学

期

项

目

二

-Legou

商

城

---熊雪媛

# 0.项目开发的过程理解

1.项目介绍

本次项目是一个商城类型的项目，用户分为商家，卖家，商家可以在这个平台发布产品出售产品，买家可以搜索，买商品、支付、评论，项目保护了：登录注册模块，首页展示，模块，商品展示模块，购物车模块，产品模块详情模块，会员中心模块……

2.项目周期：3~6个月

3.人员配置：

项目经理：1

产品经理：1

UI设计：1~2

前端：2

后端：4

测试：1人（有些公司没有， 有些公司测试是独立的）

4.项目完成流程：

需求收集=>需求评审=>项目立项=>项目排期（甘特图）=>产品原型图=>（前端后端）=>项目测试=>项目部署上线=>后期维护

5.项目计算栈：html/h5 css/css3 javaScript jQuery

6.项目职责：

1.负责了项目工程目录搭建

2.高度还原UI设计图

3.首页模块以及所有前端交互

4.购物车模块以及前端交互

5.产品详情模块以及前端交互。

# 知识点的总结

图示

描述已自动生成

# 面试的18个问题

1. js数据类型有哪些？（基本类型和引用类型）

（原始数据）string/number/boolean/undefined/symbol

（引用数据类型）object(array和函数属于object)

数据类型一共7种（6个基本＋1个引用）

1. undefind null boolean string number
2. Object (object array function)

面试题：如何检测一个变量是否是数组:

1.Array.isArray(变量)//如果是true就是数组否则不是

2.变量instanceof Array//如果是true就是数组否则不是

1. 如何转换类型:

parseFloat();//转为浮点数

parseInt();//转为整数

+ 字符串 //变为字符串

- 字符串//变为数字

转数组：parseFloat();

转字符串toString()/string()

数组转字符串join();

var a,b,c;

a = new Array(a,b,c,d,e);

b = a.join(‘-’)//a-b-c-d-e 使用拼接数组元素

c = a.join(‘’); //abcde

字符串转数组：split();

var str =’ab+c+de’;

var a = str.split(‘+’);//[ab,c,de]

var b = str.split(‘’);//[a,b,+,c,+,d,e]

3数组从小到大排序

方法一：sort方法

第二种方法

利用循环的方法进行

4、数组从大到小排序可以先使数组从大到小排序，然后在添加reverse()方法，使数组顺序颠倒。

5、显示转换与隐式转换

JS中有5种简单的数据类型（也称之为基本数据类型）：undefined\Null\Boolean\NUll\String,还有一种复杂的数据——Object，Object本质是一组无序的明值对组成的。

对一个值使用typeof操作符可以返回该值的数据类型，typeof操作符会返回一些令人迷惑但技术上却正确的值，比如调用typeof null会返回“object”，应为特殊值null被认为是一个空的对象调用。

显式转换：主要通过JS定义的数据转换方法

各种数据类型及其应对的转换规则：

|  |  |  |
| --- | --- | --- |
| 数据类型 | 转化为true的值 | 转换为false的值 |
| Boolean | True | False |
| String | 任何非空字符串 | 空字符串 |
| Number | 任何非零数字值（包含无穷大） | 0和Nan |
| Object | 任何对象 | Null |
| Undefined | N/a | undefined |

隐式转换：是系统默认的，不需要加以声明就可以的转换，一般情况下，数据的类型转换通常是由编译系统自动进行的，不需要人工干预

大致规则如下：

1. 对象和布尔值比较：对象和布尔值比较是，对象先转换为字符串，然后再转换为数字，布尔值直接转换为数字
2. 对象和字符串比较：对象和字符串比较时，对象转换为字符串，然后两者进行比较
3. 对象和数字比较：对象和数字比较时，字符串转换为数字，二者再比较
4. 字符串和数字比较：字符串和数字比较时，字符串转换为数字，二者再比较，true=1，false=0
5. 字符串和布尔值比较：字符串和布尔值比较时，二者全部转换为数值再比较
6. 布尔值和数字比较：布尔值和数字比较时，布尔转换为数字，二者比较

6、mouseover/mouseout与mousenter/mouseleave的区别与联系

1.mouseover/mouseout是标准事件，所有的浏览器都支持；mousenter/mouseleave是IE5.5引入的特有事件，后来被DOM3标准采纳现代浏览器页支持

mouseover/mouseout是冒泡事件；mousenter/mouseleave不冒泡，需要为多个元素监听鼠标移入/移除事件时，推荐使用mouseover、mouseout托管，提高性能

1. forEach和map和filter的区别

这两个API都可以遍历数组

forEach函数，是给数组种的每个都执行一遍回调函数不会返回一个值

Map方法时通过调用数组中的每个元素，映射到新元素中，从而创建一个新数组

filter()方法创建一个新的数组，新数组的元素是通过检查指定数组中符合条件的所有元素。

注意：filter()不会对空数组进行检测，也不会改变原始数组。

如果是复合类型时，如果只改变符合类型的其中某个value时，将可以正常使用

1. offsetWidth/offsetHeight\clientWidth/clientHeight与scorllWidth/scrolHeight的区别？

|  |  |
| --- | --- |
| offsetWidth | 返回元素的宽度(包括元素宽度、内边距和边距，不包括外边距) |
| offsetHeight | 返回元素的高度(包括元素高度、内边距和边距，不包括外边距) |
| clientWidth | 返回元素的宽度(包括元素宽度、内边距，不包括外边距和边距) |
| clientHeight | 返回元素的高度(包括元素高度、内边距，不包括外边距和边距) |
| scorllWidth | 返回元素的宽度(包括元素宽度和溢出尺寸、内边距，不包括外边距和边距) |
| scrolHeight | 返回元素的高度(包括元素高度和溢出尺寸、内边距，不包括外边距和边距) |

9、==与===的区别？

===为等同符 ,当左边的值与类型都完全相等时，会返回true；

==为等值符，用来判断值是否相同，不会判断类型是否相同

一个是赋值

一个是等于

10、null与undefined的去呗？

Null：用来表示尚未存在的对象，常用来表示函数企图返回一个不存在的对象

undefied主要指定了变量，但是并未赋值

ECMAscript认为undefined是从null派生出来的，他们的值是一样的但是类型却不一样。

null==undefined //true

null===undefined //false

1. 字符串操作方法（8个）
2. charCodeAt方法返回一个整数，代表指定位置字符的Unicode编码。
3. formCharCode方法从一些Unicode字符串中返回一个字符串。
4. charAt方法返回指定索引位置处的字符。如果超出范围值返回空字符串。
5. Slice方法返回字符串的片段。
6. Substring方法返回位于string对象中指定位置的子字符串。
7. Substr方法返回一个从指定位置开始的指定长度的字符串。
8. IndexOf方法返回string对象内第一次出现子字符串位置，如果没有找到子字符串，则返回-1.
9. lastIndexOf方法返回string对象中字符串最后出现的位置，如果没有匹配到子字符串，则返回-1.
10. Search帆帆返回与正则表达式查找内容匹配的第一个字符串的位置。
11. Concat方法返回字符串值，该值包含了两个或多个提供的字符串的连接。
12. 数组操作的方法（10个）
13. concat()
14. var arr1=[1,2,3];var arr2=[4,5];var arr3=arr1.concar(arr2);
15. Console.log(arr1);//[1,2,3]
16. Console.log(arr3);//[1,2,3,4,5]

Concat()方法用于连接两个或多个数组，该方法不会改变现有的数组，仅会返回被连接数组的一个副本。

1. join()

Join()方法用于把数组中的所有元素放入一个字符串。元素时通过指定的分隔符进行分隔的，默认使用’,’号分隔，不改变原数组。

var arr=[2,3,4];

Console.log(arr.join());//2,3,4

Console.log(arr);//[2,3,4]

1. Push()

Push()方法可向数组的末尾添加一个或多个元素，并返回新的长度，末尾添加返回的长度，会改变原数组。

var a=[2,3,4];var b=a.push(5);

Console.log(a);//[2,3,4,5]

Console.log(b);//4

push方法可以一次添加多个元素push(data1,data2,...)

1. Pop()

Pop()方法用于删除并返回数组的最后一个元素，会改变原数组。

var arr=[2,3,4];

Console.log(arr.pop());//4

Console.log(arr);//[2,3]

1. Shift()

Shift()方法用于把这个数组的第一个元素从其中删除，并返回第一个元素的值，返回第一个元素，改变原数组。

var arr=[2,3,4];

Console.log(arr.shift());//2

Console.log(arr);//[3,4]

1. Unshift()

unshift()方法可向数组的开头添加一个或多个元素，并返回新的元素，改变原数组。

var arr=[2,3,4];

Console.log(arr.unshift());//6

Console.log(arr);//[3,6,2,3,4,5]

Tip:该方法可以不传参数，不传入参数就是不增加元素。

1. slice()

返回一个新的数组，包含从start到end（不包括元素）的arrayObject中的元素。返回选定的元素，该方法不会修改原数组。

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

Console.log(arr.slice(1,3));//[3,4]

Console.log(arr);//[2,3,4,5]

1. splice()

Splice()方法可以删除从index处开始的零个或多个元素，并且用参数列表中声明的一个或多个值来替换那些被删除的元素。如果从arrayObject中删除了元素，则返回的是含有被删除元素的数组。Splice()方法会直接对数组进行修改。

1. Substring()和substr()

相同点：如果只是写一个参数，两者作用都一样：都是截取字符串从当前下标以后直到字符串片段。

不同点：第一个参数是截取字符串的长度。

1. Sort()排序
2. js中定时器有哪些？他们的却别及用法是什么？

setTimeout 只执行一次

setInterval 会一直重复执行

1. document.write和innerHTMl的却别？

document.write是直接写入到页面的内容流，如果在写之前没有调用document.open，浏览器会自动调用open。每次写完后重新调用该函数，会导致页面被重写。

innerHTML则是DOM页面元素的一个属性，代表该元素的HTML内容，你可以精确到某一个具体的元素来修改，如果想修改document的内容，则需要修改document.documentElement.innerElement。

innerHTML将内容写入某个DOM节点，不会导致页面全部重写，innerHTML很多情况下都优于docoment.write,其原因在于允许更精准的控制要刷新页面的那个部分。

15解释一下DOM和BOM？

BOM

1. BOM是Browser Object Model的缩写，即浏览器对象模型。
2. BOM没有相关标准。
3. BOM的最根本是window

DOM

1. DOM是Document Object Model的缩写，即文档对象模型
2. DOM是W3C的标准
3. DOM最根本对象是document（实际上是window.document）。

16、jQuery一个对象可以同时绑定多个事件，这是如何实现的？

jQuery每调用一个方法，或绑定事件，都返回this（当前的dom），所以可以链式操作，绑定多个事件

17、针对jQuery的优化方法？

缓存频繁操作DOM对象

尽量使用ID选择器代替类选择器

总是从#DI选择器来继承

尽量使用链式操作

在绑定事件上使用时间委托

采用jQuery的内部函数数据（）来存储数据

使用最新版本的jQuery

18、jQuery常用选择器

一、基本选择器

基本选择器是jQuery中最常用也是最简单的选择器，它通过元素的id\class和标签名等来查找DOM元素。

1. ID选择器#id
2. 类选择器class
3. 元素选择器element
4. \*
5. selector1,selector2,...,selectorN
6. 层次选择器

通过DOM元素之间的结构（层次关系）来获取特定元素，例如后代元素、子元素、相邻元素、兄弟元素等。

层次选择器

$("ancestor descendant") 选取ancenstor元素中所有descendant（后代）元素，不仅仅是直接子元素 集合元素 $("div span")选取<div>里的所有<span>元素

$("parent>child") 选取parent元素下的child（子）元素，直接子元素，例如：不包括子元素中的子元素 集合元素 $("div>span")选取<div>元素下名为<span>的元素

$("prev+next") 选取紧接在prev元素后的next元素 集合元素 $(".one+div")选取class为one的下一个<div>元素

$("prev~siblings") 选取prev元素之后所有siblings元素 集合元素 $("#two~div")选取id为two的元素后面的所有<div>同辈

1. 过滤选择器
2. 基本过滤器：过滤器主要通过特定的过滤规则来筛选所需的 DOM 元素，和 CSS 中的伪类的语法类似：使用冒号(:)开头。

表格

描述已自动生成

1. 内容过滤器：过滤器主要通过特定的过滤规则来筛选所需的 DOM 元素，和 CSS 中的伪类的语法类似：使用冒号(:)开头。

表格

描述已自动生成

1. 可见性过滤：可见性过滤器根据元素的可见性和不可见性来选择相应的元素。

图形用户界面, 文本, 应用程序

描述已自动生成

1. 子元素过滤：子元素过滤器的过滤规则是通过父元素和子元素的关系来获取相应的元素。

表格

描述已自动生成

四、表单选择器

1. :button [Selector](https://so.csdn.net/so/search?q=Selector)

描述：选择所有按钮元素和类型为按钮的元素

语法：$(':button')或者jQuery(":button")

2. :button Selector

描述：选择所有按钮元素和类型为按钮的元素

语法：$(':button')或者jQuery(":button")

3. :checked Selector

描述：选择所有复选框勾选的元素

语法：$(":checked")或者jQuery(":checked").

4:disabled Selector

描述：选择所有被禁用的元素

语法：$(":disabled")或者jQuery(":disabled")

5. :disabled Selector

描述：选择所有被禁用的元素

语法：$(":disabled")或者jQuery(":disabled")

6. :file Selector

描述：选择所有类型为文件的元素（file）

语法：$(":file")或者jQuery(":file")

7. :focus Selector

描述：选择当前获取焦点的元素。

语法：$(":focus")或者jQuery(":focus")

8. :focus Selector

描述：选择当前获取焦点的元素。

语法：$(":focus")或者jQuery(":focus")

9. :input Selector

描述：选择所有的input、textarea、selector和button元素

语法：$(":input")或者jQuery(":input")

10. :input Selector

描述：选择所有的input、textarea、selector和button元素

语法：$(":input")或者jQuery(":input")

11. :radio Selector

描述：选择所有类型为单选框的元素

语法：$(":radio")或者jQuery(":radio")

12. :submit Selector

描述：选择所有类型为提交的元素

语法：$(":submit")或者jQuery(":submit")

# 商城项目开发遇到的问题及其解决方案