Winter

http://note.youdao.com/share/?id=8eb4bf3f0e86251b3bef5bca7841ccb3&type=note#/

阿里

1. Display中inline和inline-block的区别，常见块状，内联标签

常见的块级元素有 DIV, FORM, TABLE, P, PRE, H1~H6, DL, OL, UL 等

常见的内联元素有 SPAN, A, STRONG, EM, LABEL, INPUT, SELECT, TEXTAREA, IMG, BR 等。

block

block元素会独占一行，多个block元素会各自新起一行。block元素可以设置width,height属性。块级元素即使设置了宽度,仍然是独占一行。block元素可以设置margin和padding属性。

Inline

inline元素不会独占一行，多个相邻的行内元素会排列在同一行里，直到一行排列不下，才会新换一行，其宽度随元素的内容而变化。inline元素设置width,height属性无效。inline元素的margin和padding属性，水平方向的padding-left, padding-right, margin-left, margin-right都产生边距效果；但竖直方向的padding-top, padding-bottom, margin-top, margin-bottom不会产生边距效果。

inline-block

简单来说就是将对象呈现为inline对象，但是对象的内容作为block对象呈现。之后的内联对象会被排列在同一行内。比如我们可以给一个link（a元素）inline-block属性值，使其既具有block的宽度高度特性又具有inline的同行特性

1. Display还有其他那些属性，区别和联系

none：此元素不会被显示；block：此元素将显示为块级元素；inline：此元素会被显示为内联元素；inline-block：行内块元素；flex：弹性布局，定义一个flex容器，将对象作为弹性伸缩盒显示；inline-flex：弹性布局，定义一个flex容器，将对象作为内联块级弹性伸缩盒显示；list-item：指定对象为列表项目。inline-block：指定对象为内联块元素。table：指定对象作为块元素级的表格。类同于html标签<table>；inline-table：指定对象作为内联元素级的表格。类同于html标签<table>；table-caption：指定对象作为表格标题。类同于html标签<caption>；table-cell：指定对象作为表格单元格。类同于html标签<td>；table-row：指定对象作为表格行。类同于html标签<tr>；table-row-group：指定对象作为表格行组。类同于html标签<tbody>；table-column：指定对象作为表格列。类同于html标签<col>；table-column-group：指定对象作为表格列组显示。类同于html标签<colgroup>；table-header-group：指定对象作为表格标题组。类同于html标签<thead>；table-footer-group：指定对象作为表格脚注组。类同于html标签<tfoot>；run-in：根据上下文决定对象是内联对象还是块级对象。

1. 简述Css盒模型

IE 盒模型：IE的content部分包含了 border 和 padding；

标准 W3C 盒模型：内容(content)、填充(padding)、边框(border)、边界(margin)；

1. Css浮动的概念，常见的清浮动的方法

浮动元素脱离文档流，不占据空间。浮动元素碰到包含它的边框或者浮动元素的边框停留。

清浮动的方法：

1. 父级div定义 height。

原理：父级div手动定义height，就解决了父级div无法自动获取到高度的问题。

优点：简单，代码少，容易掌握

缺点：只适合高度固定的布局，要给出精确的高度，如果高度和父级div不一样时，会产生问题

建议：不推荐使用，只建议高度固定的布局时使用

2. 使用空标签清除浮动。

这种方法是在所有浮动标签后面添加一个空标签 定义css clear:both. 弊端就是增加了无意义标签。

3. 使用overflow。

给包含浮动元素的父标签添加css属性 overflow:auto; zoom:1; zoom:1用于兼容IE6。

4．父级div定义 overflow:hidden

原理：必须定义width或zoom:1，同时不能定义height，使用overflow:hidden时，浏览器会自动检查浮动区域的高度

优点：简单，代码少，浏览器支持好

缺点：不能和position配合使用，因为超出的尺寸的会被隐藏。

建议：只推荐没有使用position或对overflow:hidden理解比较深的朋友使用。

5. 使用after伪对象清除浮动。

该方法只适用于非IE浏览器。具体写法可参照以下示例。使用中需注意以下几点。一、该方法中必须为需要清除浮动元素的伪对象中设置 height:0，否则该元素会比实际高出若干像素；

6．父级div 也一起浮动

1. 如何处理跨域问题，跨域通信

1. jsonp（jsonp 的原理是动态插入 script 标签）

2. document.domain + iframe

3. window.name、window.postMessage

4. 服务器上设置代理页面

1. Jsonp的概念，如何使用
2. Http协议的概念，http请求的方式
3. Http请求报结构是什么样的
4. HTML5的新标签有哪些

10、localStorage和sessionStorage的区别和联系

11、如何进行css优化，从几方面回答后问道一些细节比如为什么用link等

12、如何进行网站优化，同上个问题问道为什么把script放在页面尾部等

13、Ajax的概念，如何实现通信得

14、简单介绍下js事件流

15、问了简历上的项目，简单介绍问道项目中遇到最大的困难，如何自己解决

16、因为简历上写了node问了node一些基础知识

17、MVC框架的结构

18、如何学习前端，最近在看什么书

19、简单的自我介绍，大神直奔主题让我把做过的项目一个接着一个仔细的介绍给他听，在讲的过程中会针对性的提问如

20、Css3实现动画的方式有哪些

21、Transition的属性有哪些

22、Transform的使用方式

23、如何在项目中新加一些动画，比如在入场动画后紧接着抖动或者翻转

24、说一说canvas画图的基本步骤

25、介绍下js适配各种屏幕的方法

26、中间真的是各种头脑风暴各种煎熬啊几万字......

27、然后问了js的继承的方式

28、Js创建对象的方式

29、介绍下js原型链

30、Js继承几种方式的优劣

31、了解MVVM框架么，简单介绍几个和特性

蘑菇街

1. 从项目出发问，事件绑定和事件委托
2. Js如何实现浏览器适配
3. Js的闭包
4. Js的匿名函数
5. Css中line-height的用法
6. 如何居中一个div元素
7. 知道BFC么，介绍下BFC
8. 如何实现css响应式
9. 对css优先级的理解

10、问了项目简单几个问题

11、Js数组去重

12、了解现在前端什么新技术，有什么展望

百度

1. 介绍一下你所知道的CSS选择器
2. 谈谈对元素应用样式优先级的了解（浏览器默认样式、继承样式、选择 器特殊性、标签内联样式、!important样式）
3. CSS选择器书写优化
4. 函数作用域
5. 函数作用域链
6. 闭包
7. 函数柯里化
8. Ajax是什么，它改变了什么
9. Ajax如何跨域请求

10、Ajax如何传输文件

11、什么是Promise，带来了什么编程体验

12、如何预加载图片

13、如何懒加载图片

14、如何进行节流

15、前端模块化解决的哪些前端问题

16、cmd/amd/commonJS规范

17、HTTP状态码及其含义

18、吧啦把啦说半天项目

19、css3animation

20、js实现项目效果衍生

21、canvas曲线效果

22、js实现tab和背景联动效果

23、ajax的实现过程

24、ajax的get和post方法传参的不同

25、js事件的机制，什么是冒泡

26、js的事件委托

27、如何判断js事件冒泡后事件作用在那个子元素上

28、写一个数组去重

HTML + CSS

1. 对WEB标准以及W3C的理解与认识?

2、xhtml和html有什么区别?

3、Doctype? 严格模式与混杂模式-如何触发这两种模式，区分它们有何意义?

4、行内元素有哪些?块级元素有哪些?CSS的盒模型?

5、CSS引入的方式有哪些? link和@import的区别是?

6、CSS选择符有哪些?哪些属性可以继承?优先级算法如何计算?内联和important哪个优先级高?

7、前端页面有哪三层构成，分别是什么?作用是什么?

8、css的基本语句构成是?

9、你做的页面在哪些流览器测试过?这些浏览器的内核分别是什么?

10、说出几种IE6 BUG的解决方法

11、img标签上title与alt属性的区别是什么?

12、描述css reset的作用和用途。

13、解释css sprites，如何使用，他的优点是什么？

14、浏览器标准模式和怪异模式之间的区别是什么?

15、你如何对网站的文件和资源进行优化?期待的解决方案包括：

16、什么是语义化的HTML?

17、清除浮动的几种方式，各自的优缺点

18、css hack

19、超链接访问过后hover样式就不出现的问题?

20、IE6的双倍边距BUG

21、为什么FF下文本无法撑开容器的高度?

22、为什么web标准中IE无法设置滚动条颜色了?

23、为什么无法定义1px左右高度的容器?

24、怎么样才能让层显示在FLASH之上呢?

25、怎样使一个层垂直居中于浏览器中?

26、firefox嵌套div标签的居中问题的解决方法

27、网页制作中有哪几种样式设置方法？各有何特点？

28、用户自定义的类和ID在定义和使用时有什么区别?

29、float元素的父元素不能指定clear属性

30、是否重设了默认的样式?

31、简述一下src与href的区别。

32、知道的网页制作会用到的图片格式有哪些？

33、每个HTML文件里开头都有个很重要的东西，Doctype，知道这是干什么的吗？

34、行内元素有哪些?块级元素有哪些?CSS的盒模型?

35、css层叠是什么？介绍一下。

36、position值都有哪些，css3都有哪些新内容…

HTML5+CSS3常见面试题目

37. SGML（标准通用标记语言）和HTML（超文本标记语言），XML（可扩展标记语言）和HTML的之间有什么关系？

38. 什么是HTML5？

39. 为什么HTML5里面我们不需要DTD（Document Type Definition文档类型定义）？如果我不放入<! DOCTYPE html>标签，HTML5还会工作么？

40. 哪些浏览器支持HTML5？

41. HTML5的页面结构同HTML4或者更前的HTML有什么区别？

42. HTML5中的datalist是什么？

43. HTML5中哪些是不同的新的表单元素类型？

44. HTML5中什么是输出元素？

45. 什么是SVG（Scalable Vector Graphics可缩放矢量图形）？

46. 我们能看到使用HTML5的SVG的简单例子么？

47. HTML5中canvas是什么？

48. 我们如何使用Canvas来画一条简单的线？

49. Canvas和SVG图形之间的区别是什么？

50. 如何使用Canvas和HTML5中的SVG去画一个矩形？

51. CSS（cascading style sheets级联样式表）中的选择器是什么？

52. 如何使用ID值来应用一个CSS样式？

53. CSS中使用列布局是什么？

54. 你能解释一下CSS的盒子模型么？

55. 你能解释一些CSS3中的文本效果么？

56. 什么是Web Workers？为什么我们需要他们？

57. Web Worker线程的限制是什么？

58. 我们如何在JavaScript中创建一个worker线程？

59. 如何中止Web Worker?

60. 为什么我们需要HTML5的服务发送事件？

61. HTML5中的本地存储概念是什么？

62. 我们如何从本地存储中添加和移除数据？

63. 本地存储的生命周期是什么？

64. 本地存储和cookies（储存在用户本地终端上的数据）之间的区别是什么？

65. 什么是事务存储？我们如何创建一个事务存储？

66. 本地存储和事务存储之间的区别是什么？

67. 什么是WebSQL？

68. WebSQL 是HTML5的一个规范吗？

69. 我们如何使用WebSQL？

70. HTML5中的应用缓存是什么？

71. HTML5中我们如何实现应用缓存？

72. 我们如何刷新浏览器的应用缓存？

73. 应用缓存中的回退是什么？

74. 应用缓存中的网络是什么？

75. 如何在 HTML5 页面中嵌入音频?

76. 响应式布局用什么实现？

77. 如何在 HTML5 页面中嵌入视频？

78. HTML5 有哪些不同类型的存储？

79. HTML5 引入什么新的表单属性？

80. HTML5 标准提供了哪些新的 API？

81. HTML5的页面结构

82. HTML5的页面结构同HTML4或者更前的HTML有什么区别？

83. 浏览器本地存储

84. web storage和cookie的区别

85. CSS中 link 和@import 的区别是？

86. CSS3有哪些新特性？

87. 对前端界面工程师这个职位是怎么样理解的？它的前景会怎么样？

88. 最近在学什么？能谈谈你未来3，5年给自己的规划吗？

89. HTML5中什么是输出元素？

JavaScript常见面试题目

1：javascript的typeof返回哪些数据类型?

typeof 运算符把数据类型信息当作字符串返回。typeof 返回值有六种可能： "number," "string," "boolean," "object," "function," 和 "undefined."

alert(typeof null); // "object"

alert(typeof function () { return 1; }); // "function"

alert(typeof '梦龙小站'); // "string"

alert(typeof 1); // "number"

alert(typeof a); // "undefined"

alert(typeof undefined); // "undefined"

alert(typeof []); // "object"

2：如何判断某变量是否为数组数据类型？

JavaScript中检测对象的方法：

1.instanceof操作符 --> 这个操作符和JavaScript中面向对象有点关系，了解这个就先得了解JavaScript中的面向对象。因为这个操作符是检测对象的原型链是否指向构造函数的prototype对象的。

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

alert(arr instanceof Array); // true

2.对象的constructor属性 -->除了instanceof，每个对象还有constructor的属性，利用它似乎也能进行Array的判断。

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

alert(arr.constructor === Array); // true

3.Array.isArray() --> ECMAScript5将Array.isArray()正式引入 JavaScript，目的就是准确地检测一个值是否为数组。IE9+、 Firefox 4+、Safari 5+、Opera 10.5+和Chrome都实现了这个方法。但是在IE8之前的版本是不支持的。

注意：typeof操作符与instanceof操作符区别

1.typeof操作符

这种方法对于一些常用的类型来说那算是毫无压力，比如Function、String、Number、Undefined等，但是要是检测Array的对象就不起作用了。

2.Instanceof运算符

此运算符可以判断一个变量是否是某个对象（类）的实例，返回值是布尔类型的。

3.例举3种强制类型转换和2种隐式类型转换？

4.split() join() 的区别?

Join 函数获取一批字符串，然后用分隔符字符串将它们联接起来，从而返回一个字符串。

Split 函数获取一个字符串，然后在分隔符处将其断开，从而返回一批字符串。

主要区别： Join 可以使用任何分隔符字符串将多个字符串连接起来，而 Split 只能使用一个字符分隔符将字符串断开。如果你用split，是把一串字符（根据某个分隔符）分成若干个元素存放在一个数组里。而Join是把数组中的字符串连成一个长串，可以大体上认为是split的逆操作。

5.数组方法pop() push() unshift() shift()

6：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？

7：Javascript的事件流模型都有什么？

8.IE和DOM事件流的区别?

事件流是描述页面接收事件的顺序。

具体的有：事件冒泡，事件捕获，DOM事件流。

1、事件冒泡又叫IE的事件流，即事件开始时由最具体的元素（文档中嵌套层次最深的那个节点）接收，然后逐级向上传播到较为不具体的节点（文档）。

2、事件捕获的思想是不太具体的节点应该更早的接收到事件，而最具体的节点应该在最后接收到节点。事件捕获的用意在于事件到达预定目标之前捕获它。

3、DOM2级事件流规定的事件流包括三个阶段：事件捕获阶段、处于目标阶段和冒泡阶段。首先发生的是事件捕获，为截获事件提供了机会。然后是实际的目标接收到事件。最后一个阶段是冒泡阶段，可以在这个阶段对事件作出响应。

9.IE和标准下有哪些兼容性的写法?

10.ajax请求的时候get 和post方式的区别?

11.call和apply的区别?

12.ajax请求时，如何解释json数据?

13：Ajax同步和异步的区别，如何解决跨域问题?

14.写一个获取非行间样式的函数?

15.事件委托是什么?

16.闭包是什么，有什么特性，对页面有什么影响?

17.如何阻止事件冒泡和默认事件?

18. 添加、移除、移动、复制、创建和查找节点的方法?

19.解释jsonp的原理，以及为什么不是真正的ajax?

20.javascript的本地对象，内置对象和宿主对象?

21.document load 和document ready的区别?

22.”==”和“===”的不同?

23.javascript的同源策略?

24.编写一个数组去重的方法?

26.排序算法?

27：希望获取到页面中所有的checkbox怎么做？(不使用第三方框架) ?

28：什么是Ajax和JSON，它们的优缺点?

29：有这样一个URL：http://item.taobao.com /item.htm?a=1&b=2&c=&d=xxx&e，

请写一段JS程序提取URL中的各个GET参数(参数名和 参数个数不确定)，将其按key-value形式返回到一个json结构中，如{a:’1′, b:’2′ , c:”, d:’xxx’, e:undefined}?

30：正则表达式构造函数var reg=new RegExp(“xxx”)与正则表达字面量var reg=//有什么不同？匹配邮箱的正则表达式？

31：.Javascript中callee和caller的作用？

32： 下面这个ul，如何点击每一列的时候alert其index?（闭包）？

<ul id=”test”>

<li>这是第一条</li>

<li>这是第二条</li>

<li>这是第三条</li>

</ul>

33： 给String对象添加一个方法，传入一个string类型的参数，然后将string的每个字符间价格空格返回，例如：

addSpace(“hello world”) // -> ‘h e l l o w o r l d’

34：函数声明与函数表达式的区别？

function show(){ //声明式函数}

var show = function{ //函数表达式 }

两者的区别：

// 正常调用到函数

alert(sum(2, 6));

function sum(a, b) {

return a + b;

}

执行了声明式函数，在预编译期声明函数及被处理了，所以即使sum( )调用函数放在声明函数前也能执行。

// 不能正常调用到函数，浏览器报错

alert(sum(2, 6));

var sum = function (a, b) {

return a + b;

}

解析：声明式函数与函数表达式的区别在于：在JS的预编译期，声明式函数将会先被提取出来，然后才按顺序执行js代码；函数表达式，则必须等到解析器执行到它所在的代码行，才会真正被解释执行。

35：在Javascript中什么是伪数组？如何将伪数组转化为标准数组？

36：想实现一个对页面某个节点的拖曳？如何做？（使用原生JS）?

37：var numberArray = [3,6,2,4,1,5]; （考察基础API）?

1) 实现对该数组的倒排，输出[5,1,4,2,6,3]

2) 实现对该数组的降序排列，输出[6,5,4,3,2,1]

38：.将字符串”<tr><td>{$id}</td><td>{$name}</td></tr>”中的{$id}替换成10，{$name}替换成Tony （使用正则表达式）?

39：foo = foo||bar ，这行代码是什么意思？为什么要这样写？

40：为了保证页面输出安全，我们经常需要对一些特殊的字符进行转义，请写一个函数escapeHtml，将<, >, &, “进行转义?

41：写一个function，清除字符串前后的空格。（兼容所有浏览器）?

42：Javascript中callee和caller的作用？

43：谈一谈你对javascript的理解，javascript能做什么?

44：在javascript中如何检测客户端的浏览器和操作系统类型?

45：Cookie在客户机上是如何存储的?

46：Prompt box是什么？它的返回值有什么用？

47：如何获取javascript所能表示的最大值和最小值？

48：Null和Undefined有和区别与联系?

49：如何理解javascript中的variable typing，试用代码说明?

50：如何理解javascript中的对象?

\*HTML&CSS：

对Web标准的理解、浏览器内核差异、兼容性、hack;

CSS基本功：布局、盒子模型、选择器优先级及使用、HTML5、CSS3、移动端适应;

\*JavaScript：

数据类型、面向对象、继承、闭包、插件、作用域、跨域、原型链、模块化、自定义事件、

内存泄漏、事件机制、异步装载回调、模板引擎、Nodejs、JSON、ajax等

\*其他：

HTTP、安全、正则、优化、重构、响应式、移动端、团队协作、可维护、SEO、UED、架构、职业生涯

CSS

线路1：

介绍一下你所知道的CSS选择器——》谈谈对元素应用样式优先级的了解（浏览器默认样式、继承样式、选择器特殊性、标签内联样式、!important样式）——》CSS选择器书写优化——》样式渲染树大致原理。

线路2：

大体画一下盒模型——》display有几种值，都是什么，有什么作用——》介绍一下浮动——》容器包含若干浮动元素如何清理浮动——》是否知道BFC，如何创建BFC，有什么作用——》大致行内元素具体表现（行框，行内框，行高，行内盒子顶线，基线，中线，底线，水平垂直对其介绍）——》vertical-align:middle的定义，为什么是近似居中

JS

路线1：

函数作用域——》函数作用域链——》闭包——》函数柯里化

路线2：

如何实现方法链式调用——》JS中this可能是很么，分别说一说——》如何在方法内部定义的函数中使用该方法的this——》function.bind的用法——》实现一个function.bind方法兼容老版本浏览器

路线3：

function.length和argument.length分别是什么——》什么是类数组——》如何将类数组转换成数组——》给数组写一个方法将其从多维数组扁平化

路线4：

JS有哪些方法定义一个对象——》构造函数的特点，与普通函数有什么不同——》prototype是什么，原型指针是什么，实例如何查询属性和方法的——》如何继承一个类——》如何实现组合继承，为什么要组合继承——》实现一个Object.create方法兼容老版本浏览器

路线5：

介绍一下cookie——》同源策略是什么——》介绍一下你所知道的跨域方式——》JSONP的原理——》图片ping能干什么

路线6：

Ajax是什么，它改变了什么——》XHR的readyState所有状态值，分别代表什么——》Ajax如何实现一个异步调用，得到响应修改dom——》Ajax如何跨域请求——》Ajax如何传输文件——》了解哪些异步编程——》什么是Promise，带来了什么编程体验——》Promise.then链式调用的大致原理

综合

1、 从浏览器地址栏输入url到显示页面都发生了什么 (请求、处理、响应、渲染，无限深)

2、 HTTP request报文结构是怎样的

3、 如何进行网站性能优化

4、 HTTP状态码及其含义

编码

1、有一个大数组,var a = ['1', '2', '3', ...];a的长度是100,内容填充随机整数的字符串.请先构造此数组a,然后设计一个算法将其内容去重。

2、将字符串’aa-bb-cc’驼峰化,转换成’aaBbCc’

3、将url中所有search串转化成一个对象，如：从”http://www.engineer.com? direction=%E5%89%8D%E7%AB%AF%E5%BC%80%E5%8F%91&skills=html%2Ccss%2Cjs”得到{direction:”前端工程师”,skills:”html,css,js”}

4、写一个js对象深度拷贝

6、实现一个具有简单的观察者模式的类，在类中有on,off,trigger方法。

跨域：jsonp、window.name、window.postMessage