



# 前端基础 (JS)

蓝鲸智云讲师

# 前端基础 ( JS )

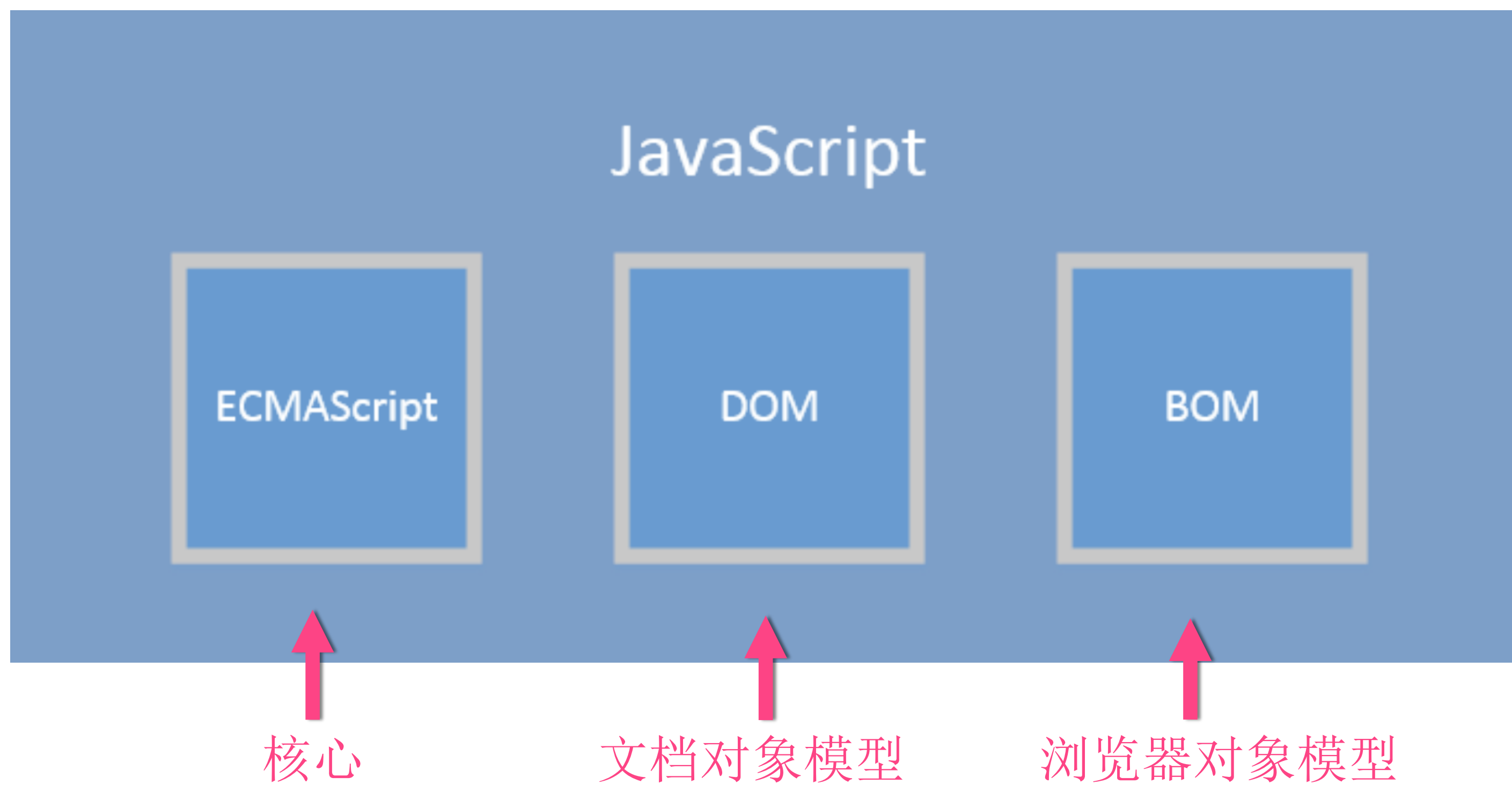
- JS 基础知识
- JS 特性
- DOM 编程
- AJAX 异步
- Chrome 调试工具简介



# 前端基础 (JS)

## 什么是 JavaScript

- JS 起源于 Netscape 公司的 LiveScript 语言
- JS 是一种基于对象和事件驱动的客户脚本语言
- JS 最初的设计是为了检验 HTML 表单输入的正确性



## 如何在页面上引入 JavaScript

- 利用 `<script>` 标签引入外部 JS 文
- 在 `<script>` 标签内部添加 JS 代码
- 在 HTML 标签中插入 JS 代码

```
<!DOCTYPE html>
<!DOCTYPE html>
<html>
  <html>
  <head>
    <head>
    <meta charset="utf-8">
    <meta charset="utf-8">
    <title>index</title>
    <title>index</title>
  </head>
  <body>
    <body>
    <div>Hello World</div>
    <div id="test" onclick="javascript:alert('hello world');">
      <script>
        <script src="hello.js"></script>
        Hello World
      </script>
    </div>
  </body>
</html>
</html>
```

## 基本类型 按值访问

- Number, String, Boolean, Null, Undefined,  
Symbol

## 引用类型 按引用访问

- Object(Object, Array, Date, RegExp, Function)

## 类型检测

- typeof
- instanceof

检测一个变量是不是基本数据类型

判断某个构造函数的 prototype 属性所指向的对象是否存在于另外一个要检测对象的原型链上（判断一个引用类型的变量具体是不是某种类型的对象）



## Null

null 是 js 保留字

`typeof null === 'object'`

null 不是 window 的属性

null 参与数值运算时自动转换为 0

## Undefined

undefined 不是 js 保留字

`typeof undefined === 'undefined'`

undefined 是 window 的属性

undefined 参与数值运算时结果为 NaN

```
null == undefined // => true
null !== undefined // => true
```

- 类 C、C++、Java 等
- 区分大小写
- 使用 Unicode 字符集
- .....

## 变量

- 变量是对“值”的具名引用。变量就是为“值”起名，然后引用这个名字，就等同于引用这个值。变量的名字就是变量名。

```
var a = 1
```

```
var A  
A = 1
```

## 条件语句

- JavaScript 提供 if 结构和 switch 结构，完成条件判断，即只有满足预设的条件，才会执行相应的语句。

```
if (a = 1) {  
    a = a + 1  
}
```

```
if (a = 1) {  
    a = a + 1  
} else {  
    a = a - 1  
}
```

```
if (a = 1) {  
    a = a + 1  
} else if (a = 2) {  
    a = a + 2  
} else {  
    a = a - 1  
}
```

```
switch (a) {  
    case 1:  
        a = a + 1  
        break  
    case 2:  
        a = a + 2  
        break  
    default:  
        a = a - 1  
}
```

## 循环语句

- 循环语句用于重复执行某个操作，它有多种形式。

```
var i = 0
while (i < 100) {
    console.log('i: ' + i)
    i = i + 1
}
```

```
var i = 0
do {
    console.log('i: ' + i)
    i++
} while(i < 100)
```

```
var i = 0
for (i = 0; i < 100; i++) {
    console.log('i: ' + i)
}
```

## 函数 function

- 函数是一段可以反复调用的代码块。函数还能接受输入的参数，不同的参数会返回不同的值。

```
var add = function (x + y) {  
    return x + y  
}
```

```
function add(x, y) {  
    return x + y  
}
```

```
var add = new Function(  
    'x',  
    'y',  
    'return x + y'  
)
```

- 作用域
- 函数
- this
- 原型和继承 ( \* )
- 闭包 ( \* )
- .....

## 作用域

- 局部变量的优先级比全局变量高

```
var w = 1

function foo() {
    var w = 100
    alert(w)
}

foo()
alert(w)
```



## 作用域

- 没有块级作用域

```
var x = 1
alert(x)

if (true) {
    var x = 2
    alert(x)
}

alert(x)
```

## 作用域

- 未使用 var 关键字定义的变量都是全局变量

```
function foo() {  
    www = 'hello'  
}
```

```
foo()
```

```
alert(www)
```

```
alert(window.www)
```

## 作用域

- 词法作用域

```
var value = 1      value=1
function foo() {  function foo() {
    console.log(value)  echo $value
}
function bar() {  function bar() {
    var value = 2      local value=2
    foo()              foo
}
bar()                 bar
```

## 作用域

- Variable Hoisting，在代码运行前，函数声明和变量定义通常会被解释器移动到其所在作用域的最顶部

```
function foo() {  
    bar()  
    var x = 1  
}  
  
function foo() {  
    var x  
    bar()  
    x = 1  
}
```



Variable Hoisting 只是定义上升，赋值并不会上升

## 函数

- 自执行函数

```
~function () {  
    alert('hello')  
}()
```

```
+function () {  
    alert('hello')  
}()
```

```
(function () {  
    alert('hello')  
})()
```

```
!function () {  
    alert('hello')  
}()
```

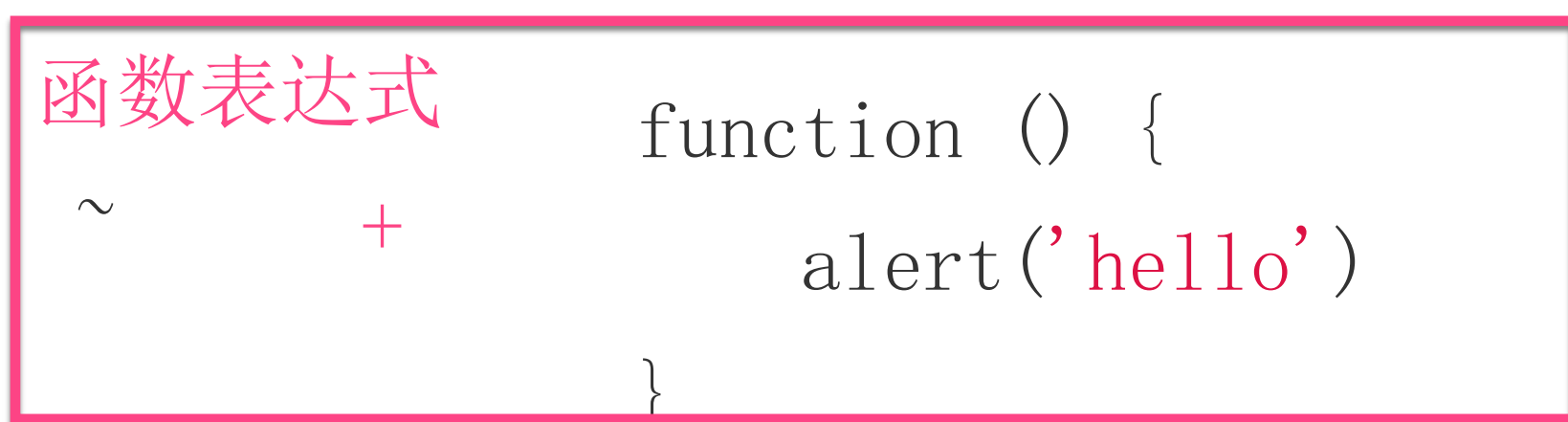
```
-function () {  
    alert('hello')  
}()
```

```
(function () {  
    alert('hello')  
})()
```

## 函数

- 自执行函数

```
~function () {  
    alert('hello')  
}()
```



↑  
运算符

↑  
函数声明

+ ()  
↑  
执行运算符

## this

- this 是 JS 语言的一个关键字。它代表函数运行时，自动生成的一个内部对象，只能在函数内部使用。

```
function test() {  
    this.x = 1  
    alert(this.x)  
}  
test()
```

```
var y = 1  
function test1() {  
    this.y = 0  
}  
test1()  
alert(y)
```

```
function test() {  
    alert(this.x)  
}  
var o = {  
    x: 5,  
    func: test  
}  
test()  
o.func()
```



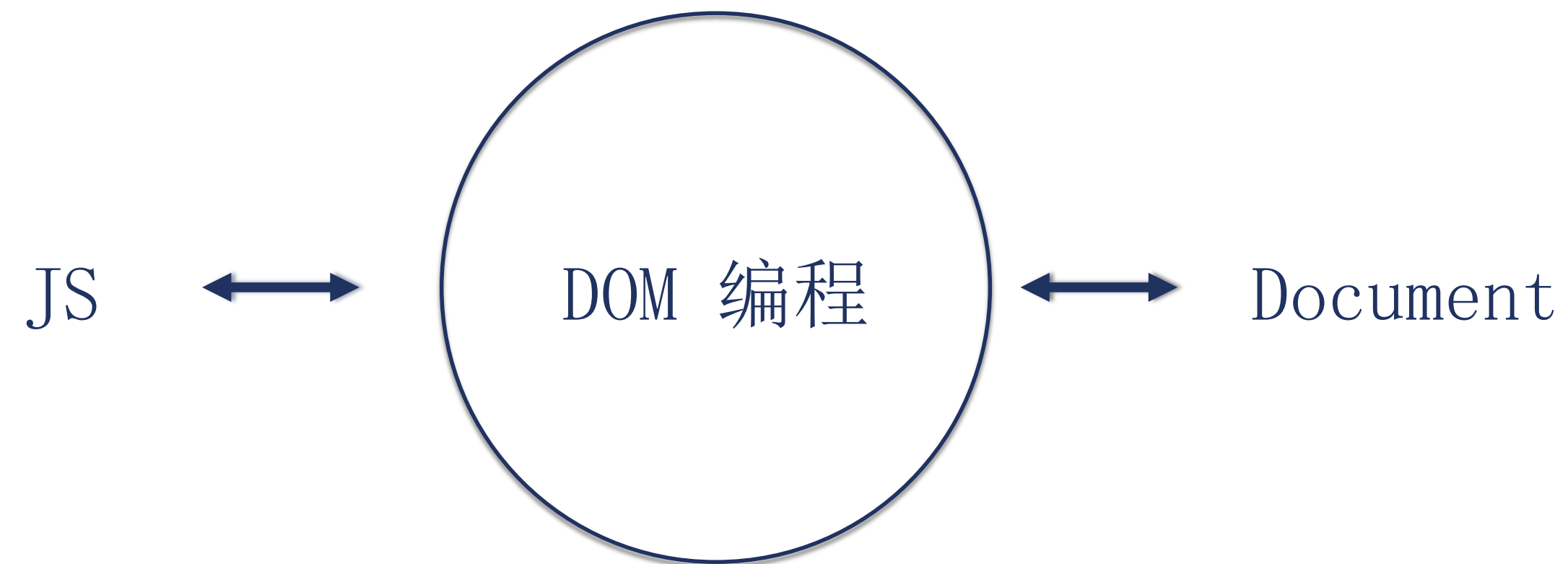
***"Any application that can be written in JavaScript, will eventually be written in JavaScript."***

*Atwood's Law, Jeff Atwood. 2007*



## 什么是 DOM 编程

- DOM 编程就是使用 W3C 定义的 API (Application Program Interface) 来操作 HTML 文档（此处不局限于 HTML，亦可操作 XHTML、XML 等），使用户可以与进行页面交互。



## 重要的数据类型

- document, element, nodeList
- attribute, nodeNameMap
- .....

## 常用核心接口

- `document.getElementById(id)`
- `document.getElementsByTagName(name)`
- `document.createElement(name)`
- `parentNode.appendChild(node)`
- `element.setAttribute()` / `element.getAttribute()`
- .....

## jQuery

- jQuery 是一个高效、精简并且功能丰富的 JavaScript 工具库。它提供的 API 易于使用且兼容众多浏览器，这让诸如 HTML 文档遍历和操作、事件处理、动画和 Ajax 操作更加简单。

<https://www.jquery123.com>

## jQuery 常用方法

- `$(selector).val() / $(selector).val(val)`
- `$(selector).text() / $(selector).text(text)`
- `$(selector).html() / $(selector).html(content)`
- `$(selector).attr(key, val) / $(selector).attr(key)`
- `$(selector).addClass(class)`
- `$(selector).removeClass(class)`
- `$(selector).css(prop, val) / $(selector).css(prop)`
- `$(selector).parent() / $(selector).children()`
- `$(selector).prev() / $(selector).next()`
- `$(selector).siblings()`
- .....

## 基本概念

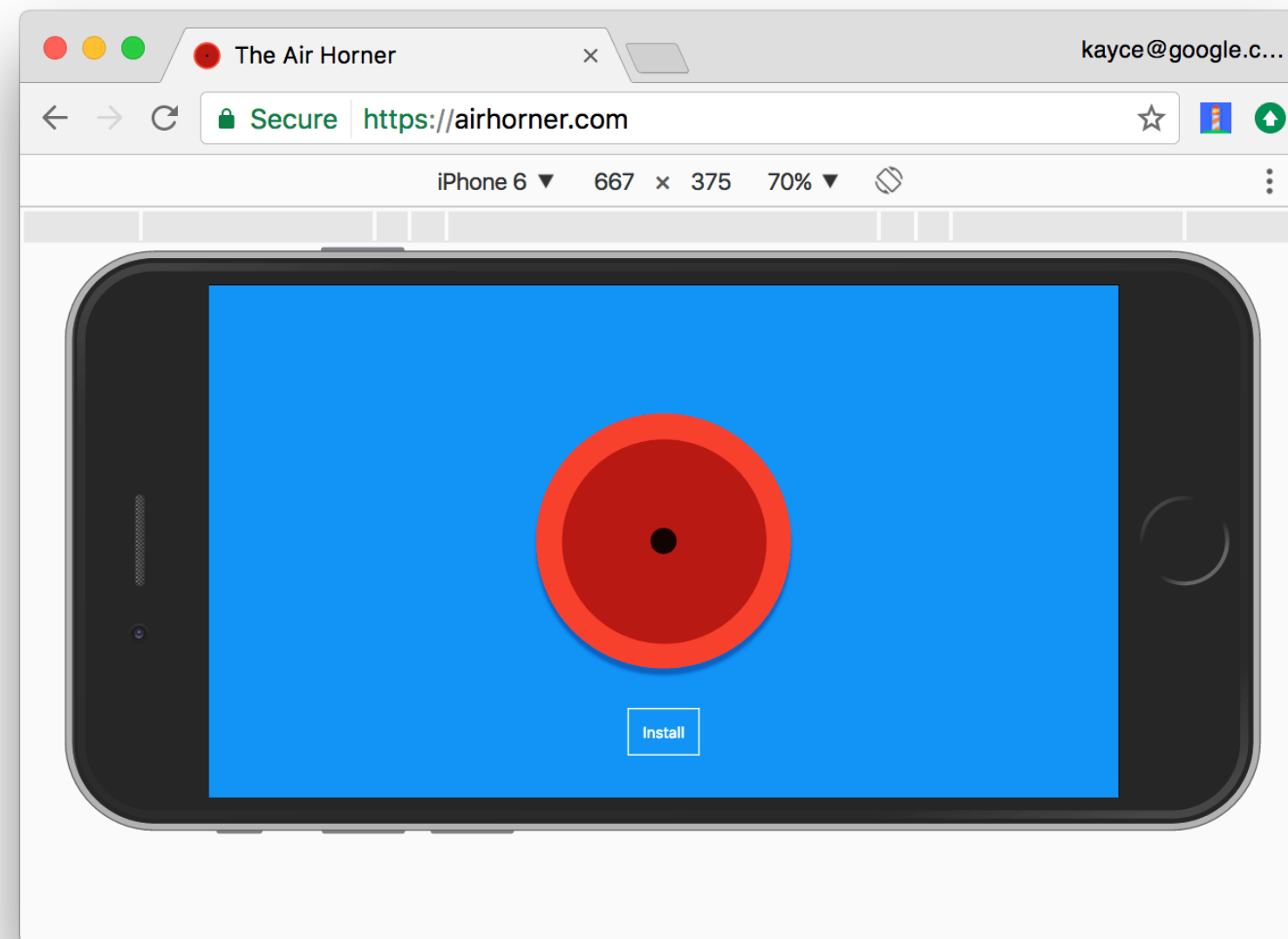
- Asynchronous JavaScript and XML。简单来说就是通过 XMLHttpRequest 对象来向服务器发异步请求，从服务器获得数据，然后用 JS 来操作 DOM 从而实现更新页面。

## jQuery 发送 AJAX 异步请求

```
$.ajax( {  
    url: 'url',  
    data: {},  
    type: 'post',  
    dataType: 'json',  
    success: function (data) {  
    },  
    error: function () {  
    }  
})
```

## 面板

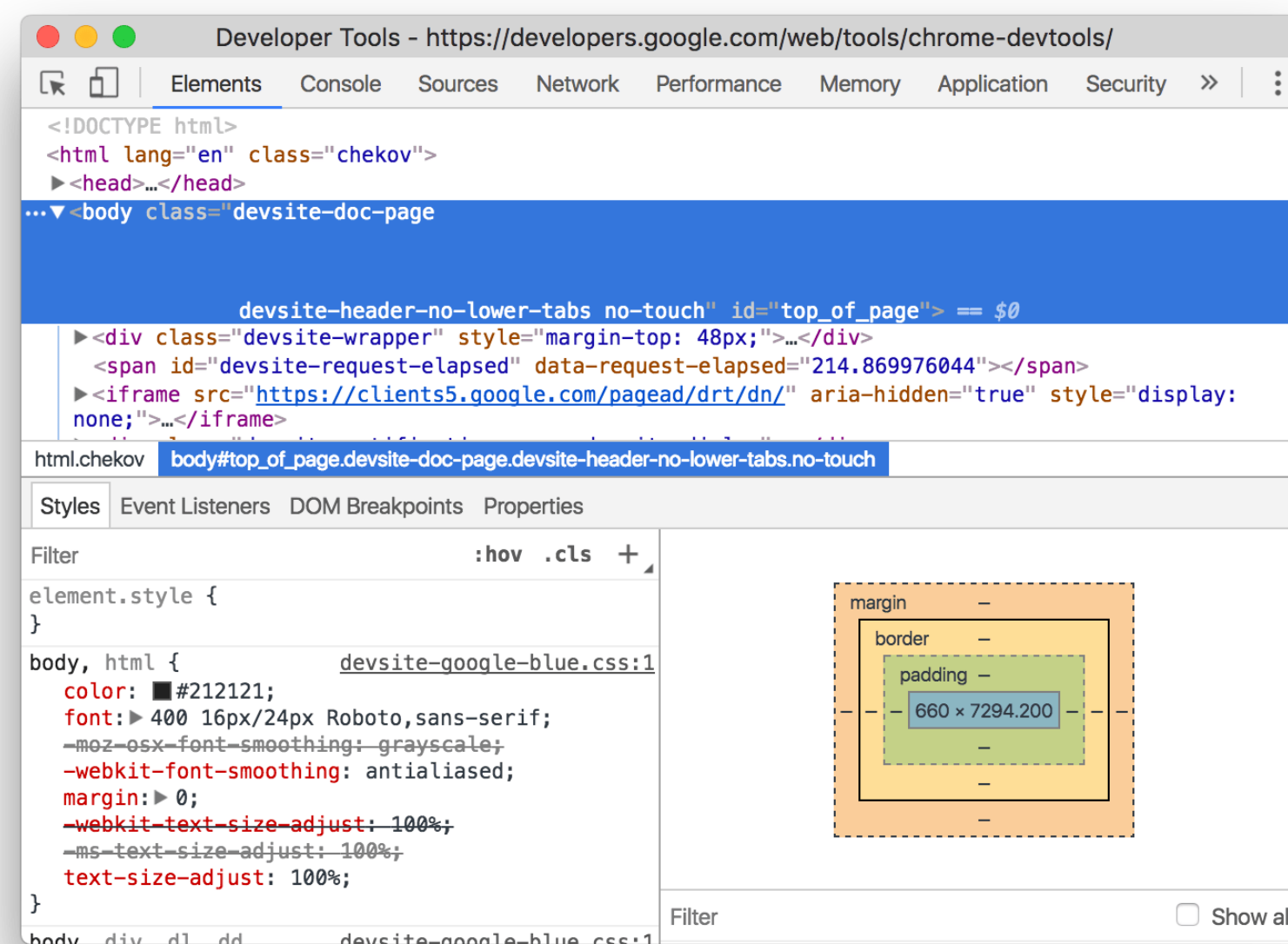
- 设备模式





## 面板

- 元素面板



## 面板

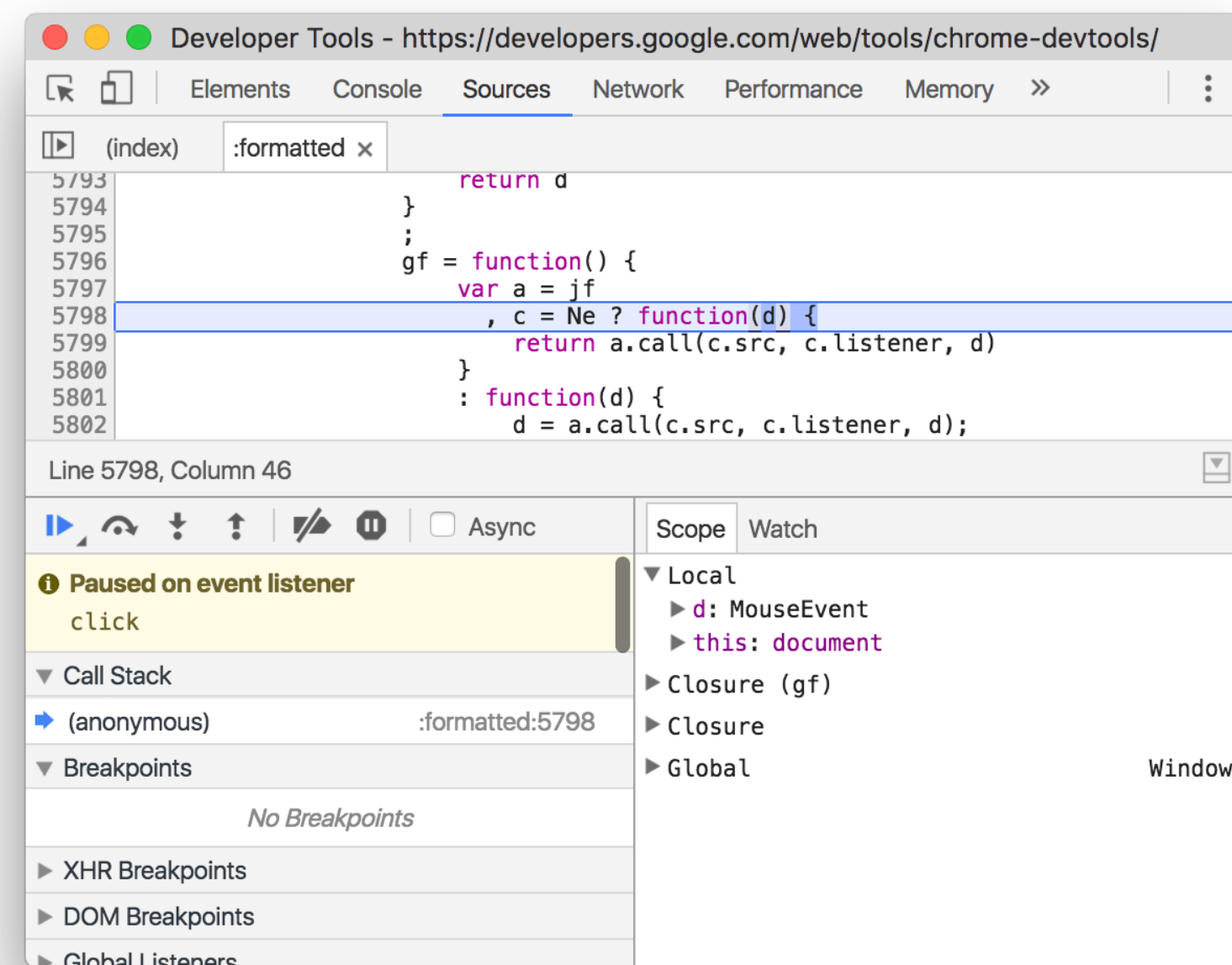
- 控制台面板



The screenshot shows the Chrome Developer Tools interface with the Console panel active. The console displays a log message: `console.log('%c wow, such blue, much nice', 'color: blue');` followed by the output `wow, such blue, much nice` in blue text. The source is identified as `VM138:1`. The console also shows `< undefined` and a prompt `> |`.

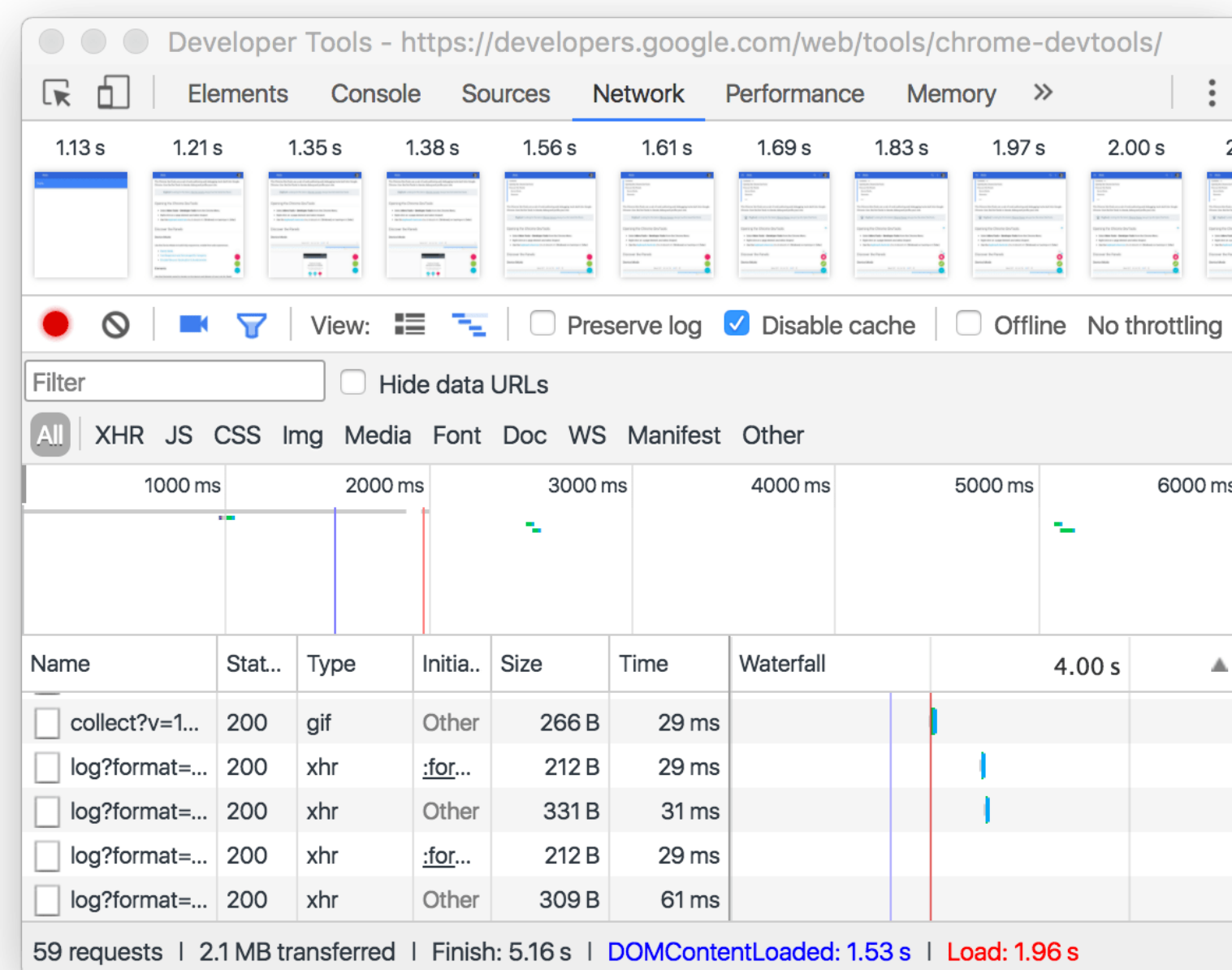
## 面板

- 源代码面板



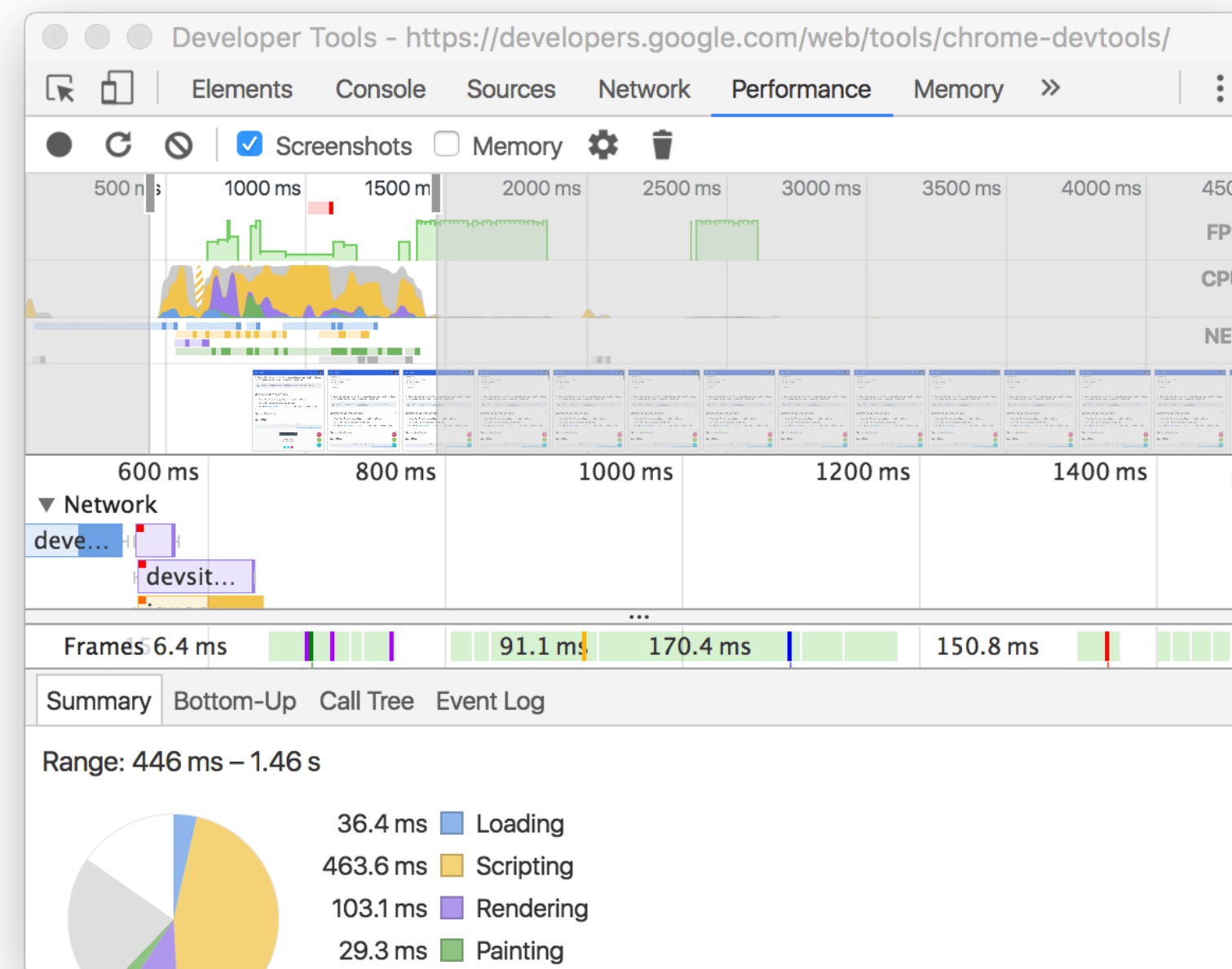
## 面板

- 网络面板



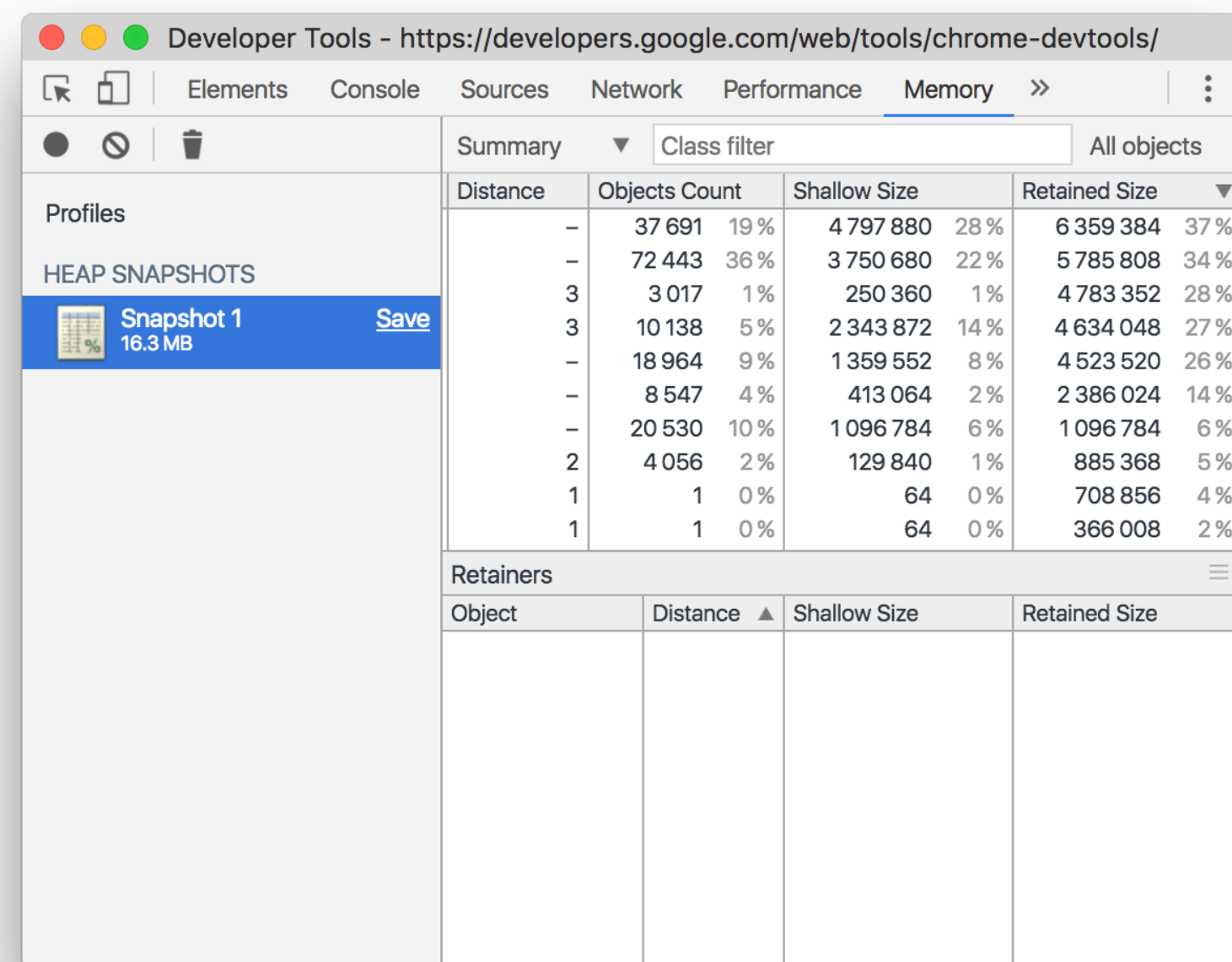
## 面板

- 性能面板



## 面板

- 内存面板



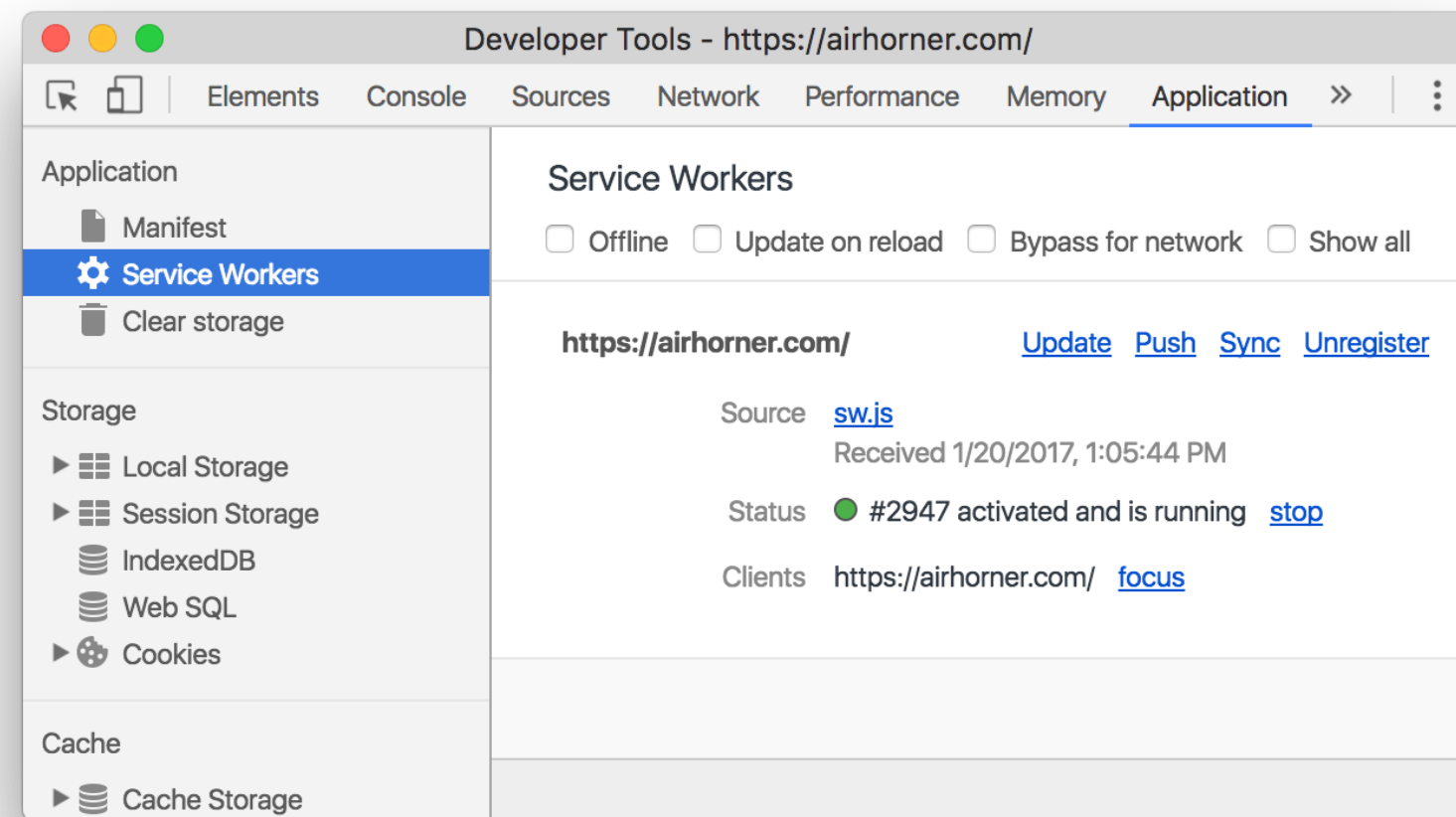
The screenshot shows the Chrome DevTools Memory panel. The 'Summary' tab is active, displaying a table of heap snapshot data. The table has columns for Distance, Objects Count, Shallow Size, and Retained Size. The data is as follows:

| Distance | Objects Count | Shallow Size  | Retained Size |
|----------|---------------|---------------|---------------|
| -        | 37 691 19%    | 4 797 880 28% | 6 359 384 37% |
| -        | 72 443 36%    | 3 750 680 22% | 5 785 808 34% |
| 3        | 3 017 1%      | 250 360 1%    | 4 783 352 28% |
| 3        | 10 138 5%     | 2 343 872 14% | 4 634 048 27% |
| -        | 18 964 9%     | 1 359 552 8%  | 4 523 520 26% |
| -        | 8 547 4%      | 413 064 2%    | 2 386 024 14% |
| -        | 20 530 10%    | 1 096 784 6%  | 1 096 784 6%  |
| 2        | 4 056 2%      | 129 840 1%    | 885 368 5%    |
| 1        | 1 0%          | 64 0%         | 708 856 4%    |
| 1        | 1 0%          | 64 0%         | 366 008 2%    |

Below the table, the 'Retainers' section is visible, showing a table with columns for Object, Distance, Shallow Size, and Retained Size. The table is currently empty.

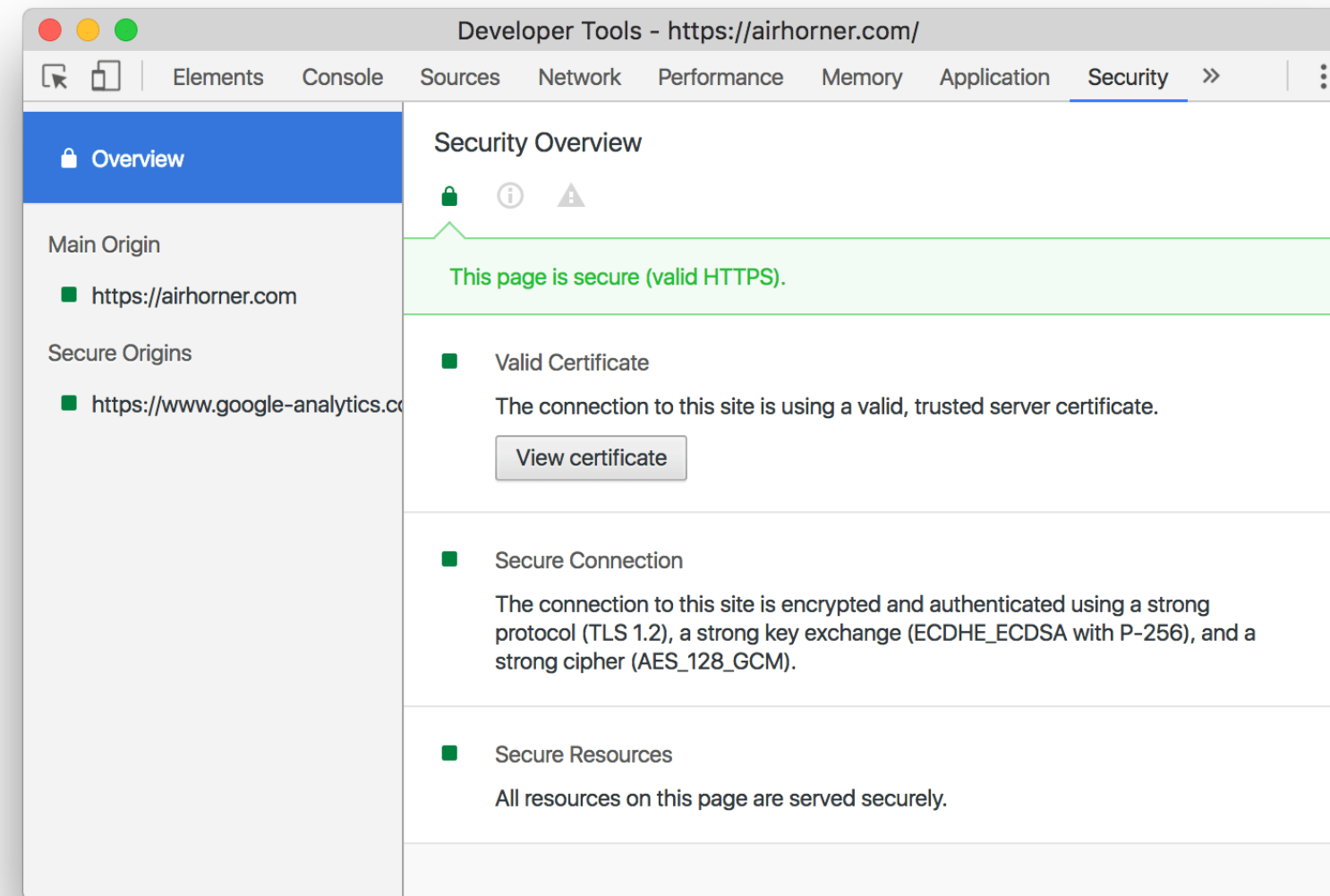
## 面板

- 应用面板



## 面板

- 安全面板







# 前端基础 ( JS ) 示例

- 表单提交（原生提交方式 + ajax 提交方式）

|      |                      |
|------|----------------------|
| 组织名称 | <input type="text"/> |
| 负责人  | <input type="text"/> |
| 参评人  | <input type="text"/> |

提交

- 表单提交（原生提交方式）

```
<form action="form-action-url" method="get">
  <div class="bk-form-item">
    <label class="bk-label">组织名称</label>
    <div class="bk-form-content"><input type="text" class="bk-form-input" name="name"></div>
  </div>
  <div class="bk-form-item">
    <label class="bk-label">负责人</label>
    <div class="bk-form-content"><input type="text" class="bk-form-input" name="people"></div>
  </div>
  <div class="bk-form-item">
    <label class="bk-label">参评人</label>
    <div class="bk-form-content"><textarea class="bk-form-textarea" name="peoples"></textarea></div>
  </div>
  <div class="bk-form-item bk-form-action">
    <div class="bk-form-content"><input type="submit" value="submit" /></div>
  </div>
</form>
```

- form 标签常见属性

| 属性               | 值  | 描述                          |
|------------------|--|-----------------------------|
| accept           | MIME_type  | HTML 5 中不支持。                |
| accept-charset   | charset_list   | 规定服务器可处理的表单数据字符集。           |
| action           | URL  | 规定当提交表单时向何处发送表单数据。          |
| autocomplete(H5) | on/off   | 规定是否启用表单的自动完成功能。            |
| enctype          | application/x-www-form-urlencoded<br>multipart/form-data<br>text/plain | 规定在发送表单数据之前如何对其进行编码。        |
| method           | get/post   | 规定用于发送 form-data 的 HTTP 方法。 |
| name             | form_name  | 规定表单的名称。                    |
| novalidate(H5)   | novalidate   | 如果使用该属性，则提交表单时不进行验证。        |
| target           | _blank/_self/_parent/_top/frameName                                    | 规定在何处打开 action URL。         |

- 表单提交（ajax 提交方式）

```
$('#submit').click(function () {  
    var params1 = $('#myform').serialize()  
    var params2 = $('#myform').serializeArray()  
    console.log(params1) // name=hiewang&people=huacao&peoples=huacao,hiewang  
    console.log(params2) // [Object, Object, Object]  
    $.ajax( {  
        type: 'GET',  
        url: 'form-action-url',  
        data: params1,  
        success: function (msg) {  
            console.log( 'success: ' + msg)  
        }  
    })  
})
```



# 前端基础 (JS) 课后作业

- 奖项申报页面

The screenshot shows a web interface for award applications. It features a top navigation bar with '奖项申报', '系统管理', '个人中心', and '蓝鲸应用平台组公共开发号'. A left sidebar contains '我的申报' and '我的审核'. The main content area is divided into two columns. The left column, titled '奖项信息', contains an envelope icon, the award name 'ceshi11112', a '评奖条件' section with an image of two silhouettes, and a table of details. The right column, titled '申报信息', contains a form for '申报人/团队\*', a text area for '事迹介绍\*', and an '附件' section with a '选择文件' button. At the bottom of the right column are '我要申报' and '取消' buttons. Red arrows point to the left sidebar with the label '左右布局' and to the '我要申报' button with the label '表单提交, 支持附件选择'.

奖项申报

系统管理 个人中心 蓝鲸应用平台组公共开发号

我的申报

我的审核

首页 / 个人中心 / 奖项申报

### 奖项信息

奖项名称 ceshi11112

评奖条件

|      |                                |
|------|--------------------------------|
| 审核人  | 337228499 1334793794 739990640 |
| 状态   | 生效                             |
| 奖项级别 | 中心级                            |
| 所属组织 | 第一组织                           |
| 开始日期 | 2018-06-01                     |
| 结束日期 | 2018-07-01                     |

### 申报信息

申报人/团队\*


事迹介绍\*

附件  未选择任何文件

左右布局

表单提交, 支持附件选择

- 我的申报页面

奖项申报 系统管理 ▾ 个人中心 ▾ 蓝鲸应用平台组公共开发号 

**我的申报** | 我的审核

首页 / 个人中心 / 我的申报

搜索选项

奖项名称  申报日期  申报状态


申报列表

| 所属中心 | 奖项名称  | 奖项状态 | 申报人/团队    | 申报状态 | 申报时间                | 操作                 |
|------|-------|------|-----------|------|---------------------|--------------------|
| 第一组织 | 测试奖项1 | 生效   | werwr     | 审核中  | 2018-01-18 09:55:14 | <a href="#">编辑</a> |
| 第一组织 | 测试奖项  | 生效   | werarew   | 审核中  | 2018-01-18 09:55:07 | <a href="#">编辑</a> |
| 第一组织 | 第二奖   | 生效   | werwr     | 审核中  | 2018-01-18 09:54:59 | <a href="#">编辑</a> |
| 第一组织 | 第一奖   | 生效   | werwrw    | 审核中  | 2018-01-18 09:54:51 | <a href="#">编辑</a> |
| 第一组织 | ABC   | 生效   | aaaaaa    | 审核中  | 2018-01-18 09:54:38 | <a href="#">编辑</a> |
| 第一组织 | 第三奖项  | 生效   | selinaguo | 审核中  | 2018-01-18 09:53:50 | <a href="#">编辑</a> |

1



- 我的审核页面

奖项申报 系统管理 ▾ 个人中心 ▾ 蓝鲸应用平台组公共开发号 

[我的申报](#)  
**[我的审核](#)**

[首页](#) / [个人中心](#) / [我的审核](#)

审核列表

| 所属中心 | 申报奖项  | 申报人/团队    | 申报状态 | 申报时间                | 最后更新时间              | 操作人        | 操作  |
|------|-------|-----------|------|---------------------|---------------------|------------|---|
| 第一组织 | 测试奖项1 | werwr     | 审核中  | 2018-01-18 09:55:14 | 2018-01-18 09:55:14 | 1796624909 | <a href="#">✓ 通过</a> <a href="#">✕ 驳回</a> |
| 第一组织 | 测试奖项  | werarew   | 审核中  | 2018-01-18 09:55:07 | 2018-01-18 09:55:07 | 1796624909 | <a href="#">✓ 通过</a> <a href="#">✕ 驳回</a> |
| 第一组织 | 第二奖   | werwr     | 审核中  | 2018-01-18 09:54:59 | 2018-01-18 09:54:59 | 1796624909 | <a href="#">✓ 通过</a> <a href="#">✕ 驳回</a> |
| 第一组织 | 第一奖   | werwrw    | 审核中  | 2018-01-18 09:54:51 | 2018-01-18 09:54:51 | 1796624909 | <a href="#">✓ 通过</a> <a href="#">✕ 驳回</a> |
| 第一组织 | ABC   | aaaaaa    | 审核中  | 2018-01-18 09:54:38 | 2018-01-18 09:54:38 | 1796624909 | <a href="#">✓ 通过</a> <a href="#">✕ 驳回</a> |
| 第一组织 | 第三奖项  | selinaguo | 审核中  | 2018-01-18 09:53:50 | 2018-01-18 09:53:50 | 1796624909 | <a href="#">✓ 通过</a> <a href="#">✕ 驳回</a> |
| 第一组织 | 第一奖   | jere      | 未通过  | 2018-01-15 20:26:31 | 2018-01-15 20:26:31 | 739990640  | ----                                      |
| 第一组织 | 测试奖项  | Test      | 审核中  | 2018-01-06 16:01:31 | 2018-01-06 16:01:31 | 236779280  | <a href="#">✓ 通过</a> <a href="#">✕ 驳回</a> |
| 第一组织 | 第三奖项  | gz        | 已获奖  | 2018-01-01 23:20:51 | 2018-01-01 23:20:51 | 916236633  | ----                                      |
| 第一组织 | 第一奖   | CG        | 已获奖  | 2018-01-01 23:17:43 | 2018-01-01 23:17:43 | 337228499  | ----                                      |

1 2 [下一页](#)

## Django基础（一）

- MVC 介绍
- Django-Settings
- Django-URLConf
- Django-Views
- Request、Response 对象介绍
- PyCharm 调试

# 腾讯问卷课后调研





# Thanks



蓝鲸智云公众号



蓝鲸高校版交流群