**第一章：**

⼀、填空题

1、// 2、undefined 3、ECMAScript DOM BOM 4、<script>alert(123);</script>

⼆、判断题

1、错 2、错 3、对 4、错 5、对 6、对

三、选择题

1、C 2、C 3、B

四、编程题

JavaScript代码如下：

<script>

var msg = prompt("请您输⼊内容");

alert('⽤户输⼊的内容是：'+ msg)

</script>

**第二章:**

一、填空题

1、基础数据类型 复杂数据类型（引用数据类型） 2、-2 3、2

二、判断题

1、对 2、错 3、错

三、选择题

1、B 2、D 3、D

四、编程题

1、JavaScript代码如下：

<script>

var number = prompt('请输入1-7之间的数字');

switch (number) {

case '7':

alert('星期日');

break;

case '6':

alert('星期六');

break;

case '5':

alert('星期五');

break;

case '4':

alert('星期四');

break;

case '3':

alert('星期三');

break;

case '2':

alert('星期二');

break;

case '1':

alert('星期一');

break;

default:

alert('请输入1~7之间的数字');

}

</script>

2、JavaScript代码如下：

<script>

var num1 = prompt('请输入第1个数');

var num2 = prompt('请输入第2个数');

if(parseFloat(num1) > parseFloat(num2)){

alert('最大的数是'+ num1)

}else{

alert('最大的数是'+ num2)

}

</script>

**第三章：**

一、填空题

1、34 2、continue

二、判断题

1、对 2、错 3、对

三、选择题

1、D 2、C 3、C

四、编程题

1、JavaScript代码如下：

<script>

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

var newArr=[];

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

if(arr[i] != 2){

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

}

}

console.log(newArr);

</script>

2、JavaScript代码如下：

<script>

var arr = ['苹果', '香蕉', '酥梨', '榴莲', '樱桃', '柿子', '葡萄', '哈密瓜', '西瓜'];

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

console.log(arr[i]);

}

</script>

第四章：

一、填空题

1、函数表达式 2、函数作用域 3、1999

二、判断题

1、错 2、错 3、对

三、选择题

1、B 2、A 3、D

四、程序分析题

1、代码运行结果如下。

<script>

var a, b;

(function() {

alert(a); // undefined

alert(b); // undefined

var a = b = 3;

alert(a); // 3

alert(b); // 3

})();

alert(a); // undefined

alert(b); // 3

</script>

2、代码运行结果：-1

<script>

var foo = function(x, y) {

return x - y;

};

function foo(x, y) {

return x + y;

}

var num = foo(1, 2);

console.log(num); // 结果为：-1

</script>

**第五章**

一、填空题

1、in 2、random() 3、Date() 4、getMonth()

二、判断题

1、对 2、错 3、对 4、对

三、选择题

1、A 2、A 3、A

四、编程题

1、JavaScript代码如下：

<script>

var str = ' sdsdsd '

console.log(str.trim());

</script>

2、JavaScript代码如下：

<script>

var dog = {

dName: '可可',

type: '阿拉斯加犬',

age: '5岁',

color: '棕红色',

call: function() {

console.log('汪汪叫');

},

tail: function() {

console.log('摇尾巴');

}

};

console.log(dog.dName);

console.log(dog.type);

console.log(dog.age);

console.log(dog.color);

dog.call();

dog.tail();

</script>

**第六章**：

一、填空题

1、事件源 事件类型 事件处理程序

2、getElementById()

3、innerHTML innerText textContent

二、判断题

1、对 2、对 3、错 4、错

三、选择题

1、A 2、B 3、C 4、D 5、A

四、编程题

JavaScript代码如下：

<body>

<div></div>

<script>

// 根据系统不同时间来判断，所以需要用到日期内置对象

// 利用多分支语句来设置不同的图片

// 1.获取元素

var div = document.querySelector('div');

// 2. 得到当前的小时数

var h = new Date().getHours();

// 3. 判断小时数改变文字信息

if(h < 6){

document.write('凌晨好')

}else if (h < 9){

document.write("早上好!")

}else if (h < 12){

document.write("上午好!")

}else if (h < 14){

document.write("中午好!")

}else if (h < 17){

document.write("下午好!")

}else if (h < 19){

document.write("傍晚好!")

}else if (h < 22){

document.write("晚上好!")

}else {

document.write("夜里好!")

}

</script>

</body>

**第七章：**

一、填空题

1、所有元素全部清除 设置当前元素

2、data-属性名

3、element.dataset.属性（或 element.dataset['属性'] ）

4、children

5、注释节点

二、判断题

1、对 2、对 3、对 4、错 5、对

三、选择题

1、D 2、D 3、D 4、A 5、C

四、简答题

1、请简单介绍排他操作的一般实现步骤。

答：第一步是：所有元素全部清除（排除掉其他的（包括自己））

第二步是：设置当前元素（给自己设置想要实现的效果）

2、请解释说明childNodes和children的区别。

答：childNodes：属性获得的是当前元素的所有子节点的集合，该集合为即时更新的集合

children: 是一个可读的属性，返回所有子元素节点，只返回子元素节点，其余节点不返回

五、编程题

1、HTML代码如下：

<table cellspacing="0">

<thead>

<tr>

<th>姓名</th>

<th>科目</th>

<th>成绩</th>

<th>操作</th>

</tr>

</thead>

<tbody>

</tbody>

</table>

2、JavaScript代码如下：

<script>

// 1.先去准备好学生的数据

var datas = [{

name: '张三',

subject: 'JavaScript',

score: 100

}, {

name: '李四',

subject: 'JavaScript',

score: 90

}, {

name: '刘五',

subject: 'JavaScript',

score: 90

}];

// 2. 往tbody 里面创建行： 有几个人（通过数组的长度）我们就创建几行

var tbody = document.querySelector('tbody');

for (var i = 0; i < datas.length; i++) { // 外面的for循环管行 tr

// 1. 创建 tr行

var tr = document.createElement('tr');

tbody.appendChild(tr);

// 2. 行里面创建单元格(跟数据有关系的3个单元格) td 单元格的数量取决于每个对象里面的属性个数 for循环遍历对象 datas[i]

for (var k in datas[i]) { // 里面的for循环管列 td

// 创建单元格

var td = document.createElement('td');

// 把对象里面的属性值 datas[i][k] 给 td

// console.log(datas[i][k]);

td.innerHTML = datas[i][k];

tr.appendChild(td);

}

// 3. 创建有删除2个字的单元格

var td = document.createElement('td');

td.innerHTML = '<a href="javascript:;">删除 </a>';

tr.appendChild(td);

}

// 4. 删除操作开始

var as = document.querySelectorAll('a');

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

as[i].onclick = function() {

// 点击a 删除 当前a 所在的行(链接的爸爸的爸爸) node.removeChild(child)

tbody.removeChild(this.parentNode.parentNode)

}

}

</script>

3、CSS样式代码如下（要显示与作业截图一致的效果需添加CSS代码，不添加CSS代码不影响功能）：

<style>

body{

width: 100%;

margin: o auto;

height: 300px;

position: fixed;

}

table {

width: 500px;

height: auto;

position: relative;

left:50%;

margin-left: -250px;

justify-content: center;

align-content: center;

align-items: center;

}

tr{

display: flex;

justify-content: center;

align-content: center;

align-items: center;

border-bottom: 1px solid #333;

text-align: center;

}

td,

th {

flex:1;

}

thead tr {

height: 40px;

background-color: #ccc;

}

</style>

**第八章：**

一、填空题

1、window

2、window.onload

3、length

4、document.DOMContentLoaded

二、判断题

1、对 2、对 3、对 4、对

三、选择题

1、A 2、D 3、A

四、编程题

编写功能代码，具体代码如下：

<a id = "time"></a>

<button onclick = "clearInterval(time1)">停止</button>

<script type="text/javascript">

var time1 = window.setInterval('timeShow()', 1000);

timeShow()

function timeShow() {

var today = new Date(); //获得系统当前时间

var intYears = today.getFullYear(); //获得年

intMonths = today.getMonth() + 1; //获得月份+1

intDates = today.getDate(); //获得天数

intHours = today.getHours(); //获得小时

intMinutes = today.getMinutes(); //获得分钟

intSeconds = today.getSeconds(); //获得秒

intWeeks = today.getDay(); //获得星期

// 修改格式为 \*年\*月\*日

years = intYears + '年';

// 如果月份小于10，在月份前面加0

intMonths < 10 ? months = '0' + intMonths + '月' : months = intMonths + '月';

// 如果日份小于10，在日前面加0

intDates < 10 ? dates = '0' + intDates + '日' : dates = intDates + '日';

// 修改星期格式

var week = ['星期日', '星期一', '星期二', '星期三', '星期四', '星期五', '星期六']

weeks = week[intWeeks] + ' ';

// 修改小时格式

if (intHours == 0) {

hours = '00:';

} else if (intHours < 10) {

hours = '0' + intHours + ':';

} else {

hours = intHours + ":";

}

// 修改分钟格式

if (intMinutes == 0) {

minutes = '00';

} else if (intMinutes < 10) {

minutes = '0' + intMinutes;

} else {

minutes = intMinutes+ ":";

}

// 修改秒数格式

if (intSeconds == 0) {

seconds = '00';

} else if (intSeconds < 10) {

seconds = '0' + intSeconds;

} else {

seconds = intSeconds;

}

var timeString = years + months + dates + weeks + hours + minutes+seconds

document.getElementById('time').innerHTML = timeString;

}

</script>

**第九章：**

一、填空题

1、offsetTop

2、offsetParent

3、内容区域的宽度（width）

二、判断题

1、对 2、对 3、错

三、选择题

1、ABC 2、D 3、D

四、简答题

1、请简述offset系列属性有哪些。

offsetLeft：返回元素相对带有定位父元素左边框的偏移。

offsetTop：返回元素相对带有定位父元素上方的偏移。

offsetWidth：返回自身的宽度（包括padding、边框和内容区域的宽度），注意返回数值不带单位。

offsetHeight：返回自身的高度（包括padding、边框和内容区域的高度），注意返回数值不带单位。

offsetParent：返回作为该元素带有定位元素的父级元素（如果父级都没有定位则返回body）。

2、请简述offset和style的区别。

offset可以得到任意样式表中的样式值，而style只能得到行内样式表中的样式值。

offset系列获得的数值是没有单位的，而style.width获得的是带有单位的字符串。

offsetWidth包含padding、border、width的值，而style.width获得的是不包含padding、border的值。

offsetwidth等属性是只读属性，只能获取不能赋值，而style.width是可读写属性，可以获取也可以赋值。

3、请简述offsetParent和parentNode的区别。

offsetParent：返回作为该元素带有定位元素的父级元素（如果父级都没有定位则返回body）。

parentNode:用来返回某个指定节点的父节点。

**第十章：**

一、填空题

1、:first

2、fadeIn()

3、:last

4、show()

5、addClass()

二、判断题

1、对 2、对 3、对 4、对 5、错

三、选择题

1、C 2、C 3、A 4、A 5、B

四、简答题

1、请列举jQuery中基本选择器有哪些。

id选择器：获取指定id的元素，语法$("#id")。

全选选择器：匹配所有元素，语法$("\*")。

类选择器：获取同一类class的元素，语法$(".class")。

标签选择器：获取相同标签名的所有元素，语法$("div")。

并集选择器：选取多个元素，语法$("div,p,li")。

交集选择器：交集元素，语法$("li.current")。

2、请列举操作元素类名的方法有哪些。

addClass()方法向被选元素添加一个或多个类名。

removeClass()方法从被选元素移除一个或多个类。

toggleClass()方法用来为元素添加或移除某个类，如果类不存在，就添加该类，如果类存在，就移除该类。

五、编程题

1、请使用jQuery设置页面中的div元素的宽度为200px，高度200px。

//定义div元素：

<div style="background-color:red"></div>

//jQuery代码如下：

$('div').css({width:'200px',height:'200px'});

完整代码如下：

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>Document</title>

//引入jquery文件

<script src = "jquery-3.4.1.min.js"></script>

</head>

<body>

<div style="background-color:red"></div>

<script>

$('div').css({width:'200px',height:'200px'});

</script>

</body>

</html>

2、请使用jQuery实现页面中div元素向右运动100px后回到初始位置的动画效果。

//定义div元素:

<button>向右移动</button>

<div style="background-color:red;

width:50px;

height:50px;

position:absolute;">

</div>

//jQuery代码如下：

$("button").click(function () {

$("div").animate({

left: 100,

}, 500);

});

$("button").click(function () {

$("div").animate({

left: 0,

}, 500);

});

完整代码如下：

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>Document</title>

//注意引入jquery文件

<script src="jquery-3.4.1.min.js"></script>

</head>

<body>

<button>向右移动</button>

<div style="background-color:red;width:50px;height:50px;position:absolute;"></div>

<script>

$("button").click(function () {

$("div").animate({

left: 100,

}, 500);

});

$("button").click(function () {

$("div").animate({

left: 0,

}, 500);

});

</script>

</body>

</html>

第十一章：