函数后要执行的 JavaScript 代码串。 |
| millisec | 必需。在执行代码前需等待的毫秒数。 |
| 示例 | setTimeout("alert('5 seconds!')",5000) |

### 雪碧图

## 27

### 发布补丁

27号晚上peacock补丁要发，中间遇到跨域和swan预览界面获取不到的问题。前者运维配置即可解决，后者是因为主干个分支修改代码不一致产生的问题，还好及时发现，改动不大。

从明天开始参与组件库的开发与维护。

## 28

今天的任务是解决前些天剩下的知识点，熟悉组件库。

不在状态，学习很少。

## 29

再次熟悉了一遍组件库的构建目录，项目中需要的文件怎么产生的。

### Gulp入门教程

这一篇就够了：[点击链接](http://www.cnblogs.com/2050/p/4198792.html)

### 自动安装gulpfile中所有依赖的模块（未成功使用）

在gulpfile.js中定义任务的时候，需要引用不同的模块，这些模块需要使用

*npm install --save-dev* 命令进行安装，非常的繁琐。可以使用一个名为gulpfile-install的模块，使用下面命令进行安装：

|  |
| --- |
| npm install -g gulpfile-install |

安装成功之后就可以在项目根目录（存放gulpfile.js的目录）下，直接执行 gulpfile-install 命令就可以自动根据引用的模块进行安装了，非常方便。

### 路由

# 2017年10月