二维数组求最大值

var arr = [[70,60,80,90],[60,10,20,30],[8,100,40,50]];

for(var i=0;i<arr.length;i++){

for(var j=0;j<arr[i].length;j++){

if(arr[0][0]<arr[i][j]){

var max = arr[i][j];

}

}

}

document.write(max);

数组去空

var arr = [1,2,3,4,,5,6,,7,8];

var newarr = [];

for(var i=0;i<arr.length;i++){

if(arr[i]!=undefined){

newarr[newarr.length]=arr[i];

}

}

console.log(newarr);

金字塔

<script>

ta(10)

function ta(row){

for(var i=1;i<=row;i++){

for(var z=1;z<=row-i;z++){

document.write('-');

}

for(var j=1;j<=2\*i-1;j++){

document.write('\*');

}

document.write('<br/>');

}

}

</script>

排序

var arr = [10,15,18,19,20,80,100,60,70,];

for(var i=0;i<arr.length;i++){

for(j=i+1;j<arr.length;j++){

if(arr[i]<arr[j]){

var tem = arr[i];

arr[i] = arr[j];

arr[j] = tem;

}

}

}

document.write(arr+'<br/>');

for(var i=0;i<arr.length;i++){

for(j=i+1;j<arr.length;j++){

if(arr[i]>arr[j]){

var tem = arr[i];

arr[i] = arr[j];

arr[j] = tem;

}

}

}

document.write(arr);

给数组添加数

<script>

/\*arguments代表的是一个函数所有的形参 可以把他的形参的赋给数组就好\*/

var arr = [1,2,3,4,5,6];

function push(arr){

for(var i=1;i<arguments.length;i++)

arr[arr.length]=arguments[i];

}

push(arr,'a','b');

console.log(arr);

/\*用剩余参数的形式来构成...rest 就是把剩余的形参一个一个给了定义的那个数组\*/

var arr = [1,2,3,4,5,6];

function push(arr,...rest){

for(var i=0;i<rest.length;i++)

arr[arr.length]=rest[i]

}

push(arr,"a","b")

console.log(arr);

</script>

数组转字符串 挑选出偶数来

//数组转字符串

/\*var arr = [1,2,3,4,6,7,8,9];

function pull(arr,value){

if(typeof arr == 'object' && arguments.length>1){

var result = '';

for(var i=0;i<arr.length;i++){

if(i<arr.length-1){

result += arr[i]+value;

}else{

result += arr[i];

}

}

console.log(result);

return result;

}

}

pull(arr,'-');\*/

// 挑选出偶数来放到一个数字

/\*var arr = [1,2,3,4,6,7,8,9];

var newarr = [];

function filter(arr){

for(i=0;i<arr.length;i++){

if (arr[i]%2==0) {

newarr[newarr.length]=arr[i];

}

}

console.log(newarr);

return newarr;

}

filter(arr)\*/

映射 删选

<script>

var arr = [1,2,3,4];

var result = map(arr,function(value){

result = value\*2;

return result;

})

console.log(result);

function map(arr,fn){

var newarr = [];

for(var i=0;i<arr.length;i++){

newarr[newarr.length] = fn(arr[i]);

}

return newarr;

}

var arr = [1,2,3,4,5,6,7,8,9];

var result = filter(arr,function(value){

result = value;

return result;

})

console.log(result);

function filter(arr,fn){

var newarr = [];

for(var i=0;i<arr.length;i++){

if (fn(arr[i])%2==1) {

newarr[newarr.length] = fn(arr[i]);

}

}

return newarr;

}

var arr = [1,2,3,4,5,6,7,8];

var result = filter(arr,function(value){

result = value;

return result;

})

console.log(result);

function filter(arr,fn){

var newarr = [];

for(var i=0;i<arr.length;i++){

if (fn(arr[i])%2==1) {

newarr[newarr.length] = fn(arr[i]);

}

}

return newarr;

}

/\*映射\*/

var arr = [1,2,3,4,5,6];

var result = map(arr,function(value){

return value\*2;

})

console.log(result);

function map(arr,fn){

var newarr = [];

for(var i=0;i<arr.length;i++){

newarr[newarr.length]=fn(arr[i])

}

return newarr;

}

/\*删选\*/

var arr = [1,2,3,4,5,6,7];

var result = filter(arr,function(value){

return value>3;

})

console.log(result);

function filter(arr,fn){

var newarr = [];

for(var i=0;i<arr.length;i++){

if(fn(arr[i])){

newarr[newarr.length]=arr[i];

}

}

return newarr;

}

</script>

翻转 万能添加和删除 some every 递归函数

/\*翻转\*/

var arr = [1,2,3,4];

var newarr = [];

function reverse(arr){

for(var i=arr.length-1;i>=0;i--){

push(newarr,arr[i]);

}

return newarr;

}

function push(arr,...rest){

for(var i=0;i<rest.length;i++){

arr[arr.length] = rest[i];

}

}

var result = reverse(arr);

console.log (result);

/\*万能删除\*/

var newarr= [];

function splicecancle(arr,pos,num){

for(var i=0;i<arr.length;i++){

if(i>=pos && i<pos+num){

continue;

}

newarr[newarr.length]=arr[i];

}

return newarr;

}

/\*var arr = [1,2,3,4,5];

var result = splice(arr,1,2);

console.log(result);\*/

/\*万能去添加\*/

function spliceadd(arr,pos,...rest){

var newarr = [];

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

newarr[newarr.length]=arr[i];

}

for(var i=0;i<rest.length;i++){

newarr[newarr.length]=rest[i];

}

for(var i=pos;i<arr.length;i++){

newarr[newarr.length]=arr[i];

}

return newarr;

}

var arr = [1,2,3,4,5];

var result = spliceadd(arr,1,'a','s','d');

console.log(result);

/\*万能的添加删除数组 先删除数组 在添加\*/

function splice(arr,pos,num,...rest){

var newarr=splicecancle(arr,pos,num)

if(rest.length>0){

var newarr = spliceadd(arr,pos,...rest);

}

return newarr;

}

var arr = [1,2,3,4,5,6];

var result=splice(arr,1,2,'a','s','d','f');

console.log(result);

/\*some用来判断数组里面一个符合条件\*/

var arr = [1,2,3,4];

var result = some(arr, function(value){

return value>0;

})

console.log(result);

function some(arr,fn){

for(var i=0;i<arr.length;i++){

if(fn(arr[i])){

return true;

}

}

return false;

}

/\*every每一个都满足条件\*/

var arr = [1,2,3,4];

var result = every(arr, function(value){

return value>0;

})

console.log(result);

function every(arr,fn){

for(var i=0;i<arr.length;i++){

if(!fn(arr[i])){

return false;

}

}

return true;

}

/\*递归函数（函数自己调用自己本身）\*/

fn(1);

function fn(num){

if(num<5){

fn(++num);

}

alert(num);

}

/\*一个数组我想把arr给到ar1 利用浅拷贝和浅拷贝\*/

/\*深拷贝和浅拷贝\*/

var arr = [1,2,3,4,5,6,['x','y','z']];

function copy(arr){

var newarr = [];

for(var i=0;i<arr.length;i++){

if(typeof arr[i] == 'object'){

newarr[newarr.length]=copy(arr[i]);

}else{

newarr[newarr.leng

th]=arr[i];

}

}

return newarr;

}

var result = copy(arr);

console.log(result);

arr[0]="a";

console.log(arr);

let声明函数

<script>

/\*块级作用域\*/

var arr = [];

for(var i=0;i<10;i++){

arr[i]=function(){

alert(i)

}

}

arr[2]();

console.log(i);

var arr = [];

for(let i=0;i<10;i++){

arr[i]=function(){

alert(i)

}

}

arr[2]();

console.log(i);

</script>

闭包函数

<script>

/\*闭包函数 在一个函数内部嵌套一个函数\*/

function fn(){

var num = 10;

return fn1

function fn1(){

num++;

return num;

}

}

var result = fn();

alert(result());

</script>

箭头函数

<script>

/\*箭头函数一般用在回调函数中\*/

/\*var fn = a =>a;

alert(fn('a'));

var arr = [1,2,3,4];

var fn = (a,b) =>a+b;

alert(fn(1,2));\*/

/\*var fn = (a,b) =>{

alert(a-b);

alert(a+b);

return (a,b);

}

alert(fn(1,2));\*/

/\*arr.sort(function(a,b){return a-b;})

var arr=[12,2,123,23,1,3];

arr.sort((a,b)=>b-a);

arr.splice(1,2);

console.log(String(arr));

console.log(typeof String(arr));\*/

/\*箭头函数在回调函数中使用\*/

/\* var result = map(arr,function(value){

return value\*2;

})\*/

var resuit = map(arr,(value =>value\*2));

function map(arr,fn){

var newarr = [];

for(var i=0;i<arr.length;i++){

newarr[newarr.length]=fn(arr[i]);

}

return newarr;

}

console.log(result);

</script>

构造函数

<script>

function Person(){

}

let zhangsan = new Person();

zhangsan.age = 18;

zhangsan.name = 'zhangsan';

zhangsan.sex = '男';

alert(zhangsan.age);

alert(zhangsan.name);

alert(zhangsan.sex);

zhangsan.say = function(){

alert(zhangsan.name);

}

zhangsan.say();

</script>

内置顶层函数

console.log(escape('白羊'));

console.log(unescape('%u767D%u7F8A'));

console.log(Number(0123));

console.log(parseInt('-100px'));

json创建对象

let iphone = {

"name" : 'ihpone8',

"screen" : 'OLED',

"size" : 5.8,

"technology" : '虹膜识别+屏下指纹识别',

music : function(){

alert('听音乐');

},

mas : function(){

alert('发短信');

}

}

console.log(iphone);

iphone.mas();

/\*原生的 不建议用\*/

let lisi = new object();

lisi.age = 18;

通过类的方式创建对象

<script>

class Student{

constructor(){

this.name = 'zhangsan';

this.age = 18;

this.say = function(){

alert(1);

}

}

/\*\*/

Study(){

alert('学习');

}/\*一般情况不加;绝对不能加，\*/

}

let zhangsan = new Student();

console.log(zhangsan);

zhangsan.Study();

</script>

传参

<script>

function Student(name,age){

this[1] = 'a';

this[2] = 'b'

this.name = name;

this.age = age;

this.barin = function(){

alert(`${this.name}脑子`)

}

this.study = function(value){

this.say(value);

}

this.say = function(value){

alert(value)

}

}

let zhangsan = new Student('zhangsan',18);

console.log(zhangsan);

zhangsan.study('学习');

zhangsan.barin()

</script>

创建数组

<script>

/\*数组 用构造函数的形式来\*/

/\*创建自己的数组\*/

let arr = new myArray('a','b','c','d',1,2,3,4,5,6);

arr.push('x','y','z');

let join = arr.join();

function myArray(){

for(let i=0;i<arguments.length;i++){

this[i] = arguments[i];

}

this.length=arguments.length;

this.push = function(){

for(let i=0;i<arguments.length;i++){

this[this.length]=arguments[i];

this.length++;

}

}

this.join = function(value){

/\* value = value == undefined? '-': flase;\*/

/\* value=value||'-'\*/

let result = '';

for(let i=0;i<this.length;i++){

if(i<this.length-1){

result += this[i]+value;

}else{

result += this[i]

}

}

return result;

}

}

console.log(join);

console.log(arr)

/\*indexof lastindexof includes reverse join拼接 sort排序

some every map filter

\*/

</script>

调用以前的

<script>

let arr = new myArray(-1,-2,8,9,5,10,7,6);

/\*首次出现输出他的位置 indexOf\*/

let index = arr.indexOf(5);

/\*满足条件则输出true\*/

let some = arr.some(function(value){

return typeof value == 'number';

})

/\*满足全部条件输出true\*/

let every = arr.every(function(value){

return typeof value == 'number';

})

/\*映射\*/

let map = arr.map(function(value){

return value\*2;

})

/\*filter\*/

let filter = arr.filter(function(value){

return value>0;

})

/\*include\*/

let include = arr.include(10);

/\*reverse\*/

let reverse = arr.reverse();

/\*join\*/

let join = arr.join('-');

function myArray(){

for(let i=0;i<arguments.length;i++){

this[i] = arguments[i];

}

this.length = arguments.length;

/\*首次出现输出他的位置\*/

this.indexOf = function(value){

for(let i=0;i<this.length;i++){

if(this[i]==value){

return i;

}

}

}

/\*满足条件则输出true\*/

this.some = function(fn){

for(let i=0;i<this.length;i++){

if(fn(this[i])){

return true;

}

}

return -1;

}

/\*满足全部条件输出true\*/

this.every = function(fn){

for(let i=0;i<this.length;i++){

if(!fn(this[i])){

return false;

}

}

return true;

}

/\*映射\*/

this.map = function(fn){

var newarr = [];

for(let i=0;i<this.length;i++){

newarr[newarr.length]=fn(this[i]);

}

return newarr;

}

/\*filter\*/

this.filter = function(fn){

var newarr = [];

for(let i=0;i<this.length;i++){

if(fn(this[i])){

newarr[newarr.length]=this[i];

}

}

return newarr;

}

/\*include\*/

this.include = function(value){

for(let i=0;i<this.length;i++){

if(this[i]==value){

return true;

}

}

return false;

}

this.reverse = function(){

var newarr = [];

for(let i=this.length-1;i>=0;i--){

newarr[newarr.length]=this[i];

}

return newarr;

}

this.join = function(value){

let result = '';

for(let i=0;i<this.length;i++){

if(i<this.length-1){

result += this[i]+value;

}else{

result += this[i]

}

}

return result;

}

this.sort = function(){

for(let i=0;i<this.length;i++){

for(let j=i+1;j<this.length;j++){

if(this[j]<this[i]){

var tem = this[i];

this[i] = this[j];

this[j] = tem;

}

}

}

}

}

arr.sort();

console.log(arr);

console.log(join);

console.log(reverse);

console.log(include);

console.log(filter);

console.log(map);

console.log(every);

console.log(some);

console.log(index);

</script>

原型对象

<script>

function Person(nmae){

this.name = name;

}

Person.prototype = {

play : function(){

alert('wan');

},

eat : function(){

alert('chi');

}

}

/\* Person.prototype.aa = function(){

alert('1111')

}\*/

let lisi = new Person('lisi');

let zhangsan = new Person('zhangsan');

zhangsan.play();

/\*删除属性和方法delete zhangsan.sex\*/

</script>

字符串对象

<script>

/\*var str = "123456";\*/

/\*console.log(str.\_\_proto\_\_)\*/

/\*返回指定位置的值\*/

/\*var str1 = str.charAt(1);

console.log(str1);

var str1 = str.charCodeAt(2);

console.log(str1);

var str1 = String.fromCharCode(100);

console.log(str1)\*/

var str = 'abcdabcdabcd';

var re = replaceAll(str,'cd');

function replaceAll(str,rStr){

/\*var star = '';

for(var i=0;i<rStr.length;i++){

star += '\*';

}\*/

var star = '\*'.repeat(rStr.length);

while(str.includes(rStr)){

str = str.replace(rStr,star);

}

return str;

}

console.log(re);

</script>

替换

<script>

let url = 'https//www.baidu.comhttps//www.baidu.comhttps//www.baidu.comhttps//www.baidu.comhttps//www.baidu.comhttps//www.baidu.comhttps//www.baidu.comhttps//www.baidu.com'

/\*let str = 'baidu'\*/

/\*function replace(arr,str){

let str1 = '\*'.replace(str.length);

let newarr = [];

for(let i=0;i<url.length;i++){

if(url.includes(str)){

newarr[newarr.length] = url.indexOf(str);

let re = url.indexOf(str);

newarr.push(re);

url = url.replace(str,str1)

}

}

return newarr;

}

let result = replace(url,str)\*/

function replace(arr,str){

let str1 = '\*'.repeat(str.length);

let newarr = [];

for(let i=0;i<url.length;i++){

if(url.substr(i,5) == str){

let re = i;

newarr.push(re);

}

}

return newarr;

}

let result = replace(url,'baidu')

console.log(result);

</script>

继承

<script>

function Person(){

this.name = 'Person';

this.age = 18;

this.say = 'nihao'

this.eat = function(){

alert("eat");

}

this.sleep = function(){

alert('sleep');

}

this.aa = function(){

alert('Person的构造函数')

}

}

Person.prototype = {

aa : function(){

alert('person的原型对象');

}

}

function Student(){

this.class = 'wuif1707';

this.number = '1707001';

this.skill = function(){

alert('我是fullstuck')

}

}

let zhangsan = new Person();

Student.prototype = new Person();

let lisi = new Student();

/\*console.log(lisi.name);\*/

lisi.aa();

</script>

数学对象

<script>

/\*let arr = ['a','b','c','d','e','f','g'];

let newarr = [];

function random(arr){

for(let i=0;i<3;i++){

let value = Math.floor(Math.random()\*arr.length);

if(!newarr.includes(arr[value])){

newarr.push(arr[value]);

}

}

return newarr

}

let result = random(arr);

console.log(result);\*/

/\* let arr = ['a','b','c','d','e','f','g'];

function Randoma(arr){

let newarr = [];

for(let i=0;i<3;i++){

let num = Math.floor(Math.random()\*arr.length);

if(!newarr.includes(arr[num])){

newarr.push(arr[num]);

}else{

i --;

}

}

return newarr;

}

console.log(Randoma(arr));\*/

/\*let arr = ['a','b','c','d','e','f','g'];

function Randoma(arr,n){

let newarr = [];

for(let i = 0;i<n;i ++){

let num = Math.floor(Math.random()\*arr.length);

while(newarr.includes(arr[num])){

num = Math.floor(Math.random()\*arr.length);

}

newarr.push(arr[num]);

}

return newarr;

}

console.log(Randoma(arr,3));\*/

/\* let arr = ['a','b','c','d','e','f','g'];

function random(arr){

let newarr = [];

for(let i=0;i<3;i++){

let num = Math.floor(Math.random()\*arr.length);

if(!newarr.includes(arr[num])){

newarr.push(arr[num]);

}else{

i--;

}

}

return newarr;

}

let result = random(arr);

console.log(result);\*/

/\*错的\*/

/\*let arr = ['a','b','c','d','e','f','g'];

function random(arr){

let newarr = [];

for(let i=0;i<3;i++){

if(newarr.includes(arr[Math.floor(Math.random()\*arr.length)])){

newarr.push(arr[Math.floor(Math.random()\*arr.length)])

}

}

return newarr;

}

let result = random(arr);

console.log(result);\*/

</script>

数组去重复

<script>

/\*let arr = [1,2,3,4,5,1,2,3];

let newarr = [];

arr.forEach = (value => {

for(let i=0;i<newarr.length;i++){

if(!newarr.includes(arr[value])){

let result = arr[value];

newarr.push(result);

}

}

return newarr;

})

console.log(newarr)\*/

/\*方法一\*/

/\*let arr = [1,2,3,4,5,1,2,3];

let newarr = [];

let re = qu(arr);

function qu(arr){

for(let i=0;i<arr.length;i++){

if(newarr.includes(arr[i]) == false){

newarr.push(arr[i]);

}

}

return newarr;

}

console.log(re);\*/

/\*方法二\*/

/\* let arr = [1,2,3,4,5,1,2,3];

function qu(arr){

let newarr = [];

for(let i=0;i<arr.length;i++){

let flag = true;

for(let j=i+1;j<arr.length;j++){

if(arr[j] == arr[i]){

flag = false;

break;

}

}

if(flag){

newarr.push(arr[i]);

}

}

return newarr;

}

let result = qu(arr);

console.log(result);\*/

/\*\*/

/\* let arr = [1,2,3,4,5,1,2,3];

function qu(arr){

let newarr = [];

for(let i in arr){

let flag = true;

for(let j=i+1;j<arr.length;j++){

if(arr[j] == arr[i]){

flag = false;

break;

}

}

if(flag){

newarr.push(arr[i]);

}

}

return newarr;

}

let result = qu(arr);

console.log(result);\*/

/\*Array.prototype. = function(){

}\*/

/\*第三种方法\*/

var arr = [1,2,3,4,5,1,2,3]

function unique(arr){

var tmp = [];

for(var i in arr){

if(!tmp.includes(arr[i])){

tmp.push(arr[i]);

}

}

return tmp;

}

var arr = unique(arr);

console.log(arr);

</script>

<script>

/\*从数组中随机抽取3个数，组成新的数组

新数组 3个元素

不存在 push

存在 再产生一个随机数\*/

let arr = ['a','b','c','d','e','f','g'];

function Randoma(arr,n){

let newarr = [];

for(let i = 0;i < n;i ++){

let num = Math.floor(Math.random()\*arr.length);

/\*if(!newarr.includes(arr[num])){

newarr.push(arr[num]);

}else{

i --;

}\*/

while(newarr.includes(arr[num])){

num = Math.floor(Math.random()\*arr.length);

}//当数组存在时重新产生一个随机数，让数组中产生一些不重复的随机数

newarr.push(arr[num]);

}

return newarr;

}

//新数组

//数组长度小于，产生随机数，往新数组中添加数组元素

//当数组元素存在时，不输出

//不存在时 添加新的数组元素

/\*function Randoma(arr,n){

let newarr = [];

while(newarr.length < n){

let num = Math.floor(Math.random()\*arr.length);

if(!newarr.includes(arr[num])){

newarr.push(arr[num]);

}

}

return newarr;

}\*/

console.log(Randoma(arr,4));

</script>

浏览器兼容 获取类名

function getClass(classname){

/\*判断在某一范围的class 给定默认值没有的话输出默认值 用三元表达式\*/

ranger = ranger ? ranger :document;

// 判断是否有ClassName这个方法

var newarr = [];

if(document.getElementsByClassName){

return document.getElementsByClassName(classname);

}else{

var all = document.getElementsByTagName('\*');

for(var i=0;i<all.length;i++){

if(checkClass(all[i].className,classname)){

newarr.push(all[i]);

}

/\*if(all[i].ClassName = classname;){

newarr.push(all[i]);

}\*/

}

return newarr;

}

}

// 检查是否存在classname这个类名

function checkClass(className,classname){

var arr = className.split(' ');

for(var i=0;i<arr.length;i++){

if(arr[i] == classname){

return true;

}

}

return false;

}

$函数

function getClass(classname,ranger){

/\*判断在某一范围的class 给定默认值没有的话输出默认值 用三元表达式\*/

ranger = ranger ? ranger :document;

// 判断是否有ClassName这个方法

var newarr = [];

if(document.getElementsByClassName){

return ranger.getElementsByClassName(classname);

}else{

var all = ranger.getElementsByTagName('\*');

for(var i=0;i<all.length;i++){

if(checkClass(all[i].className,classname)){

newarr.push(all[i]);

}

/\*if(all[i].ClassName = classname;){

newarr.push(all[i]);

}\*/

}

return newarr;

}

}

// 检查是否存在classname这个类名

function checkClass(className,classname){

var arr = className.split(' ');

for(var i=0;i<arr.length;i++){

if(arr[i] == classname){

return true;

}

}

return false;

}

/\*

用$获取指定的元素

$(select);

$('.one') $('#one')

\*/

/\*$函数\*/

function $(select){

// 范围

var ranger = ranger||document;

var first = select.charAt(0);

if(first == '.'){

// 获取class类名

return getClass(select.substring(1),ranger);

}else if(first == '#'){

// 获取id

return ranger.getElementById(select.substring(1));

}else if(/^[a-z][a-z1-6]{0,7}$/.test(select)){

return ranger.getElementsByTagName(select);

}

}

/\*

悬浮对象

属性：对象的描述

谁 速度 偏移量

方法：功能

动画 move

\*/

function Float(obj){

this.obj = obj;

this.maxX = window.innerWidth - this.obj.offsetWidth;

this.maxY = window.innerHeight - this.obj.offsetHeight;

this.speedx = 1;

this.speedy = 1;

}

/\*

计算浏览器的尺寸的修改

\*/

Float.prototype.resize = function(){

this.maxX = window.innerWidth - this.obj.offsetWidth;

this.maxY = window.innerHeight - this.obj.offsetHeight;

}

/\*

鼠标移动上去停止

\*/

Float.prototype.stop = function(){

clearInterval(this.t);

}

/\*移动\*/

Float.prototype.move = function(){

this.t = setInterval(fn, 6)

let that = this;

function fn(){

let lefts = that.obj.offsetLeft + that.speedx;

let tops = that.obj.offsetTop + that.speedy;

if(lefts>=that.maxX){

lefts = that.maxX;

that.speedx\*=-1;

}

if(lefts<=0){

lefts = 0;

that.speedx\*=-1;

}

if(tops>=that.maxY){

tops = that.maxY;

that.speedy\*=-1;

}

if(tops<=0){

tops = 0;

that.speedy\*=-1;

}

that.obj.style.left = lefts +'px';

that.obj.style.top = tops +'px';

// console.log(that.obj.style.left);

}

}

小广告面向对象

<script>

window.onload = function(){

let box = document.getElementsByClassName('box')[0];

let obj = new Float(box);

obj.move();

box.onmouseover = function(){

obj.stop();

}

box.onmouseout = function(){

obj.move();

}

window.onresize = function(){

obj.resize();

}

}

</script>

function Float(obj){

this.obj = obj;

this.maxX = window.innerWidth - this.obj.offsetWidth;

this.maxY = window.innerHeight - this.obj.offsetHeight;

this.speedx = 1;

this.speedy = 1;

}

/\*

计算浏览器的尺寸的修改

\*/

Float.prototype.resize = function(){

this.maxX = window.innerWidth - this.obj.offsetWidth;

this.maxY = window.innerHeight - this.obj.offsetHeight;

}

/\*

鼠标移动上去停止

\*/

Float.prototype.stop = function(){

clearInterval(this.t);

}

/\*移动\*/

Float.prototype.move = function(){

this.t = setInterval(fn, 6)

let that = this;

function fn(){

let lefts = that.obj.offsetLeft + that.speedx;

let tops = that.obj.offsetTop + that.speedy;

if(lefts>=that.maxX){

lefts = that.maxX;

that.speedx\*=-1;

}

if(lefts<=0){

lefts = 0;

that.speedx\*=-1;

}

if(tops>=that.maxY){

tops = that.maxY;

that.speedy\*=-1;

}

if(tops<=0){

tops = 0;

that.speedy\*=-1;

}

that.obj.style.left = lefts +'px';

that.obj.style.top = tops +'px';

// console.log(that.obj.style.left);

}

}

小广告

<script>

window.onload =function(){

let box = document.getElementById('box');

setInterval(move,100);

let maxY = window.innerHeight-box.offsetHeight;

let maxX = window.innerWidth-box.offsetWidth;

let speed = 10;

let speeds = 10;

function move(){

let left = box.offsetLeft+speeds;

let top = box.offsetTop +speed;

console.log(left);

if(left>=maxX){

left = maxX;

speeds\*=-1;

}

if(left<=0){

left = 0;

speeds\*=-1;

}

if(top>=maxY){

top = maxY;

speed\*=-1;

}

if(top<=0){

top = 0;

speed\*=-1;

}

box.style.top = top + 'px';

box.style.left = left+'px';

}

}

</script>

继承

<script>

class Person{

constructor(){

this.name = "zhangsan";

this.age = "20"

}

}

class Student extends Person{

constructor(){

super();

}

}

</script>

节点

<body>

<div class="box">

<div class="son">

<!-- 这是个注释 -->

</div>

<span>你好</span>

<span>你好</span>

<div class="son1"></div>

<div class="son2"></div>

<div class="son3"></div>

<h1>123</h1>

</div>

<script>

window.onload = function(){

let box = document.querySelector('div.son:first-child');

/\*let childs = children(box);\*/

/\*console.log(box);\*/

/\*console.log(all);\*/

/\*function filter(all,fn){

let newarr=[];

for(let i=0;i<all.length;i++){

if(fn(all[i])){

newarr[newarr.length] = all[i];

}

}

}

filter(all,function(value){

return value.nodeType == 1;

})

return newarr;

}

console.log(all);\*/

/\*function children(parent){

let newarr = [];

let childs = parent.childNodes;

let arr = Array.from(childs);

let newarr = arr.filter(elements=>elements.nodeType==1);

return newarr;

}\*/

/\*

element 兄弟元素

父元素 childs 截取

比较截取的名字和寻找的名字比较

\*/

/\*function next(elements,tagname){

let parent = elements.parentNode;

let childs = parent.children;

let index = 0;

for(let i=0;i<childs.length;i++){

if(childs[i] == elements){

index = i;

break;

}

}

let nextAll = Array.from(childs).slice(index+1);

for(let i=0; i<nextAll.length;i++){

if(nextAll[i].nodeName == tagname.toUpperCase()){

return nextAll[i];

}

}

return null;

}

let son = document.querySelector('div.son');

let nexts = next(son,'div')

console.log(nexts);\*/

/\*function next(elements,tagname){

let parent = elements.parentNode;

let childs = parent.children;

let index = 0;

let i = 0;

while(i<childs.length){

if(childs[i] == elements){

index = i;

break;

}

i++;

}

let nextAll = Array.from(childs).slice(index+1);

let j = 0;

while(j<nextAll.length){

if(nextAll[j].nodeName == tagname.toUpperCase()){

return nextAll[i];

}

j++;

}

return null;

}

let son = document.querySelector('div.son');

let nexts = next(son,'div')

console.log(nexts);\*/

function next(elements,tagname){

let nextS = elements.nextElementSibling;

console.log(nextS)

if(nextS == null){

return null;

}

while(nextS.nodeName == tagname.toUpperCase()){

nextS = nextS.nextElementSibling;

if(nextS == null){

return null;

}

if(nextS.nodeName == tagname.toUpperCase()){

return nextS;

}

}

}

let son = document.querySelector('div.son');

let nexts = next(son,'div')

console.log(nexts);

/\*function parent(element){

if(element.nodeName == 'BODY' ||element.nodeName =='HTML'){

return;

}

let arr = [];

let parent = element.parentNode;

while(parent.nodeName !='BODY'){

arr.push(parent);

parent = parent.parentNode;

}

return arr;

}\*/

/\*function nextAll(element){

let arr = [];

let nextS = element.nextElementSibling;

if(!nextS){

return null;

}

arr.push();

while(nextS != null){

arr.push(nextS);

nextS = nextS.nextElementSibling;

}

return arr;

}

let re = nextAll(son);;

console.log(re);\*/

}

</script>

圆圈

<body>

<div class='pointer'></div>

<script>

let pointer = document.querySelector('div.pointer');

circle(100,30);

function circle(r,num){

/\*let num = 20;\*/

let deg = 360/num;

// let r = 300;

for(let i=0;i<num;i++){

if(i%30 == 0){

r+=30

}

let divs = document.createElement('div');

divs.className = 'circle';

let rad = deg\*i\*Math.PI/180;

let lefts = r\*Math.cos(rad);

let tops = r\*Math.sin(rad);

/\*divs.style.left = `${lefts}px`;

divs.style.top =`${tops}px`;\*/

divs.style.background = color();

setTimeout(function(){

divs.style.transitionDelay=`${i\*100}ms`

divs.style.left = `${lefts}px`;

divs.style.top =`${tops}px`;

}, 5)

pointer.appendChild(divs);

}

}

/\*随机颜色\*/

function color(){

let str = 'rgb(';

for(let i=0;i<3;i++){

let num = Math.floor(Math.random()\*256);

str += num + ',';

}

return str.slice(0,-1)+')';

}

</script>

</body>

节点库

/\*在特定父元素的后面去添加 \*/

HTMLElement.prototype.insertAfter=function(element){

let next = this.nextElementSibling;

let parent = this.parentNode;

if(next){

parent.insertBefore(element,next);

}else{

parent.appendChild(element);

}

}

/\*在特定元素的儿子里面最前面加一份元素\*/

HTMLElement.prototype.prependChild = function(element){

let first = this.firstElementChild;

if(first){

this.insertBefore(element,first);

}else{

this.appendChild(element);

}

}

/\*把一个元素插入到父元素中\*/

HTMLElement.prototype.appeadTo = function(element){

emement.appendChild(this);

}

/\*把一个元素插入到父元素中的最后面\*/

HTMLElement.prototype.prependTo = function(element){

element.appendChild(this);

}

/\*清空\*/

HTMLElement.prototype.empy = function(){

/\*let child = this.children;

for(let i=child.length-1;i<=0;i--){

this.removeChild(child[i]);

child[i] = null;

}\*/

this.innerHTML = '';

}

随机产生20div

<script>

for(let i = 0; i<20; i++){

let divs = document.createElement('div');

divs.className = 'cricle';

let lefts = Math.floor(Math.random()\*(innerWidth-50));

let tops = Math.floor(Math.random()\*(innerHeight-50));

let num = Math.floor(Math.random()\*256);

let num1 = Math.floor(Math.random()\*256);

let num2 = Math.floor(Math.random()\*256);

let heig = Math.floor(Math.random()\*70+30);

let wid = Math.floor(Math.random()\*70+30);

let rgb;

let back = 'rgb' +'('+num+','+num1+','+num2+')';

/\*divs.style.left = `${lefts}px`;

divs.style.top = `${tops}px`;\*/

/\*divs.style.height =`${heig}px`;

divs.style.width = `${wid}px`;\*/

divs.style.background = back;

document.body.appendChild(divs);

setTimeout(function(){

divs.style.transitionDelay=`${i\*100}ms`

divs.style.left = `${lefts}px`;

divs.style.top = `${tops}px`;

}, 5)

}

</script>

按需加载

window.onload = function(){

let ch = innerHeight;

let floor = document.querySelectorAll('.floor');

let aside = document.querySelectorAll('aside .box');

let top = document.querySelector('.top')

console.log(top);

let floorAll = [];

let flag = true;

let color = ['orange','red','green','purple','skyblue'];

floor.forEach(element=>{

floorAll.push(element.offsetTop);

})

/\*侧面\*/

aside.forEach((element,index)=>{

document.body.scrollTop = floorAll[index];

element.onclick =function(){

for(let i=0;i<aside.length;i++){

aside[i].style.background = '#FFF';

}

element.style.background = color[index];

flag = false;

animate(document.body,{scrollTop:floorAll[index]},function(){

flag = true;

});

}

})

/\*滚动\*/

let flag1 = true;

window.onscroll = function(){

if(!flag){

return

}

let scrolltop = document.body.scrollTop;

if(scrolltop>=500){

if(flag1){

flag1 = !flag1;

animate(top,{top:0});

}

}else{

if(!flag1){

flag1 = !flag1;

animate(top,{top:-50});

}

}

floorAll.forEach((value,index) =>{

if(scrolltop+ch>=value+300){

let imgs = floor[index].getElementsByTagName('img');

for(let i=0;i<aside.length;i++){

aside[i].style.background = '#fff';

}

aside[index].style.background = color[index];

for(let i=0;i<imgs.length;i++){

imgs[i].src = imgs[i].getAttribute('tem');

}

}

})

}

}

拖拽面向对象

class Drag{

constructor(obj){

this.obj = obj

}

move(){

let that = this;

this.obj.addEventListener('mousedown',function(e){

let ox = e.offsetX,oy = e.offsetY;

document.addEventListener('mousemove',fn);

that.obj.addEventListener('mouseup',function(){

document.removeEventListener('mousemove',fn)

})

function fn(e){

let lefts = e.clientX,tops = e.clientY;

that.obj.style.left = `${lefts-ox}px`;

that.obj.style.top = `${tops-oy}px`;

}

})

}

}

拖拽

window.addEventListener('load',function(){

let box = document.querySelector('div.box');

box.addEventListener('mousedown',function(e){

let ox = e.offsetX ,oy = e.offsetY;

document.addEventListener('mousemove',fn);

box.addEventListener('mouseup',function(){

document.removeEventListener('mousemove',fn);

})

function fn(e){

let lefts = e.clientX,tops = e.clientY;

box.style.left = `${lefts-ox}px`;

box.style.top = `${tops-oy}px`;

}

})

})

放大镜

$(function(){

let small = $('.small')[0];

let big = $('.big')[0];

let imgs = $('img',big)[0];

let zhe = $('.zhe')[0];

let opacity = $('.opacity')[0];

let zhew = zhe.offsetWidth,zheh = zhe.offsetHeight;

let sw = small.offsetWidth,sh = small.offsetHeight;

let bw = big.offsetWidth,bh = big.offsetHeight;

let bili = imgs.offsetWidth/parseInt(getComputedStyle(small,null).width);

opacity.addEventListener('mousemove',fn);

function fn(e){

let ox = e.offsetX-zhew/2 , oy = e.offsetY-zheh/2;

console.log(ox)

/\*imgs.style.left = `${-ox\*bili}px`;

imgs.style.top = `${-oy\*bili}px`;\*/

if(ox>=sw-zhew){

ox = sw-zhew;

}

if(ox<=0){

ox = 0;

}

if(oy>=sh-zheh){

oy = sh - zheh;

}

if(oy<=0){

oy = 0;

}

zhe.style.left = `${ox}px`;

zhe.style.top = `${oy}px`;

/\*

zhe big ox

small imgs x

\*/

imgs.style.width = sw \* bw / zhew +'px';

imgs.style.height = sh \* bh / zheh +'px';

imgs.style.left = -sw \* ox / zhew +'px';

imgs.style.top = -sh \* oy /zheh + 'px';

}

})

留言

window.onload = function(){

let son = document.querySelector('div.son');

let text = document.getElementById('text');

let input = document.querySelector('input')

let btn = document.querySelector('.button');

let num = document.querySelector('.text .num');

let rest = document.querySelector('.text .rest')

console.log(num)

text.onkeyup = function(){

let value = this.value;

num.innerText = value.length;

rest.innerText = this.maxLength -value.length;

}

/\*btn.onclick = function(){

let divs = document.createElement('div');

divs.className = 'divs';

let texts = text.value;

let str = '';

str = texts;

// son.innerHTML =str;

divs.innerHTML = str;

text.value = '';

console.log(divs);

son.appendChild(divs)

} \*/

btn.onclick = function(){

let mas = document.querySelector('.mas');

mas.innerHTML += `

<ul class='mas1'>

<img src="1.jpg" alt="">

<li>

<h3>${input.value}</h3>

<p>${text.value}</p>

</li>

</ul>

`

input.value = '';

text.value = '';

}

}

可编辑表格storage.js

class storage{

constructor(){

this.date = [];

}

inte(){

let students = [

{name:'张三',sex:'男',iphone:'123456',class:'wuif1701',address:'山西太原'},

{name:'李四',sex:'男',iphone:'123456',class:'wuif1701',address:'山西晋中'},

{name:'王五',sex:'男',iphone:'123456',class:'wuif1701',address:'山西吕梁'},

{name:'赵六',sex:'男',iphone:'123456',class:'wuif1701',address:'山西榆次'},

{name:'张三1',sex:'男',iphone:'123456',class:'wuif1701',address:'山西晋城'},

{name:'李四1',sex:'男',iphone:'123456',class:'wuif1701',address:'山西运城'},

{name:'王五1',sex:'男',iphone:'123456',class:'wuif1701',address:'山西大同'},

{name:'赵六1',sex:'男',iphone:'123456',class:'wuif1701',address:'山西太原'},

];

localStorage.setItem('students',JSON.stringify(students));

}

getdate(){

let date = localStorage.getItem('students');

if(!date){

this.inte();

}

return date = JSON.parse(localStorage.getItem('students'))

}

update(row,type,value){

let date = this.getdate();

date[row][type] = value;

this.save(date);

}

del(row){

let date = this.getdate();

date.splice(row,1);

this.save(date);

}

add(obj){

let date = this.getdate();

date.push(obj);

this.save(date);

}

save(date){

localStorage.students = JSON.stringify(date);

}

}

table.js

window.addEventListener('load',function(){

let table = document.querySelector('tbody');

let addbtn = document.querySelector('.addbtn');

let dateObj = new storage();

load();

function load(){

let students = dateObj.getdate();

students.forEach((element,index) =>{

serch(element,index);

})

}

function serch(obj,index){

let trs = document.createElement('tr');

trs.id = index;

trs.innerHTML=`

<td type="name">${obj.name}</td>

<td type="sex">${obj.sex}</td>

<td type="iphone">${obj.iphone}</td>

<td type="class">${obj.class}</td>

<td type="address">${obj.address}</td>

<td class="del">

<button class="del">删除</button>

</td>

`

/\* for(let j in obj){

trs.innerHTML=`

<td type="${j}">{obj[j]}</td>

`

}\*/

table.appendChild(trs);

}

addbtn.onclick = function(){

let trs = {name:'张三',sex:'男',iphone:'123456',class:'wuif1701',address:'山西太原'};

let index = trs.childElementCount;

serch(trs,index);

dateObj.add(trs);

table.innerHTML='';

load();

}

table.ondblclick = function(e){

let element = e.target;

if(element.nodeName == 'TD'&&element.className!='del'){

let oldv = element.innerText;

element.innerText = '';

let inputs = document.createElement('input');

inputs.value = oldv;

element.appendChild(inputs);

inputs.onblur = function(){

let noww = inputs.value;

element.removeChild(inputs);

/\*element.innerText = neww;\*/

/\*if(neww){

element.innerText = noww;

}else{

element.innerText = oldv;

}\*/

if(!noww){

noww = oldv;

}

element.innerText = noww;

let row = element.parentNode.id;

let type = element.getAttribute('type');

dateObj.update(row,type,noww);

}

}

else if(element.nodeName =="BUTTON"){

let parent = element.parentNode.parentNode;

table.removeChild(parent);

let row = element.parentNode.id;

dateObj.del(row);

table.innerHTML = '';

load();

}

}

})

通讯录

window.addEventListener('load',function () {

let dl = document.querySelector('dl');

let search = document.querySelector('header > input')

let aside = document.querySelector('aside');

let tp = document.querySelector('div.tp');

let info = [

{name:'苏凡',tell:'18635625026',py:'sufan'},

{name:'王有帅',tell:'18635625027',py:'wangyoushuai'},

{name:'燕剑豪',tell:'18635625028',py:'yanjianhao'},

{name:'宋浩',tell:'18635625028',py:'songhao'},

{name:'王鹏',tell:'18635625029',py:'wangpeng'},

{name:'黄博伟',tell:'18635625029',py:'huangbowei'},

{name:'白杨',tell:'18635625030',py:'baiyang'},

{name:'许壮壮',tell:'18635625031',py:'xuzhuangzhuang'},

]

/\*按照拼音分下类

\* {

\* 'z'[存入一个数组首字符打头信息]

\*

\* }

\*

\* \*/

/\*搜索\*/

search.onkeyup = function(){

let value = this.value.trim();

let date = info.filter(function(element){

return element.py.includes(value) || element.name.includes(value)||element.tell.includes(value);

})

render (date);

}

render(info)

/\*

滚动

scrollTop +height > = offsettop\*/

let dt = document.querySelectorAll('dt');

let topAll = [];

let heights = document.querySelector('header').offsetHeight + tp.offsetHeight;

dt.forEach(element=>{

topAll.push(element.offsetTop)

})

window.onscroll = function(){

let scrolltop = document.documentElement.scrollTop || document.body.scrollTop;

topAll.forEach((element,index)=>{

if(scrolltop+heights >= element){

tp.innerText = dt[index].innerText;

}

})

}

function render(data){

dl.innerHTML='';

let obj = {};

/\* 遍历找首字符 转换大小写

如果没有创建属性 添加值\*/

data.forEach(function(element){

let first = element.py.charAt(0).toUpperCase();

if(!obj[first]){

obj[first] = [];

}

obj[first].push (element);

})

let char = Object.keys(obj).sort()

tp.innerText = char[0];

char.forEach(element=>{

dl.innerHTML+=`<dt>${element}</dt>`;

aside.innerHTML +=`<li>${element}</li>`

obj[element].forEach(value=>{

dl.innerHTML +=`

<dd><a href="tel:${value.tell}">${value.name}</a></dd>

`

})

})

aside.style.marginTop=`${5}rem`;

/\*滚动\*/

/\* let topAll = [];

dt.forEach(element=>{

topAll.push(element.offsetTop)

})

let tph = tp.offsetHeight;

window.onscroll = function(){

let scrolltop = document.body.scrollTop;

topAll.forEach((element,index) =>{

if(element-scrolltop < tph){

char.forEach(element=>{

tp.innerHTML = dt[index].innertext;

})

}

})

}\*/

}

})

打字游戏

function Game(){

//要产生的字符

this.charArr = [

['Q','AZ/Q.png'],

['W','AZ/W.png'],

['E','AZ/E.png'],

['R','AZ/R.png'],

['T','AZ/T.png'],

['Y','AZ/Y.png'],

['U','AZ/U.png'],

['I','AZ/I.png'],

['O','AZ/O.png'],

['P','AZ/P.png'],

['A','AZ/A.png'],

['S','AZ/S.png'],

['D','AZ/D.png'],

['F','AZ/F.png'],

['G','AZ/G.png'],

['H','AZ/H.png'],

['J','AZ/J.png'],

['K','AZ/K.png'],

['L','AZ/L.png'],

['Z','AZ/Z.png'],

['X','AZ/X.png'],

['C','AZ/C.png'],

['V','AZ/V.png'],

['B','AZ/B.png'],

['N','AZ/N.png'],

['M','AZ/M.png']

]

//存放页面中的元素

this.current = [];

//页面产生的个数

this.number = 5;

this.speed = 10;

/\*关卡\*/

this.gk = 10;

this.score = 0;

this.position = [];

this.life = 6;

this.lifeArr = [];

this.num = 1;

}

Game.prototype = {

start:function(){

for(let i=0;i<this.current.length;i++){

document.body.removeChild(this.current[i]);

}

this.getChars();

this.drop();

this.key();

},

getChars:function(){

// 对页面的产生个数进行遍历

for(let i=0;i<this.number;i++){

this.getChar();

}

},

getChar:function(){

// 随机获取一个

let num = Math.floor(Math.random()\*this.charArr.length);

let divs = document.createElement('div');

/\*去重复\*/

for(let i=0; i<this.current.length; i++){

while(this.current[i].innerText.includes(this.charArr[num][0])){

num = Math.floor(Math.random()\*this.charArr.length);

}

}

divs.innerText = this.charArr[num][0];

divs.classList.add('divs');

let tops = Math.random()\*100;

let lefts = Math.random()\*(innerWidth-400)+200;

while(this.checkPosition(lefts)){

lefts = Math.random()\*(innerWidth-400)+200;

}

this.position.push(lefts);

/\*divs.style.left = `${lefts}px`;

divs.style.top =`${tops}px`;\*/

divs.style.cssText=`

left: ${lefts}px;

top:${tops}px;

background-image:url(${this.charArr[num][1]});

font-size:0;

`

document.body.appendChild(divs);

this.current.push(divs);

},

/\*去重叠\*/

checkPosition:function(lefts){

let flag = this.position.some(function(value){

return Math.abs(value-lefts) < 60

})

return flag;

},

drop:function(){

let that = this

this.t = setInterval(fn,300)

function fn(){

for(let i=0;i<that.current.length;i++){

let tops = that.current[i].offsetTop+that.speed;

that.current[i].style.top = `${tops}px`;

if(tops>=500){

let re = document.body.removeChild(that.current[i]);

that.current.splice(i,1);

that.position.splice(i,1);

let life = that.life -= 1;

let button4 = document.getElementById('button4');

button4.innerText = life;

that.getChar();

that.lifeArr.push(re.innerHTML);

if(that.lifeArr.length>5){

let flag = confirm('是否重新开始');

if(flag){

that.restart();

that.life = 5;

button4.innerText = that.life;

}

}

}

}

}

},

key:function(){

// 键盘按下去 e.keyCode divs.innerText.charCodeAt() String.fromCharCode()

let that = this;

document.onkeydown = function(e){

for(let i=0;i<that.current.length;i++){

if( that.current[i].innerText == String.fromCharCode(e.keyCode)){

that.score+=2;

document.body.removeChild(that.current[i]);

that.current.splice(i,1);

that.position.splice(i,1);

that.getChar();

if(that.score == that.gk){

that.next();

}

}

}

}

},

next:function(){

clearInterval(this.t);

for(let i=0;i<this.current.length;i++){

document.body.removeChild(this.current[i]);

}

this.current.length = 0;

this.position.length = 0;

this.number++;

// this.gk+=10;

this.score = 0;

this.start();

},

left:function(){

for(let i=lifeArr.length;i>=0;i--){

button4.value = i

}

},

restart:function(){

clearInterval(this.t);

for(let i=0;i<this.current.length;i++){

document.body.removeChild(this.current[i]);

}

this.lifeArr.length =0 ;

this.current.length = 0;

this.position.length = 0;

this.life = 6;

this.start();

},

pause:function(){

if(this.num==1){

clearInterval(this.t);

this,num = 0;

}else{

setInterval(fn,300)

this.num = 1;

}

}

}

贪吃蛇

function Snake(){

this.snake = ["3\_0","4\_0","5\_0"];

this.sence = document.querySelector('div.sence');

this.flag = {'3\_0':true,'4\_0':true,'5\_0':true};

this.direction = 40;

this.food ='';

}

Snake.prototype ={

start:function(){

this.drawLine();

this.drawSnake();

this.move();

this.direction = 40;

this.key();

this.dropfood();

},

drawLine:function(){

for(let i=0;i<20;i++){

for(let j=0;j<20;j++){

this.sence.innerHTML += `<div class="block" id="${i}\_${j}"></div>`

}

}

},

drawSnake:function(){

this.snake.forEach(element=>{

document.getElementById(element).classList.add('hot');

})

},

move:function(){

let that = this;

this.t = setInterval(function(){

let oldt = that.snake[that.snake.length-1];

let arr = oldt.split('\_')

let newt ='';

/\*新头坐标 = = 食物 新头 dropfood\*/

if(that.direction==37){

newt = `${arr[0]\*1}\_${arr[1]\*1-1}`;

}else if(that.direction==38){

newt = `${arr[0]\*1-1}\_${arr[1]\*1}`;

}else if(that.direction==39){

newt = `${arr[0]\*1}\_${arr[1]\*1+1}`;

}else if(that.direction==40){

newt = `${arr[0]\*1+1}\_${arr[1]\*1}`;

}

let newarr = newt.split('\_');

if( newarr[1]<0 || newarr[1]>19 || newarr[0]<0 || newarr[0]>19 || that.flag[newt]){

clearInterval(that.t);

alert('Game Over');

}

if(newt == that.food){

that.snake.push(newt);

that.flag[newt] = true;

document.getElementById(that.food).style.background = '';

that.dropfood();

}else{

that.snake.push(newt);

that.flag[newt] = true;

let weiba = that.snake.shift();

delete that.flag[weiba];

document.getElementById(weiba).classList.remove('hot');

}

/\*that.snake.push(newt);

let weiba = that.snake.shift();

document.getElementById(weiba).classList.remove('hot');\*/

that.drawSnake();

},300)

},

key:function(){

document.onkeydown = function(e){

let keycode = e.keyCode;

if(Math.abs(keycode-this.direction) == 2){

return;

}

this.direction = keycode;

}.bind(this)

},

dropfood:function(){

let x = Math.floor(Math.random()\*20);

let y = Math.floor(Math.random()\*20);

do{

x = Math.floor(Math.random()\*20);

y = Math.floor(Math.random()\*20);

}while(this.flag[`${x}\_${y}`])

this.food = `${x}\_${y}`;

document.getElementById(this.food).style.background='red';

},

/\*eatfood:function{

this.dropfood();

if(newt){

}

}\*/

}

倒计时

<body>

<!-- <span></span>天

<span></span>时

<span></span>分

<span></span>秒 -->

<span class='span1'></span>分

<span class='span2'></span>秒

</body>

<script>

/\*计算时间差\*/

/\*setInterval(fn,1000);

function fn(){

let span = document.querySelectorAll('span');

let arr = [];

let now = new Date();

let zq = new Date('2017/10/4/ 0:0:0');

let cha = Math.floor((zq.getTime()-now.getTime())/1000);

let day = Math.floor(cha/(60\*60\*24));

arr.push(day);

cha = cha%(60\*60\*24);

let hours = Math.floor(cha/(60\*60));

arr.push(hours);

cha = cha%(60\*60);

let min = Math.floor(cha/60);

arr.push(min);

cha = cha%60;

let sec = cha;

arr.push(sec);

arr.forEach((element,index)=>{

span[index].innerText = element;

})

}\*/

/\*10分钟倒计时\*/

let old = new Date();

let time = 10\*60;

let old1 = Math.floor(old.getTime()/1000)+time;

setInterval(fn,1000);

function fn(){

let now = new Date();

span1 = document.querySelector('.span1');

span2 = document.querySelector('.span2');

let cha = Math.floor(old1-now.getTime()/1000);

let min = Math.floor(cha/60);

cha = cha%60;

let sec = cha;

span1.innerText = min;

span2.innerText = sec;

}

/\*3分钟之后的时间

let after = new Date();

after.setMinutes(after.getMinutes()+3)\*/

</script>

画板



class palette{

constructor(canvas,ctx,opacity){

this.canvas = canvas;

this.ctx = ctx;

this.opacity = opacity;

this.cw = this.canvas.width;

this.ch = this.canvas.height;

/\*样式\*/

this.lineWidth = 1;

this.lineCap = 'butt';

/\*描边或者填充\*/

this.style = 'fill';

/\*颜色\*/

this.fillStyle = '#000';

this.strokeStyle = '#000';

/\*历史记录\*/

this.history = [];

this.box = document.querySelector('.box');

/\*caiqi\*/

this.tem = null;

};

line(cx,cy,ox,oy){

this.ctx.setLineDash([5,0])

this.ctx.beginPath();

this.ctx.moveTo(cx,cy);

this.ctx.lineTo(ox,oy);

this.ctx.stroke();

};

dash(cx,cy,ox,oy){

this.ctx.setLineDash([5,10])

this.ctx.beginPath();

this.ctx.moveTo(cx,cy);

this.ctx.lineTo(ox,oy);

this.ctx.stroke();

};

circle(cx,cy,ox,oy){

let r = Math.sqrt(Math.pow(cx-ox,2) +Math.pow(cy-oy,2));

this.ctx.beginPath();

this.ctx.arc(cx, cy, r, 0, Math.PI\*2);

this.ctx.closePath();

this.ctx.stroke();

};

rect(cx,cy,ox,oy){

this.ctx.beginPath();

this.ctx.rect(cx, cy, ox-cx, oy-cy);

this.ctx.closePath();

this.ctx.stroke();

};

poly(cx,cy,ox,oy,n){

let r = Math.sqrt(Math.pow(cx-ox,2) +Math.pow(cy-oy,2));

this.ctx.beginPath();

this.ctx.moveTo(cx+r,cy);

let rad = Math.PI\*2/n;

for(let i=0;i<n;i++){

let x = cx+r\*Math.cos(rad\*i),y = cy+r\*Math.sin(rad\*i);

this.ctx.lineTo(x, y);

}

this.ctx.closePath();

this.ctx.stroke();

};

polyJ(cx,cy,ox,oy,n){

let r = Math.sqrt(Math.pow(cx-ox,2) +Math.pow(cy-oy,2));

this.ctx.beginPath();

this.ctx.moveTo(cx+r,cy);

let rad = Math.PI/n;

for(let i=0;i<2\*n;i++){

let r1;

r1 = i%2 ==0 ? r:r/2;

let x = cx+r1\*Math.cos(rad\*i),y = cy+r1\*Math.sin(rad\*i);

this.ctx.lineTo(x, y);

}

this.ctx.closePath();

this.ctx.stroke();

};

pencil(){

this.opacity.style.display = 'none';

this.canvas.onmousedown = function(e){

let cx = e.offsetX ,cy = e.offsetY;

this.ctx.beginPath();

this.ctx.moveTo(cx, cy);

this.canvas.onmousemove = function(e){

let ox = e.offsetX , oy = e.offsetY;

this.ctx.clearRect(0,0,this.cw,this.ch);

if(this.history.length){

this.ctx.putImageData(this.history[this.history.length-1],0,0)

}

// this.ctx[this.strokeStyle]();

this.ctx.lineTo(ox,oy);

this.ctx.stroke();

}.bind(this)

this.canvas.onmouseup = function(){

this.history.push(this.ctx.getImageData(0, 0, this.cw, this.ch))

this.canvas.onmousemove = null;

this.canvas.onmouseup = null;

}.bind(this)

}.bind(this)

};

draw(type,num){

this.opacity.style.display = 'none';

this.canvas.onmousedown = function(e){

let cx = e.offsetX, cy = e.offsetY;

this.canvas.onmousemove = function(e){

let ox = e.offsetX,oy = e.offsetY;

this.ctx.clearRect(0,0,this.cw,this.ch);

if(this.history.length){

this.ctx.putImageData(this.history[this.history.length-1],0,0)

}

this.ctx.setLineDash([5,0]);

this[type](cx,cy,ox,oy,num);

this.ctx[this.style]();

this.ctx.strokeStyle=this.strokeStyle;

this.ctx.fillStyle = this.fillStyle;

}.bind(this)

this.canvas.onmouseup = function(){

this.history.push(this.ctx.getImageData(0, 0, this.cw, this.ch))

this.canvas.onmousemove = null;

this.canvas.onmouseup = null;

}.bind(this)

}.bind(this)

};

/\*橡皮擦\*/

eraser(){

this.opacity.style.display='block';

this.opacity.onmousedown = function(e){

e.preventDefault();

this.box.style.display = 'block';

this.opacity.onmousemove = function(e){

e.preventDefault();

let ox = e.offsetX-15, oy = e.offsetY-15;

/\*边界\*/

let ex = this.box.style.left = `${ox}px`;

let ey = this.box.style.top = `${oy}px`;

this.ctx.clearRect(ox, oy, 30, 30)

}.bind(this)

this.opacity.onmouseup = function(){

this.history.push(this.ctx.getImageData(0, 0, this.cw, this.ch))

this.box.style.display = 'none';

this.opacity.onmousemove = null;

this.opacity.onmouseup = null;

}.bind(this)

}.bind(this)

};

/\*字体\*/

font(){

let that = this;

let lefts = 0 ,tops=0;

this.opacity.style.display='block';

this.opacity.onmousedown = function(e){

that.opacity.onmousedown = null;

let divs = document.createElement('div');

let cx = e.offsetX,cy = e.offsetY;

divs.contentEditable = true;

divs.style.cssText = `

width:100px;height:30px;

border:1px dashed #ccc;

position:absolute;top:${cy}px;left:${cx}px;

cursor:move;

`;

this.appendChild(divs);

divs.onmousedown = function(e){

let cx = e.clientX,cy = e.clientY;

let left = divs.offsetLeft,top = divs.offsetTop;

that.opacity.onmousemove = function(e){

let ox = e.clientX,oy = e.clientY;

lefts = left+ox-cx,tops = top+oy-cy;

if(lefts<=0){

lefts=0;

}

if(lefts>=that.cw-100){

lefts = that.cw-100;

}

divs.style.left = `${lefts}px`;

divs.style.top = `${tops}px`;

}

divs.onmouseup = function(){

that.opacity.onmousemove = null;

this.onmouseup = null;

}

}

divs.onblur = function(){

let value = this.innerText;

that.opacity.removeChild(divs);

that.ctx.font = 'bold 20px sans-serif';

that.ctx.textAlign ='center';

that.ctx.textBaseLine='middle';

that.ctx.fillText(value,lefts,tops);

that.history.push(that.ctx.getImageData(0,0,that.cw,that.ch))

}

}

};

/\*撤销\*/

cx(){

if(!this.history.length){return};

this.history.pop();

this.ctx.clearRect(0, 0, this.cw, this.ch);

this.ctx.putImageData(this.history[this.history.length-1],0,0);

};

/\*描边\*/

miaobian(){

this.style = 'stroke';

};

/\*填充\*/

tianchong(){

this.style= 'fill';

};

/\*描边颜色\*/

miaocolor(c1){

c1.onblur = function(){

this.strokeStyle = c1.value;

}.bind(this)

};

/\*填充颜色\*/

tiancolor(c2){

c2.onblur = function(){

this.fillStyle = c2.value;

}.bind(this)

}

clip(obj){

let that = this;

let minX,minY,w,h;

this.opacity.style.display='block';

obj.style.display='block';

let re = obj.style.width ;

// console.log(re);

this.opacity.onmousedown = function(e){

obj.style.display='block';

obj.style.width = 0;

obj.style.height = 0;

let cx = e.offsetX , cy=e.offsetY;

that.opacity.onmousemove = function(e){

let ox = e.offsetX,oy = e.offsetY;

w = Math.abs(cx-ox);

h = Math.abs(cy-oy);

minX = ox>=cx? cx : ox;

minY = oy>=cy? cy : oy;

obj.style.left = `${minX}`;

obj.style.top = `${minY}`;

obj.style.width = `${w}`;

obj.style.height = `${h}`;

}

that.opacity.onmouseup = function(){

that.tem = that.ctx.getImageData(minX,minY,w,h);

that.ctx.clearRect(minX, minY, w, h);

that.history.push(that.ctx.getImageData(0,0,that.cw,that.ch));

that.ctx.putImageData(that.tem,minX,minY)

that.opacity.onmousemove = null;

that.opacity.onmouseup =null;

that.drag(minX,minY,obj);

}

}

};

drag(x,y,obj){

let that = this;

this.opacity.style.display='block';

this.opacity.onmousedown = function(e){

e.preventDefault();

let cx = e.offsetX,cy = e.offsetY;

that.opacity.onmousemove = function(e){

e.preventDefault();

let ox = e.offsetX,oy = e.offsetY;

let lefts = x + ox-cx , tops = y + oy-cy;

that.ctx.clearRect(0, 0, that.cw, that.ch);

if (that.history.length) {

that.ctx.putImageData(that.history[that.history.length-1],0,0)

}

obj.style.left = `${lefts}px`;

obj.style.top = `${tops}px`;

that.ctx.putImageData(that.tem,lefts,tops);

}

that.opacity.onmouseup = function(){

that.history.push(that.ctx.getImageData(0,0,that.cw,that.ch));

that.tem = null;

obj.style.display = 'none';

that.opacity.onmousemove = null;

that.opacity.onmouseup = null;

}

}

};

/\*清空\*/

clearAll(){

this.ctx.clearRect(0, 0, this.cw, this.ch);

this.history.push(this.ctx.getImageData(0,0,this.cw,this.ch));

}

/\*保存\*/

/\*反向\*/

reverse(){

let imagedata = this.ctx.getImageData(0, 0, this.cw, this.ch);

for(let i=0;i<imagedata.data.length;i+=4){

imagedata.data[i] = 255-imagedata.data[i];

imagedata.data[i+1] =255-imagedata.data[i+1];

imagedata.data[i+2] = 255-imagedata.data[i+2];

/\*灰色\*/

/\*imagedata.data[i] = (imagedata.data[i]+imagedata.data[i+1]+imagedata.data[i+2])/3;\*/

}

this.ctx.putImageData(imagedata,0,0);

this.history.push(this.ctx.getImageData(0,0,this.cw,this.ch));

}

}

小米

index.js



\*

\* @Author: Administrator

\* @Date:   2017-09-19 18:04:54

\* @Last Modified by:   Administrator

\* @Last Modified time: 2017-09-30 10:09:08

\*/

window.onload =function(){

​

/\*中间轮播图\*/

let section = $('.section')[0];

let imgs = $('.banner',section);

let circle = $('.circle')[0];

let lis = $('.lis',circle);

let arrow = $('.arrow')[0];

let arrow1 = $('.arrow1')[0];

​

/\*头部和侧边的抓元素\*/

let nav = document.getElementsByClassName('nav')[0];

let cons = nav.getElementsByClassName('cons');

let liss = nav.getElementsByTagName('li');

let aside = document.getElementsByClassName('aside')[0];

let alis = aside.getElementsByTagName('li');

let item = aside.getElementsByClassName('item');

let item1 = aside.getElementsByClassName('item1');

let item2 = aside.getElementsByClassName('item2');

let item3 = aside.getElementsByClassName('item3');

​

/\*头部和侧边抓元素结束\*/

​

/\*小米明星单品\*/

let danpin1 = document.getElementById('danpin1');

let danpin2 = document.getElementById('danpin2');

let mingxing = document.getElementsByClassName('mingxing')[0];

let xingw = 1226;

/\*小米明星单品结束\*/

​

​

// console.log(lis);

let t = setInterval(fn,4000);

let num = 0;

/\*图片自动播放\*/

function fn(){

num++;

if(num == imgs.length){

num = 0;

}

for(let i=0;i<imgs.length;i++){

// imgs[i].style.display ='none'

animate(imgs[i],{opacity:0});

lis[i].style.background = '#111114'

}

// imgs[num].style.display = 'block'

animate(imgs[num],{opacity:1});

lis[num].style.background = '#6a6a6e'

}

​

/\*点击圆圈更换图片\*/

for(let j=0;j<imgs.length;j++){

lis[j].onclick = function(){

// num = j

clearInterval(t);

for(let i=0;i<imgs.length;i++){

// imgs[i].style.display ='none'

animate(imgs[i],{opacity:0});

lis[i].style.background = '#111114'

}

// imgs[num].style.display = 'block'

animate(imgs[j],{opacity:1});

lis[j].style.background = '#6a6a6e'

num = j;

}

lis[j].onmouseout = function(){

clearInterval(t);

t = setInterval(fn,4000);

}

}

​

/\*左箭头点击更换图片\*/

function fnL(){

num--;

if(num < 0){

num = imgs.length-1;

}

for(let i=0;i<imgs.length;i++){

// imgs[i].style.display ='none'

animate(imgs[i],{opacity:0});

lis[i].style.background = '#111114'

}

// imgs[num].style.display = 'block'

animate(imgs[num],{opacity:1});

lis[num].style.background = '#6a6a6e'

}

arrow.onmouseout = function(){

clearInterval(t);

t = setInterval(fn,4000);

}

arrow.onclick = function(){

fnL()

}

​

/\*右箭头点击更换图片\*/

function fnR(){

num++;

if(num>imgs.length-1){

num = 0;

}

for(let i=0;i<imgs.length;i++){

// imgs[i].style.display ='none'

animate(imgs[i],{opacity:0});

lis[i].style.background = '#111114'

}

// imgs[num].style.display = 'block'

animate(imgs[num],{opacity:1});

lis[num].style.background = '#6a6a6e'

}

arrow1.onmouseout = function(){

clearInterval(t);

t = setInterval(fn,4000);

}

arrow1.onclick = function(){

fnR()

}

​

​

// console.log(item3);

​

/\*头部的导航框\*/

for(let i=0;i<liss.length;i++){

liss[i].onmouseover = function(){

cons[i].style.display = 'block';

}

liss[i].onmouseout = function(){

cons[i].style.display = 'none';

}

}

​

​

/\*侧部导航框\*/

alis[0].onmouseover = function(){

item[0].style.display = 'block';

}

alis[0].onmouseout = function(){

item[0].style.display = 'none';

}

alis[1].onmouseover = function(){

item1[0].style.display = 'block';

}

alis[1].onmouseout = function(){

item1[0].style.display = 'none';

}

alis[2].onmouseover = function(){

item[1].style.display = 'block';

}

alis[2].onmouseout = function(){

item[1].style.display = 'none';

}

alis[3].onmouseover = function(){

item2[0].style.display = 'block';

}

alis[3].onmouseout = function(){

item2[0].style.display = 'none';

}

alis[4].onmouseover = function(){

item1[1].style.display = 'block';

}

alis[4].onmouseout = function(){

item1[1].style.display = 'none';

}

alis[5].onmouseover = function(){

item2[1].style.display = 'block';

}

alis[5].onmouseout = function(){

item2[1].style.display = 'none';

}

for(let i=6;i<alis.length;i++){

alis[i].onmouseover = function(){

for(let j=0;j<item3.length;j++){

item3[j].style.display = 'block';

}

}

alis[i].onmouseout = function(){

for(let j=0;j<item3.length;j++){

item3[j].style.display = 'none';

}

}

}

​

​

/\*小米明星单品\*/

danpin1.onclick = function(){

  mingxing.style.left = `${-xingw}px`;

  danpin1.style.color = '#B1B4BC';

  danpin2.style.color = '#e0e0e0';

}

danpin2.onclick = function(){

mingxing.style.left = `${0}px`;

danpin2.style.color = '#B1B4BC';

danpin1.style.color = '#e0e0e0';

}

let mingmove = 1;

setInterval(function(){

if(mingmove == 1){

mingxing.style.left = `${-xingw}px`;

danpin1.style.color = '#B1B4BC';

danpin2.style.color = '#e0e0e0';

mingmove = 0;

}else if(mingmove == 0){

mingxing.style.left = `${0}px`;

danpin2.style.color = '#B1B4BC';

danpin1.style.color = '#e0e0e0';

mingmove = 1

}

},10000)

​

​

​

​

/\*

左右动画

新确定位置

要及时跟新

\*/

/\*let now = 0;

let next = 0;

let flag = false;

let t = setInterval(move,1000);

let imgw = parseInt(getComputedStyle(section,null).width);

function move(){

next++;

if(next == imgs.length){

next = 0;

}

imgs[next].style.left = `${imgw}px`

animate(imgs[now],{left:-imgw});

animate(imgs[next],{left:0},function(){

flag = true;

});

lis[now].style.background = '#111114';

lis[next].style.background = '#6a6a6e';

now = next;

}

function mover(){

next--;

if(next < 0){

next = imgs.length-1;

}

imgs[next].style.left = `${-imgw}px`

animate(imgs[now],{left:imgw});

animate(imgs[next],{left:0},function(){

flag = true;

});

lis[now].style.background = '#111114';

lis[next].style.background = '#6a6a6e';

now = next;

}

arrow.onmouseover = function(){

clearInterval(t);

}

arrow.onmouseout = function(){

t = setInterval(move,1000);

}

arrow1.onmouseover = function(){

clearInterval(t);

}

arrow1.onmouseover = function(){

t = setInterval(move,1000);

}

arrow.onclick = function(){

if(!flag){

return;

}

mover();

flag = false;

}

arrow1.onclick = function(){

if(!flag){

return;

}

move();

flag = false;

}

for(let i=0;i<lis.length;i++){

lis[i].onclick = function(){

clearInterval(t);

next++;

if(next == imgs.length){

next = 0;

}

imgs[next].style.left = `${imgw}px`

animate(imgs[now],{left:-imgw});

animate(imgs[next],{left:0});

lis[now].style.background = '#111114';

lis[i].style.background = '#6a6a6e';

now = next = i;

}

lis[i].onmuseout = function(){

t = setInterval(move,1000);

}

}\*/

let ss = document.querySelector('div.ss');

let ss1 = document.querySelector('div.ss1');

ss.onmouseover = function(){

ss1.style.display = 'block';

}

ss.onmouseout = function(){

ss1.style.display = 'none';

}

}

content.js



window.addEventListener('load',function(){

let btn = document.querySelector('div.icon');

let btn1 = document.querySelector('div.icon1');

let content = document.querySelector('div.content0');

let ul = document.getElementById('con10');

let lis =ul.getElementsByTagName('li');

/\*btn.onclick = function(){

content.style.left = `{-cw}px`;

content.style.left = 0;

}\*/

  let cw = 296;

let i = 2

let num = 0;

btn1.onclick = function(){

if(i==0){

return;

}

i--;

num++;

if(num ==3){

num =0

}

content.style.left = `${-cw\*i}px`;

for(let i=0;i<lis.length;i++){

lis[i].style.background = '#B0B0B0';

lis[i].style.border = '2px solid #fff';

}

lis[num].style.background = '#fff';

      lis[num].style.border = '2px solid #FF7010';

}

btn.onclick = function(){

if(i==2){

return;

}

i++;

if(num == 0){

num = 3

}

num--;

content.style.left = `${-cw\*i}px`;

for(let i=0;i<lis.length;i++){

lis[i].style.background = '#B0B0B0';

lis[i].style.border = '2px solid #fff';

}

lis[num].style.background = '#FFf';

lis[num].style.border = '2px solid #FF7010';

}

​

​

/\*吴亦凡主题\*/

  let btn2 = document.querySelector('div.icon2');

  let btn3 = document.querySelector('div.icon3');

  let content1 = document.querySelector('div.content1');

  let ul1 = document.getElementById('con11');

  let lis1 =ul1.getElementsByTagName('li');

  let cw1 = 296;

  let i1 = 0

  let num1 = 0;

  btn3.onclick = function(){

      if(i1==3){

          return;

      }

      i1++;

      num1++;

      if(num1 ==4){

          num1 =0

      }

      content1.style.left = `${-cw1\*i1}px`;

      for(let i=0;i<lis.length;i++){

          lis1[i].style.background = '#B0B0B0';

          lis1[i].style.border = '2px solid #fff';

      }

      lis1[num1].style.background = '#fff';

      lis1[num1].style.border = '2px solid #FF7010';

  }

  btn2.onclick = function(){

      if(i1==0){

          return;

      }

      i1--;

      if(num1 == 0){

          num1 = 4

      }

      num1--;

      content1.style.left = `${-cw1\*i1}px`;

      for(let i=0;i<lis.length;i++){

          lis1[i].style.background = '#B0B0B0';

          lis1[i].style.border = '2px solid #fff';

      }

      lis1[num1].style.background = '#fff';

      lis1[num1].style.border = '2px solid #FF7010';

  }

​

  /\* 游戏\*/

​

  let btn4 = document.querySelector('div.icon4');

  let btn5 = document.querySelector('div.icon5');

  let content2 = document.querySelector('div.content3');

  let ul2 = document.getElementById('con12');

  let lis2 =ul2.getElementsByTagName('li');

  console.log(btn4)

  /\*btn.onclick = function(){

      content.style.left = `{-cw}px`;

      content.style.left = 0;

  }\*/

  let cw2 = 296;

  let i2 = 2

  let num2 = 0;

  btn5.onclick = function(){

      if(i2==0){

          return;

      }

      i2--;

      num2++;

      if(num2 ==3){

          num2 =0

      }

      content2.style.left = `${-cw2\*i2}px`;

      for(let i=0;i<lis.length;i++){

          lis2[i].style.background = '#B0B0B0';

          lis2[i].style.border = '2px solid #fff';

      }

      lis2[num2].style.background = '#fff';

      lis2[num2].style.border = '2px solid #FF7010';

  }

  btn4.onclick = function(){

      if(i2==2){

          return;

      }

      i2++;

      if(num2 == 0){

          num2 = 3

      }

      num2--;

      content2.style.left = `${-cw2\*i2}px`;

      for(let i=0;i<lis.length;i++){

          lis2[i].style.background = '#B0B0B0';

          lis2[i].style.border = '2px solid #fff';

      }

      lis2[num2].style.background = '#fff';

      lis2[num2].style.border = '2px solid #FF7010';

  }

  /\*应用\*/

  let btn6 = document.querySelector('div.icon6');

  let btn7 = document.querySelector('div.icon7');

  let content3 = document.querySelector('div.content5');

  let ul3 = document.getElementById('con13');

  let lis3 =ul3.getElementsByTagName('li');

  let cw3 = 296;

  let i3 = 0

  let num3 = 0;

  btn7.onclick = function(){

      if(i3==3){

          return;

      }

      i3++;

      num3++;

      if(num3==4){

          num3=0

      }

      content3.style.left = `${-cw3\*i3}px`;

      for(let i=0;i<lis.length;i++){

          lis3[i].style.background = '#B0B0B0';

          lis3[i].style.border = '2px solid #fff';

      }

      lis3[num3].style.background = '#fff';

      lis3[num3].style.border = '2px solid #FF7010';

  }

  btn6.onclick = function(){

      if(i3==0){

          return;

      }

      i3--;

      if(num3 == 0){

          num3 = 4

      }

      num3--;

      content3.style.left = `${-cw3\*i3}px`;

      for(let i=0;i<lis.length;i++){

          lis3[i].style.background = '#B0B0B0';

          lis3[i].style.border = '2px solid #fff';

      }

      lis3[num3].style.background = '#fff';

      lis3[num3].style.border = '2px solid #FF7010';

  }

​

})

inte.js



/\*window.addEventListener('load',function(){

let inte = document.querySelector('.inte1-1');

let inli = inte.getElementsByTagName('li');

let inte1 = document.querySelectorAll('#inte');

console.log(inte1);

let in1li = inte1.getElementsByTagName('li');

for(let i=0;i<inli.length;i++){

inli[i].onclick = function(){

for(let i=0;i<in)

}

}

})

\*/

jiadian.js



/\*

\* @Author: Administrator

\* @Date:   2017-09-23 10:08:18

\* @Last Modified by:   Administrator

\* @Last Modified time: 2017-09-25 22:14:26

\*/

window.addEventListener('load',function(){

let elec = document.getElementById('elec');

let elec1 = elec.getElementsByTagName('li');

let elec2 = document.querySelector('.elec2');

let elec3 = elec2.querySelectorAll('div.elec2-right');

let elecnum = 0;

for(let i=0;i<elec1.length;i++){

elec1[i].onmouseover = function(){

if(elecnum == 4){

elecnum = 0;

}

elecnum = i

for(let i=0;i<elec3.length;i++){

elec3[i].style.display = 'none';

}

elec3[elecnum].style.display = 'block';

elec3[elecnum].style.transitionDelay=`${i\*100}ms`;

elecnum++

​

}

}

})

recommend.js



window.addEventListener('load',function(){

​

let re1 = document.getElementById('re1');

let re2 = document.getElementById('re2');

let re3 = document.querySelector('div.re');

let rew = 1226;

re1.onclick = function(){

  re3.style.left = `${-rew}px`;

  re1.style.color = '#B1B4BC';

  re2.style.color = '#e0e0e0';

}

re2.onclick = function(){

re3.style.left = `${0}px`;

re2.style.color = '#B1B4BC';

re1.style.color = '#e0e0e0';

}

let mingmove = 1;

setInterval(function(){

if(mingmove == 1){

re3.style.left = `${-rew}px`;

re1.style.color = '#B1B4BC';

re2.style.color = '#e0e0e0';

mingmove = 0;

}else if(mingmove == 0){

re3.style.left = `${0}px`;

re2.style.color = '#B1B4BC';

re1.style.color = '#e0e0e0';

mingmove = 1

}

},10000)

})

天猫

index.js



window.onload = function(){

/\*侧导航\*/

let ban = document.getElementsByClassName('ban4')[0];

let ban1 =ban.getElementsByTagName('li');

let item = document.getElementsByClassName('item');

for(let i=0;i<ban1.length;i++){

ban1[i].onmouseover = function(){

item[i].style.display = 'block';

}

ban1[i].onmouseout = function(){

item[i].style.display = 'none';

}

}

/\*轮播图\*/

let ban5 = document.getElementsByClassName('ban5')[0];

let ban5li =ban5.getElementsByTagName('li');

let ban2 = document.getElementsByClassName('ban2')[0];

let ban2img =ban2.getElementsByTagName('img');

let ban6 = document.getElementsByClassName('ban')[0];

/\* console.log(ban6);\*/

let banarr = ['#E8E8E8','#FF7B54','#A9020C','#FF9601','#0B3472','#8429D2']

let c = 0;

let result =setInterval(fn,4000)

function fn(){

c++;

if(c == 6){

c = 0;

}

for(let i=0;i<6;i++){

animate(ban2img[i],{opacity:0})

// ban2img[i].style.display = 'none';

ban5li[i].style.background = 'rgba(162,162,162,0.7)';

}

animate(ban2img[c],{opacity:1})

// ban2img[c].style.display = 'block';

ban5li[c].style.background = 'rgba(255,255,255,0.7)';

ban6.style.background = banarr[c];

}

​

for(let j=0;j<ban5li.length;j++){

ban5li[j].onmouseover = function(){

c = j;

clearInterval(result);

for(let z=0;z<6;z++){

animate(ban2img[z],{opacity:0})

/\*ban2img[z].style.display = 'none';\*/

ban5li[z].style.background = 'rgba(162,162,162,0.7)';

}

animate(ban2img[c],{opacity:1})

// ban2img[c].style.display = 'block';

ban5li[c].style.background = 'rgba(255,255,255,0.7)';

ban6.style.background = banarr[c];

}

ban5li[j].onmouseout = function(){

/\*setInterval(fn,1000);\*/

result = setInterval(fn,4000);

}

}

​

/\*

左右动画

now 长度

next的长度

相互变换

\*/

/\*用var的时候在 需要添加一个属性 ban5li[j].index = j\*/

/\*for(let i=0;i<ban5li.length;i++){

ban5li[i].onmouseover = function(){

ban5li[i].style.background = 'rgba(255,255,255,0.7)'

}

ban5li[i].onmouseout = function(){

ban5li[i].style.background = 'rgba(162,162,162,0.7)'

ban2img[j].style.display = 'none'

}

}\*/

/\*窗口   和 点击 \*/

/\*let now = 0;

for(let i = 0;i<ban5li.length;i++){

ban5li[i].onclick = function(){

ban2img[now].style.display = 'none';

ban2img[i].style.display = 'block';

now = i;

}

}\*/

/\*利用闭包函数进行执行\*/

/\* var now = 0;

for(var i=0;i<ban5li.length;i++){

ban5li[i].index = i;

ban5li[i].onclick = (function(){

return function(){

ban2img[now].style.display = 'none';

ban2img[this.index].style.display = 'block';

now = this.index;

}

})(i);

}\*/

​

}

header.js



window.addEventListener('load',function(){

  let mytao = document.querySelector('.mytao');

  let mytao2 = document.querySelector('div.mytao2');

  let shou = document.querySelector('.shou');

  let shou2 = document.querySelector('div.shou2');

  let zhi = document.querySelector('.zhi');

  let zhi2 = document.querySelector('div.zhi2');

  let wang1 = document.querySelector('.wang1');

  let wang2 = document.querySelector('div.wang2');

  mytao.addEventListener('mouseover',function(){

      mytao.style.background = '#fff';

      mytao2.style.display = 'block';

  })

  mytao.addEventListener('mouseout',function(){

      mytao.style.background = '#F2F2F2'

      mytao2.style.display = 'none';

  })

  shou.addEventListener('mouseover',function(){

      shou.style.background = '#fff';

      shou2.style.display = 'block';

  })

  shou.addEventListener('mouseout',function(){

      shou.style.background = '#F2F2F2'

      shou2.style.display = 'none';

  })

  zhi.addEventListener('mouseover',function(){

      zhi.style.background = '#fff';

      zhi2.style.display = 'block';

  })

  zhi.addEventListener('mouseout',function(){

      zhi.style.background = '#F2F2F2'

      zhi2.style.display = 'none';

  })

  wang1.addEventListener('mouseover',function(){

      wang1.style.background = '#fff';

      wang2.style.display = 'block';

  })

  wang1.addEventListener('mouseout',function(){

      wang1.style.background = '#F2F2F2'

      wang2.style.display = 'none';

  })

​

​

​

})

load.js



window.addEventListener('load',function(){

  let near = document.querySelector('.near');

  let near1 = document.querySelectorAll('.near > li');

  let fx = document.querySelector('div.fx');

  let ding = document.getElementById('ding');

  console.log(near1)

  let ch = innerHeight;

  let floor = document.querySelectorAll('div.floor');

  let floor1 = document.querySelector('.love');

  let floorArr= [];

  let flag = true;

  let color = ['#64C333','#FF0036','#EA5F8D','#0AA6E8','#64C333','#F15453','#19C8A9','#FF0036'];

  floor.forEach(element=>{

floorArr.push(element.offsetTop);

})

  console.log(floorArr)

  near1.forEach((element,index)=>{

element.onclick =function(){

​

for(let i=0;i<near1.length;i++){

near1[i].style.background = '#626262';

}

element.style.background = color[index];

flag = false;

document.documentElement.scrollTop = floorArr[index]-100;

​

animate(document.body,{scrollTop:floorArr[index]},function(){

flag = true;

});

}

})

  ding.onclick = function(){

  animate(fx,{opacity:0})

  document.documentElement.scrollTop = ding.offsetHeight-100;

  }

  window.onscroll = function(){

if(!flag){

return;

}

let scrolltop = document.documentElement.scrollTop;

if(scrolltop>=500){

animate(near,{opacity:1})

animate(fx,{opacity:1})

}

if(scrolltop<=500){

animate(near,{opacity:0})

animate(fx,{opacity:0})

}

floorArr.forEach((elements,index)=>{

if(scrolltop + ch >= elements+400){

  // let imgs = floor[index].getElementsByTagName('img');

for(let i=0;i<near1.length;i++){

near1[i].style.background = '#626262';

}

near1[index].style.background = color[index];

}

})

}

​

​

​

})