1. 双向数据绑定的原理

<http://js.jirengu.com/qexiv/1/edit?js,output>

Object.defineProperty()

数据劫持+发布订阅者模式

1. Mvc 与 mvvm之间的区别

Mvvm 是最早由微软提出， Model 用 js 对象表示， view 负责显示， View-model 负责吧model 的数据同步到 View 显示出来， 还负责把View 的修改同步回 Model

1. 闭包的优点

闭包常常用来「间接访问一个变量」。换句话说，「隐藏一个变量」。

我的理解是，闭包就是能够读取其他函数内部变量的函数。

由于在Javascript语言中，只有函数内部的子函数才能读取局部变量，因此可以把闭包简单理解成"定义在一个函数内部的函数"。

所以，在本质上，闭包就是将函数内部和函数外部连接起来的一座桥梁。

闭包会造成内存泄露？

错。

说这话的人根本不知道什么是内存泄露。内存泄露是指你用不到（访问不到）的变量，依然占居着内存空间，不能被再次利用起来。

闭包里面的变量明明就是我们需要的变量（lives），凭什么说是内存泄露？

1. 移动端布局—可以看方应杭录屏
2. 跨域方式

般来说，当一个请求url的协议、域名、端口三者之间任意一个与当前页面地址不同即为跨域。最常见的就是在一个域名下的网页中，调用另一个域名中的资源。

当浏览器报这样的错的时候，就是跨域请求出问题了

这是因为W3C推出的了一个标准----"跨域资源共享"（Cross-origin resource sharing），简称CORS。该标准定义了跨域访问资源时服务器和浏览器怎么通信。通俗讲就是浏览器在发现跨域请求的时候会附加一些头信息和服务器进行沟通，来确定跨域请求通不通过。现在除IE10以下的浏览器都支持这个标准。浏览器会把跨域请求分成两类：简单和非简单请求。

JSONP 服务器代理

1. 事件冒泡
2. Jq事件监听

Jquery事件监听的方式 bind on

Jquery 可以被监听的事件 click

1. 移动端的页面优化
2. Js 基本变量类型

一般来说 typeof 返回值有7种，当然这里指的是浏览器中的String number Boolean object undefined function symbol

基本数据类型有6种：Undefined、Null、布尔值（Boolean）、字符串（String）、数值（Number）、对象（Object）。

这里新添加了一种：Symbol ES6引入了一种新的原始数据类型Symbol，表示独一无二的值。

1. LazyMan url: <http://js.jirengu.com/caneb/4/edit>
2. js 获取背景色 <http://js.jirengu.com/pesew/1/edit?css,js,console,output>
3. 饥人谷笔试1 flatten <http://js.jirengu.com/zacox/2/edit>
4. 自己实现 reduce <http://js.jirengu.com/huyaq/1/edit>
5. Css 实现 modal 蒙层 <http://js.jirengu.com/wuqof/1/edit?html,css,output>

#checkbox:checked~ .modal

1. <http://js.jirengu.com/naqoj/1/edit?css,output>
2. 自己实现 map <http://js.jirengu.com/tegam/1/edit>
3. 自己实现 find <http://js.jirengu.com/hacop/1/edit?js,console>
4. 自己实现 filter <http://js.jirengu.com/juxez/1/edit?js,console>
5. 自己实现 invoke
6. 自己实现 parseQuery <http://js.jirengu.com/sosad/2/edit>

<https://zhuanlan.zhihu.com/p/29135388>

<http://blogread.cn/it/wap/article/7847?f=wb_news>

<https://segmentfault.com/a/1190000013565024>

<https://zhuanlan.zhihu.com/p/22476797?refer=FEXFJ>

<https://juejin.im/post/5aa00229f265da239b40fc02>

<https://juejin.im/post/5aa001306fb9a028c3685a79>

<https://juejin.im/post/5a0bef0551882554bd509865>

<https://juejin.im/post/5815f4abbf22ec006893b431>

React-reduce