[一、js 2](#_Toc12986)

[1、数组操作 2](#_Toc15625)

[2、Json 3](#_Toc23065)

[3、 base64加密 3](#_Toc12771)

[二、css 3](#_Toc13504)

[1、box-shadow/text-shadow 3](#_Toc2022)

[2、变换 4](#_Toc19441)

[3、渐变 5](#_Toc5456)

[4、动画 5](#_Toc20195)

[5、自适应字体大小rem 6](#_Toc7815)

[6、超出文本显示圆的 7](#_Toc16944)

[7、水平垂直居中 7](#_Toc18534)

[8、弹出层时，阻止页面滑动 9](#_Toc10086)

[9、 修改input placeholder字体颜色 9](#_Toc9714)

[三、功能 9](#_Toc18008)

[1、分页加载/图片懒加载(lazyImg.js) 9](#_Toc1266)

[2、 判断浏览器类别 11](#_Toc19956)

**一、js**

**1、数组操作**

var arrObj = [];

// 将一个值(或多个值)添加到数组的结尾，并返回新数组的长度

var length = arrObj.push(item1, ...);

// 删除数组最后一个值，并返回该值

var pop = arrObj.pop();

// 将一个值(或多个值)添加到数组的头部，并返回新数组的长度

var unshift = arrObj.unshift(item1, ...);

// 删除数组第一个值，并返回该值

var shift = arrObj.shift();

// 1、item 存在：将一个值(或多个值)替换数组指定位置(index)的长度(length)，若length=0，则在指定位置(index)插入item，并返回删除的数组或""

// 2、item 不存在：删除数组指定位置(index)后的length个值，并返回删除的数组

var splice = arrObj.splice(index, length, item);

// 截取并返回数组从begin到end(不包括end)的部分数组值，若end省略，则返回begin 后面的所有值

// arrObj.slice(0); 返回数组的拷贝数组，注意是一个新的数组，不是指向

var slice= arrObj.slice(begin, end);

// 合并数组或字符串，并返回新的数组

// arrObj.concat(); 返回数组的拷贝数组，注意是一个新的数组，不是指向

var concat = arrObj.concat(item, ...);

// 反转数组

var reverse = arrObj.reverse();

// 排序数组

var sort = arrObj.sort();

// 将数组的值用 str 进行拼接，并返回字符串

var join= arrObj.join(str);

**2、Json**

var jsonObj = {};

// 将json 对象转为字符串

var str = JSON.stringify(jsonObj);

// 将字符串转为json

var jsonObj = JSON.parse(str);

// 删除指定key 的值

var deleteO = delete jsonObj[key];

1. **base64加密**

加密：window.btoa

解密：window.atob

**二、css**

**1、****box-shadow/text-shadow**

参数: x-shadow --x方向偏移量

y-shadow --y方向偏移量

blur --模糊的宽度

spread --阴影的尺寸

color --阴影颜色

inset --内部

用法:

text-shadow: 5px 6px 7px #000,

0 0 20px #df76ec,

0 0 50px #ff006b,

0 0 80px #000;

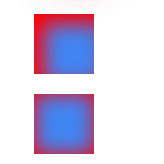


参数可以是一组或多组值，用逗号分开

box-shadow内部阴影：

box-shadow: inset 10px 10px 30px #ff0000;

box-shadow: inset 0 0 30px #ff0000;



其中x,y 的偏移量，可以为负数。

**2、变换**

-webkit-transform: translate(x, y);

-moz-transform: translate(x, y);

-ms-transform: translate(x, y);

transform: translate(x, y);

transform的参数可以是translate(x, y)、scale(x, y)、rotate(n)、skew(x, y);一个或多个

如：transform: translate(x, y) skew(x, y);

位移(px)：translate3d(x, y, z);

缩放(无单位(倍数))：scale3d(x, y, z);

旋转(deg)：rotate3d(x, y, z, a) --x, y, z 旋转的位置，a旋转角度

斜切(deg)：skew(x-a, y-a)

透视(无单位)：perspective(n)

矩阵变换(无单位)：matrix(n,n,n,n,n,n)

matrix3d(n,n,n,n,n,n,n,n,n,n,n,n,n,n,n,n)

transform-style属性：

必需与transform同用，

transform-style: flat|preserve-3d;

flat --子元素将不保留其 3D 位置

preserve-3d --子元素将保留其 3D 位置

transition: [transition-property](http://www.w3school.com.cn/cssref/pr_transition-property.asp" \o "CSS3 transition-property 属性) [transition-duration](http://www.w3school.com.cn/cssref/pr_transition-duration.asp" \o "CSS3 transition-duration 属性) [transition-timing-function](http://www.w3school.com.cn/cssref/pr_transition-timing-function.asp" \o "CSS3 transition-timing-function 属性) [transition-delay](http://www.w3school.com.cn/cssref/pr_transition-delay.asp" \o "CSS3 transition-delay 属性);

参数:(各参数应带兼容表示-webkit-,...)

[transition-property](http://www.w3school.com.cn/cssref/pr_transition-property.asp" \o "CSS3 transition-property 属性)--变换的属性名称

[transition-duration](http://www.w3school.com.cn/cssref/pr_transition-duration.asp" \o "CSS3 transition-duration 属性)--变换持续时间

[transition-timing-function](http://www.w3school.com.cn/cssref/pr_transition-timing-function.asp" \o "CSS3 transition-timing-function 属性) --变换速度曲线

[transition-delay](http://www.w3school.com.cn/cssref/pr_transition-delay.asp" \o "CSS3 transition-delay 属性)--变换执行延迟时间

其中变换速度曲线可以用贝塞尔曲线(cubic-bezier) (http://cubic-bezier.com/)来描述。

-webkit-transition: property duration;

-moz-transition: property duration;

-ms-transition: property duration;

transition: property duration;

变化位置：

-webkit-transform-origin: @ox @oy;

-moz-transform-origin: @ox @oy;

-ms-transform-origin: @ox @oy;

transform-origin: @ox @oy;

**3、渐变**

线性渐变linear-gradient(to bottom, #fff,..., #000);

|  |  |  |
| --- | --- | --- |
| 角度 | 关键字 | 作用 |
| 0deg | to top | 从下倒上 |
| 90deg | to right | 从左到右 |
| 180deg | to bottom | 从上到下 |
| 270deg | to left | 从右到左 |
|  | to top left | 从右下到左上 |
|  | to top right | 从左下到右上 |

径向渐变radial-gradient(#fff,..., #000);

... 表示可以设置多个值，radial-gradient(#fff, #ccc, #ddd, #000);

background: -webkit-linear-gradient(to bottom, start 0%, stop 100%);

background: -moz-linear-gradient(to bottom, start 0%, stop 100%);

background: -ms-linear-gradient(to bottom, start 0%, stop 100%);

background: -o-linear-gradient(to bottom, start 0%, stop 100%);

background: linear-gradient(to bottom, start 0%, stop 100%);

**4、动画**

动画规则：

@-webkit-keyframes name {

0% {}

100% {}

}

@-moz-keyframes name {

0% {}

100% {}

}

@-ms-keyframes name {

0% {}

100% {}

}

@keyframes name {

0% {}

100% {}

}

animation 参数:(各参数应带兼容表示-webkit-,...)

[animation-name](http://www.w3school.com.cn/cssref/pr_animation-name.asp" \o "CSS3 animation-name 属性): 动画名称

[animation-duration](http://www.w3school.com.cn/cssref/pr_animation-duration.asp" \o "CSS3 animation-duration 属性)：持续时间

[animation-timing-function](http://www.w3school.com.cn/cssref/pr_animation-timing-function.asp" \o "CSS3 animation-timing-function 属性)：运动曲线

[animation-delay](http://www.w3school.com.cn/cssref/pr_animation-delay.asp" \o "CSS3 animation-delay 属性)：开始延迟时间

[animation-iteration-count](http://www.w3school.com.cn/cssref/pr_animation-iteration-count.asp" \o "CSS3 animation-iteration-count 属性)：动画播放的次数 --infinite 无限次

[animation-direction](http://www.w3school.com.cn/cssref/pr_animation-direction.asp" \o "CSS3 animation-direction 属性)：是否轮流反向播放动画 --alternate 反向轮播

animation-play-state：running /paused

-webkit-animation: @arguments;

-moz-animation: @arguments;

-ms-animation: @arguments;

-o-animation: @arguments;

animation: @arguments;

**5、自适应字体大小rem**

@media screen and (min-width: 320px) {

html {

font-size: 100px;

}

}

@media screen and (min-width: 360px) {

html {

font-size: 112.5px;

}

}

@media screen and (min-width: 400px) {

html {

font-size: 125px;

}

}

@media screen and (min-width: 440px) {

html {

font-size: 137.5px;

}

}

@media screen and (min-width: 480px) {

html {

font-size: 150px;

}

}

@media screen and (min-width: 640px) {

html {

font-size: 200px;

}

}

@media screen and (min-width: 720px) {

html {

font-size: 225px;

}

}

html {

font-size: 31.25vw;

}

**6、超出文本显示圆的**

white-space: nowrap;

overflow: hidden;

text-overflow: ellipsis;

**7、水平垂直居中**

弹出层水平垂直居中：

<div class="cover">

<div class="tip">

<div class="info"></div>

</div>

</div>

.cover {

position: fixed;

display: table;

width: 100%;

height: 100%;

top: 0;

left: 0;

background: rgba(171, 171, 171, 0.65);

.tip {

display: table-cell;

vertical-align: middle;

text-align: center;

}

.info {

display: inline-block;

border-radius: 5px;

width: 80%;

height: 40%;

background: #fff;

}

}

水平居中：

1、固定宽度 + margin // 需要设置宽度

.child {width: \*\*, margin: 0 auto;}

2、table + margin //

.child {display: table, margin: 0 auto;}

3、absolute + transform //兼容性

.parent {position: relative;}

.child {position: absolute, left: 50%, transform: translateX(-50%);}

垂直居中(父容器和子容器的高度都未知)：

1、absolute + transform //兼容性

.parent {position: relative;}

.child {position: absolute, top: 50%, transform: translateY(-50%);}

2、table + table-cell + vertical-align //

.preparent{dispaly: table;}

.parent{display: table-cell; vertical-align: middle;}

水平垂直居中：

1、absolute + transform

.parent{position: relative;}

.child{position: absolute; left: 50%; top: 50%; transform: translate(-50%,-50%);}

**8、弹出层时，阻止页面滑动**

<body style="overflow: hidden;">

1. **修改input placeholder字体颜色**

(less)

input {

&::-webkit-input-placeholder {

color: #dec9b7;

}

&:-moz-placeholder {

/\* Mozilla Firefox 4 to 18 \*/

color: #999;

}

&::-moz-placeholder {

/\* Mozilla Firefox 19+ \*/

color: #999;

}

&:-ms-input-placeholder {

/\* Internet Explorer 10+ \*/

color: #999;

}

}

**三、功能**

**1、分页加载/图片懒加载(lazyImg.js)**

// 页面滚动事件

$window.on("scroll", function(){

requestAnimationFrame(function(){

// 加载下一页 imgArr需要懒加载的图片集合

if (isAutoAddNext && imgArr.length <= 1){

isAutoAddNext = false;

currentPage++;

// 查询照片

getPhotoInfo(currentPage);

}

// 懒加载图片

lazyAddImg();

});

});

// 懒加载图片的集合

getImgArr();

// 第一次加载

lazyAddImg();

if (resLenth < 10){

isLastPage = true;

isAutoAddNext = false;

} else {

isAutoAddNext = true;

}

/\*\*

\* 获取需要懒加载图片的集合

\*/

function getImgArr(){

$(".img-lazy").each(function(index, target){

imgArr.push(target);

target.className = "";

});

}

/\*\*

\* 懒加载显示图片

\*/

function lazyAddImg(){

var arr = imgArr;

var imgArrLength = arr.length;

if (imgArrLength <= 0){

if (isLastPage){

$window.off("scroll");

}

return false;

}

for(var i = 0; i < imgArrLength; i++){

var item = arr[0],

top = item.getBoundingClientRect().top;

if (top < winHeight){

var img = new Image();

img.src = item.getAttribute("data-img");

img.id = item.id;

// 删除已经加载的图片

arr.shift();

console.log(arr);

img.onload = function(){

var \_img = this,

$\_img = $("#" + \_img.id);

$\_img.attr("src", \_img.src);

$\_img.siblings(".load-img").addClass("hide");

};

} else {

break;

}

}

}

1. **判断浏览器类别**

var \_ua = navigator.userAgent.toLowerCase(),

userAgent = {

//Android终端或uc浏览器

Android:\_ua.indexOf('android')>-1||\_ua.indexOf('linux')>-1,

//Apple iphone ipad

Apple: u.indexOf('iphone') > -1 || u.indexOf('ipad') > -1,

//微信

weChat: u.indexOf('micromessenger') > -1,

//微博

weibo: u.indexOf('weibo') > -1,

windowsphone: u.indexOf('windows phone') > -1

};