1. 介绍js有哪些内置对象？

Object 是 JavaScript 中所有对象的父对象

数据封装类对象：Object、Array、Boolean、Number 和 String

其他对象：Function、Arguments、Math、Date、RegExp、Error

1. 说几条写JavaScript的基本规范？

1.不要在同一行声明多个变量。

2.请使用 ===/!==来比较true/false或者数值

4.不要使用全局函数。

6.函数不应该有时候有返回值，有时候没有返回值。

9.for-in循环中的变量 应该使用var关键字明确限定作用域，从而避免作用域污染。

2.例举3种强制类型转换和2种隐式类型转换?

* 强制（parseInt,parseFloat,number）隐式（== – ===）；

3.数组方法pop() push() unshift() shift()

* Push()尾部添加 pop()尾部删除
* Unshift()头部添加 shift()头部删除

4.ajax请求的时候get 和post方式的区别?

* 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;

5.call和apply的区别

* Object.call(this,obj1,obj2,obj3)
* Object.apply(this,arguments)

6.ajax请求时，如何解释json数据

* 使用eval parse,鉴于安全性考虑 使用parse更靠谱;

7.事件委托是什么

* 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！

9.添加 删除 替换 插入到某个接点的方法

obj.appendChidl()  
obj.innersetBefore  
obj.replaceChild  
obj.removeChild  
10.说一下什么是javascript的同源策略？

* 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

11.编写一个b继承a的方法;

JavaScript

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | **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)};  } |

12.如何阻止事件冒泡和默认事件

JavaScript

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | **function** stopBubble(e)  {  **if** (e && e.stopPropagation)          e.stopPropagation()  **else**          window.event.cancelBubble=**true**  }  **return** **false** |

13.下面程序执行后弹出什么样的结果?

JavaScript

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19 | **function** fn() {  **this**.a = 0;  **this**.b = **function**() {  **alert**(**this**.a)      }  }  fn.prototype = {      b: **function**() {  **this**.a = 20;  **alert**(**this**.a);      },      c: **function**() {  **this**.a = 30;  **alert**(**this**.a);      }  }  **var** myfn = **new** fn();  myfn.b();  myfn.c(); |

15.下面程序的结果

JavaScript

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | **function** fun(n,o) {    console.log(o)  **return** {      fun:**function**(m){  **return** fun(m,n);      }    };  }  **var** a = fun(0);  a.fun(1);  a.fun(2);  a.fun(3);  **var** b = fun(0).fun(1).fun(2).fun(3);  **var** c = fun(0).fun(1);  c.fun(2);  c.fun(3); |

//答案：

//a: undefined,0,0,0  
//b: undefined,0,1,2  
//c: undefined,0,1,1

16.下面程序的输出结果

JavaScript

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | **var** **name** = 'World!';  (**function** () {  **if** (**typeof** **name** === 'undefined') {  **var** **name** = 'Jack';          console.log('Goodbye ' + **name**);      } **else** {          console.log('Hello ' + **name**);      }  })(); |

17.了解Node么？Node的使用场景都有哪些？

* 高并发、聊天、实时消息推送

18.介绍下你最常用的一款框架

* jquery,rn,angular等;

19.对于前端自动化构建工具有了解吗？简单介绍一下

* Gulp,Grunt等；

20.new操作符具体干了什么呢?

1、创建一个空对象，并且 this 变量引用该对象，同时还继承了该函数的原型。

2、属性和方法被加入到 this 引用的对象中。

3、新创建的对象由 this 所引用，并且最后隐式的返回 this 。

var obj = {};

obj.\_\_proto\_\_ = Base.prototype;

Base.call(obj);