**四：JavaScript**

(一) javaScript1

1、JavaScript是一门什么样的语言，它有哪些特点？

回答:

一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。它的解释器被称为JavaScript引擎，为浏览器的一部分，广泛用于客户端的脚本语言，最早是在HTML网页上使用，用来给HTML网页增加动态功能。JavaScript兼容于ECMA标准，因此也称为ECMAScript。

Js具有简单性，动态性，跨平台型，安全性。

2、document load 和document ready的区别

回答:

Document.onload 是在结构和样式加载完才执行js

Document.ready原生种没有这个方法，jquery中有 $().ready(function)

3、”==”和“===”的不同

回答:

前者会自动转换类型，数据相等

后者不会，数据和数据类型相等

4、javascript的typeof返回哪些数据类型

回答:

Object number function boolean underfind

5、form中的input有哪些类型？

回答:

text:文本框，password:密码框，radio:[单选按钮](https://www.baidu.com/s?wd=%E5%8D%95%E9%80%89%E6%8C%89%E9%92%AE&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YLrjIBuhwWmWI9rjwBnHNh0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3En104nj03PWc4" \t "https://zhidao.baidu.com/question/_blank)，checkbox:复选框，file:文件选择域，hidden:隐藏域，button:按钮，reset:重置按钮，submit:表单提交按钮

6、javaScript的2种变量范围有什么不同？

回答:

全局变量：当前页面内有效

局部变量：方法内有效

7、form中的input可以设置为readonly和disable，请问2者有什么区别？

回答:

readonly不可编辑，但可以选择和复制；值可以传递到后台

disabled不能编辑，不能复制，不能选择；值不可以传递到后台

8、js中的3种弹出式消息提醒（警告窗口，确认窗口，信息输入窗口）的命令式什么？

回答:

Alert，confirm，prompt

9、列举javaScript的3种主要数据类型，2种复合数据类型和2种特殊数据类型。

回答:

主要数据类型：string, boolean, number

复合数据类型：function, object

10、程序中捕获异常的方法？

回答:

window.error

try{}catch(){}finally{}

11、浏览器的对象模型？

回答:

Window：顶级对象

History：当用户浏览网页时，浏览器保存了一个最近所访问网页的url列表。这个列表就是用history对象表示。

Location：表示当前打开的窗口或框架的URL信息。

Navigator：表示浏览器的信息及js运行的环境

Screen：用于显示网页的显示器的大小和颜色

12、javascript的常用对象有哪些？

回答:

String, Math, Date和Array对象

13、innerHTML，innerText，outerHTML，innerText的区别？

回答:

innerHTML是符合W3C标准的属性，而innerText、outerText、outerHTML只适用于IE浏览器

innerHTML设置或获取位于对象起始和结束标签内的 HTML

outerHTML设置或获取对象及其内容的 HTML 形式

innerText  设置或获取位于对象起始和结束标签内的文本

outerText  设置(包括标签)或获取(不包括标签)对象的文本

14、超链接的属性target的可选值：\_blank, \_parent, \_self, \_top和框架名称有什么区别？

回答:

\_blank 浏览器会另开一个新窗口显示链接

\_self 在同一框架或窗口中打开所链接的文档。此参数为默认值，通常不用指定。

\_parent 将链接的文件载入含有该链接框架的父框架集或父窗口中。如果含有该链接的框架不是嵌套的，则在浏览器全屏窗口中载入链接的文件，就象\_self参数一样。

\_top在当前的整个浏览器窗口中打开所链接的文档，因而会删除所有框架

15、添加 插入 替换 删除到某个接点的方法，怎样创建和查找节点？

回答:

obj.appendChidl()

obj.innersetBefore insertBefore() //插入

obj.replaceChild

obj.removeChild

1）创建新节点

createDocumentFragment() //创建一个DOM片段

createElement() //创建一个具体的元素

createTextNode() //创建一个文本节点

2）查找

getElementsByTagName() //通过标签名称

getElementsByName() //通过元素的Name属性的值

getElementById() //通过元素Id，唯一性

16、javascript怎样选中一个checkbox，怎样设置它无效？

回答:

document.all.cb1[0].disabled = true;

17、

回答:

(二) javaScript2

1、b继承a的方法

回答:

function A(name){

    this.name = name;

    this.sayHello = function(){alert(this.name+” say Hello!”);};

}

function B(name,id){

    this.temp = A;

    this.temp(name);        //相当于new A();

    delete this.temp;

    this.id = id;

    this.checkId = function(ID){alert(this.id==ID)};

}

2、split() join() 的区别

回答:

前者是切割成数组的形式，后者是将数组转换成字符串

3、call和apply的区别

回答:

call方法: 调用一个对象的一个方法，以另一个对象替换当前对象。

apply方法：应用某一对象的一个方法，用另一个对象替换当前对象。

4、Javascript中callee和caller的作用？

回答:

caller是返回一个对函数的引用，该函数调用了当前函数；

callee是返回正在被执行的function函数，也就是所指定的function对象的正文。

5、数组方法pop() push() unshift() shift()

回答:

Push()尾部添加 pop()尾部删除

Unshift()头部添加 shift()头部删除

6、例举3种强制类型转换和2种隐式类型转换?

回答:

强制（parseInt,parseFloat,number）

隐式（== ===）

7、事件绑定和普通事件有什么区别

回答:

普通添加事件的方法不支持添加多个事件，最下面的事件会覆盖上面的，而事件绑定（addEventListener）方式添加事件可以添加多个。

8、JS和JAVA的作用域有什么区别

回答:

java是块级作用域，js是函数作用域，没有块级作用域。

9、foo = foo||bar ，这行代码是什么意思？为什么要这样写？

回答:

if(!foo) foo = bar; //如果foo存在，值不变，否则把bar的值赋给foo。

短路表达式：作为”&&”和”||”操作符的操作数表达式，这些表达式在进行求值时，只要最终的结果已经可以确定是真或假，求值过程便告终止，这称之为短路求值。

10、JS继承与原型问题

回答:

Js所有的函数都有一个prototype属性，这个属性引用了一个对象，即原型对象，也简称原型。每一个对象都有原型，在浏览器中它体现在一个隐藏的 proto 属性上。当一个对象需要调用某个方法时，它回去最近的原型上查找该方法，如果没有找到，它会再次往下继续查找。这样逐级查找，一直找到了要找的方法。这些查找的原型构成了该对象的原型链条。原型最后指向的是null。我们说的原型继承，就是将父对像的方法给子类的原型。子类的构造函数中不拥有这些方法和属性

11、javascript的同源策略

回答:

一段脚本只能读取来自于同一来源的窗口和文档的属性，这里的同一来源指的是主机名、协议和端口号的组合

12、事件委托是什么

回答:

让利用事件冒泡的原理，让自己的所触发的事件，让他的父元素代替执行！

13、javascript的本地对象，内置对象和宿主对象

回答:

本地对象为array obj regexp等可以new实例化

内置对象为gload Math 等不可以实例化的

宿主为浏览器自带的document,window 等

14、在Javascript中什么是伪数组？如何将伪数组转化为标准数组？

回答:

伪数组（类数组）：无法直接调用数组方法或期望length属性有什么特殊的行为，但仍可以对真正数组遍历方法来遍历它们。典型的是函数的argument参数，

15、javascript是面向对象的，怎么体现javascript的继承关系？

回答:

使用prototype来实现。

16、当一个DOM节点被点击时候，我们希望能够执行一个函数，应该怎么做？

回答:

直接在DOM里绑定事件：<div onclick=”test()”></div>

在JS里通过onclick绑定：xxx.onclick = test

通过事件添加进行绑定：addEventListener(xxx, ‘click’, test)

17、es6你了解多少

回答:

es6是一个新的标准，它包含了许多新的语言特性和库，是JS最实质性的一次升级。他还引入module模块的概念

18、

回答:

(三) javaScript3

1、想实现一个对页面某个节点的拖曳？如何做？（使用原生JS）。

回答:

需要三个事件：

onmousedown 控制节点扩展的属性droping为true(表示可以移动节点了)

onmousemove 控制节点的位置(判断是否可以移动如果是就移动)

ommouseup控制节点的扩展属性为false(表示不能移动节点了)

<!DOCTYPE HTML>

<html>

<head>

<title>演示</title>

<style type="text/css">

</style>

</head>

<body>

<div id="dropDiv">

  我可以拖动

</div>

<script type="text/javascript">

function dropDivFun(Element){

//设置节点的定位为fixed

    Element.style.position="fixed";

    //获取事件对象函数 方便获取事件的clientX

    function getEvent(){

        if(event){

            e=event;

        }else if(window.event){

            e=window.event;

        }else{

            e=arguments.callee.arguments[0];

        }

        return e;

    }

    //添加mousedown事件

    Element.addEventListener("mousedown",function(){

        var e=getEvent();

        if(!Element.droping){

        //设置droping为true并记录初始的位置

            Element.droping=true;

            Element.start\_x=e.clientX-Element.offsetLeft;

            Element.start\_y=e.clientY-Element.offsetTop;

        }

    });

    Element.addEventListener("mousemove",function(){

        var e=getEvent();

        //移动节点

        if(Element.droping){

            Element.style.left=e.clientX-Element.start\_x+"px";

            Element.style.top=e.clientY-Element.start\_y+"px";

        }

    });

    Element.addEventListener("mouseup",function(){

        if(Element.droping){

        //设置droping为false并释放初始的变量

            Element.droping=false;

            Element.start\_x=null;

            Element.start\_y=null;

        }

    });

}

//为节点添加drop事件

dropDivFun(document.getElementById("dropDiv"));

</script>

</body>

</html>

2、如何阻止冒泡事件和默认事件

回答:

通过cancelBubble事件对象可以阻止冒泡事件

用return false来阻止默认事件

3、

回答:

(四) Ajax与json

1、ajax请求的时候get 和post方式的区别

回答:

一个在url后面 一个放在虚拟载体里面

有大小限制

安全问题

应用不同 一个是论坛等只需要请求的，一个是类似修改密码的

2、什么是Ajax和JSON，它们的优缺点。

回答:

Ajax是异步JavaScript和XML，用于在Web页面中实现异步数据交互。

优点：

　可以使得页面不重载全部内容的情况下加载局部内容，降低数据传输量

　避免用户不断刷新或者跳转页面，提高用户体验

缺点：

　对搜索引擎不友好（

　要实现ajax下的前后退功能成本较大

　可能造成请求数的增加

　跨域问题限制

JSON是一种轻量级的数据交换格式，ECMA的一个子集

优点：轻量级、易于人的阅读和编写，便于机器（JavaScript）解析，支持复合数据类型（数组、对象、字符串、数字）

3、ajax请求时，如何解释json数据

回答:

使用eval parse 鉴于安全性考虑 使用parse更靠谱

4、解释jsonp的原理，以及为什么不是真正的ajax

回答:

动态创建script标签，回调函数

Ajax是页面无刷新请求数据操作

5、JSONP是什么？它是如何实现跨域的？为什么它可以实现跨域？

回答:

JSONP(JSON with Padding)是一个非官方的协议，它允许在服务器端集成Script tags返回至客户端，通过javascript callback的形式实现跨域访问。

由于同源策略的限制，XmlHttpRequest只允许请求当前源（域名、协议、端口）的资源，为了实现跨域请求，可以通过script标签实现跨域请求

（因为同源策略不阻止动态脚本插入，并且将脚本看作是从提供 Web 页面的域上加载的），然后在服务端输出JSON数据并执行回调函数，从而解决了跨域的数据请求。

6、对json有什么了解

回答:

JSON(JavaScript Object Notation) 是一种轻量级的数据交换格式。它是基于JavaScript的一个子集。数据格式简单, 易于读写, 占用带宽小。

7、

回答:

(五) 闭包与关键字this

1、闭包是什么，有什么特性，对页面有什么影响

回答:

闭包就是能够读取其他函数内部变量的函数。

闭包的特性：

①.封闭性：外界无法访问闭包内部的数据，如果在闭包内声明变量，外界是无法访问的，除非闭包主动向外界提供访问接口；  
②.持久性：一般的函数，调用完毕之后，系统自动注销函数，而对于闭包来说，在外部函数被调用之后，闭包结构依然保存在  
系统中，闭包中的数据依然存在，从而实现对数据的持久使用。

优点：

① 减少全局变量。

② 减少传递函数的参数量

③ 封装；  
 缺点：  
 使用闭包会占有内存资源，过多的使用闭包会导致内存溢出等.

2、闭包问题

回答:

闭包：函数能被外部调用到，则该作用连上的所有变量都会被保存下来。

作用：①可以读取函数内部的变量；②相当于划出了一块私有作用域，避免数据污染；③让变量始终保存在内存中

使用全局变量被认为是不好的习惯，而且容易造成错误并且维护成本较高，所以js可以采用闭包的方式读取函数的内部变量。

但是如果大量使用闭包就会造成过多的变量始终保存在内存中，会造成内存泄漏。

一个简单的闭包例子：

function f1(){

var n=999;

function f2(){

alert(n);

}

return f2;

}

var result=f1();

result(); // 999

3、js中this的工作原理

回答:

①全局范围内：指向全局对象；

this;

②函数调用：指向全局对象

foo();

③方法调用：this指向test对象

test.foo();

④调用构造函数：this指向新创建的对象

new foo();

⑤显示的设置this：使用call或者apply方法时，this指向第一个参数

function foo(a, b, c) {}

var bar = {};

foo.apply(bar, [1, 2, 3]); // 数组将会被扩展，如下所示

foo.call(bar, 1, 2, 3); // 传递到foo的参数是：a = 1, b = 2, c = 3

4、谈谈This对象的理解。

回答:

this是js的一个关键字，随着函数使用场合不同，this的值会发生变化。但是有一个总原则，那就是this指的是调用函数的那个对象。this一般情况下：是全局对象Global。 作为方法调用，那么this就是指这个对象

5、

回答:

(六) JQuery框架

1、jQuery与jQuery UI 有啥区别？

回答:

jQuery是一个js库，主要提供的功能是选择器，属性修改和事件绑定等等。

jQuery UI则是在jQuery的基础上，利用jQuery的扩展性，设计的插件。提供了一些常用的界面元素，诸如对话框、拖动行为、改变大小行为等等

1. JQuery的源码看过吗？能不能简单说一下它的实现原理？

回答:

没有什么，就是封装javascript，使调用更简单而已。

3、对Node的优点和缺点提出了自己的看法？

回答:

（优点）因为Node是基于事件驱动和无阻塞的，所以非常适合处理并发请求，因此构建在Node上的代理服务器相比其他技术实现（如Ruby）的服务器表现要好得多。此外，与Node代理服务器交互的客户端代码是由javascript语言编写的，因此客户端和服务器端都用同一种语言编写，这是非常美妙的事情。

（缺点）Node是一个相对新的开源项目，所以不太稳定，它总是一直在变，而且缺少足够多的第三方库支持。看起来，就像是Ruby/Rails当年的样子。

4、你有哪些性能优化的方法？

回答:

（1） 减少http请求次数：CSS Sprites, JS、CSS源码压缩、图片大小控制合适；网页Gzip，CDN托管，data缓存，图片服务器。

（2） 前端模板 JS+数据，减少由于HTML标签导致的带宽浪费，前端用变量保存AJAX请求结果，每次操作本地变量，不用请求，减少请求次数

（3） 用innerHTML代替DOM操作，减少DOM操作次数，优化javascript性能。

（4） 当需要设置的样式很多时设置className而不是直接操作style。

（5） 少用全局变量、缓存DOM节点查找的结果。减少IO读取操作。

（6） 避免使用CSS Expression（css表达式)又称Dynamic properties(动态属性)。

（7） 图片预加载，将样式表放在顶部，将脚本放在底部，加上时间戳。

（8） 避免在页面的主体布局中使用table，table要等其中的内容完全下载之后才会显示出来，显示比div+css布局慢。

5、

回答:

(七) IE

1、IE和DOM事件流的区别

回答:

1.执行顺序不一样、

2.参数不一样

3.事件加不加on

4.this指向问题

2、IE和标准下有哪些兼容性的写法

回答:

Var ev = ev || window.eventdocument.documentElement.clientWidth || document.body.clientWidth

Var target = ev.srcElement||ev.target

3、

回答:

(八) 代码

1、使用原生js实现ajax

回答:

创建一个 XMLHttpRequest 对象。如果不支持创建该对象的浏览器，则需要创建 ActiveXObject，具体方法如下：

var xmlHttpRequest;

function createXmlHttpRequest(){

if(window.XMLHttpRequest)//非IE

xmlHttpRequest = new XMLHttpRequest();

else if(window.ActiveObject)//IE6+

xmlHttpRequest = new ActiveObject("Msxml2.XMLHTTP");

else//IE6-

xmlHttpRequest = new ActiveObject("Microsoft.XMLHTTP");

}

提交请求的方法：

open(method,url);//分别为提交的方法（GET或者POST）和提交的url

send(content);

onreadystatechange(){

if(xmlHttpRequest.readyState == 4){

if(xmlHttpRequest.state == 200){

//请求成功

}

}else{

//请求失败

}

}

2、JavaScript的数据类型都有什么？

回答:

基本数据类型：String,boolean,Number,Undefined, Null

引用数据类型：Object(Array,Date,RegExp,Function)

3、编写一个数组去重的方法

回答:

function oSort(arr)

{

var result ={};

var newArr=[];

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

{

if(!result[arr])

{

newArr.push(arr)

result[arr]=1

}

}

return newArr

}

4、如何消除一个数组里面重复的元素？

回答:

var arr1 =[1,2,2,2,3,3,3,4,5,6],

arr2 = [];

for(var i = 0,len = arr1.length; i< len; i++){

if(arr2.indexOf(arr1[i]) < 0){

arr2.push(arr1[i]);

}

}

document.write(arr2); // 1,2,3,4,5,6

5、统计字符串”aaaabbbccccddfgh”中字母个数或统计最多字母数

回答:

var str = "aaaabbbccccddfgh";

var obj = {};

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

var v = str.charAt(i);

if(obj[v] && obj[v].value == v){

obj[v].count = ++ obj[v].count;

}else{

obj[v] = {};

obj[v].count = 1;

obj[v].value = v;

}

}

for(key in obj){

document.write(obj[key].value +'='+obj[key].count+' '); // a=4 b=3 c=4 d=2 f=1 g=1 h=1

}

6、实现对该数组的倒排，输出[5,1,4,2,6,3]/实现对该数组的降序排列，输出[6,5,4,3,2,1]

回答:

function combo(msg){

var arr=msg.split("-");

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

arr[i]=arr[i].charAt(0).toUpperCase()+arr[i].substr(1,arr[i].length-1);

}

msg=arr.join("");

return msg;

}

function combo(msg){

var arr=msg.split("-");

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

arr[i]=arr[i].charAt(0).toUpperCase()+arr[i].substr(1,arr[i].length-1);

}

msg=arr.join("");

return msg;

}

7、用js实现随机选取10–100之间的10个数字，存入一个数组，并排序

回答:

var iArray = [];

funtion getRandom(istart, iend){

var iChoice = istart - iend +1;

return Math.floor(Math.random() \* iChoice + istart;

}

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

iArray.push(getRandom(10,100));

}

iArray.sort();

var iArray = [];

funtion getRandom(istart, iend){

var iChoice = istart - iend +1;

return Math.floor(Math.random() \* iChoice + istart;

}

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

iArray.push(getRandom(10,100));

}

iArray.sort();

8、写一个function，清除字符串前后的空格。（兼容所有浏览器）

回答:

function trim(str) {

if (str && typeof str === "string") {

return str.replace(/(^\s\*)|(\s\*)$/g,""); //去除前后空白符

}

}

9、实现一个函数clone，可以对JavaScript中的5种主要的数据类型（包括Number、String、Object、Array、Boolean）进行值复制。

回答:

/\*\*

\* 对象克隆

\* 支持基本数据类型及对象

\* 递归方法

\*/

function clone(obj) {

var o;

switch (typeof obj) {

case "undefined":

break;

case "string":

o = obj + "";

break;

case "number":

o = obj - 0;

break;

case "boolean":

o = obj;

break;

case "object": // object 分为两种情况 对象（Object）或数组（Array）

if (obj === null) {

o = null;

} else {

if (Object.prototype.toString.call(obj).slice(8, -1) === "Array") {

o = [];

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

o.push(clone(obj[i]));

}

} else {

o = {};

for (var k in obj) {

o[k] = clone(obj[k]);

}

}

}

break;

default:

o = obj;

break;

}

return o;

}

10、已知ID的Input输入框，希望获取这个输入框的输入值，怎么做？(不使用第三方框架)

回答:

document.getElementById(“ID”).value

1

document.getElementById(“ID”).value

11、设置一个已知ID的DIV的html内容为xxxx，字体颜色设置为黑色(不使用第三方框架)

回答:

var dom = document.getElementById(“ID”);

dom.innerHTML = “xxxx”

dom.style.color = “#000”

var dom = document.getElementById(“ID”);

dom.innerHTML = “xxxx”

dom.style.color = “#000”

12、求y和z的值是多少？

<script type="text/javascript">

var x = 1;

var y = 0;

var z = 0;

function add(n){n=n+1;}

y = add(x);

function add(n){n=n+3;}

z = add(x);

</script>

回答:

都为undefined，因为没有返回值

13、写出函数DateDemo的返回结果，系统时间假定为今天

function DateDemo(){

var d, s="今天日期是：";

d = new Date();

s += d.getMonth() + "/";

s += d.getDate() + "/";

s += d.getYear();

return s;

}

回答:

今天日期是：7/17/2010

14、写出程序运行的结果？

for(i=0, j=0; i<10, j<6; i++, j++){

k = i + j;

}

回答:

10（小心陷阱）

15、运行的结果？

function hi(){

var a;

alert(a);

}

function hi(){

var a = null;

alert(a);

}

回答:

undefined

null

16、

回答: