

下列事件哪个不是由鼠标触发的事件 ()

正确答案: D 你的答案: 空 (错误)

click

contextmenu

mouseout

keydown

下面关于 CSS 布局的描述, 不正确的是?

正确答案: D 你的答案: 空 (错误)

块级元素实际占用的宽度与它的 width 属性有关;

块级元素实际占用的宽度与它的 border 属性有关;

块级元素实际占用的宽度与它的 padding 属性有关;

块级元素实际占用的宽度与它的 background 属性有关。

下面有关 html 的描述, 不推荐的是?

正确答案: B 你的答案: 空 (错误)

在页面顶部添加 doctype 声明;

在 </head> ... <body> 中间插入 HTML 代码;

避免使用 标签;

使用 <table> 元素展现学生成绩表等数据。

浏览器在一次 HTTP 请求中, 需要传输一个 4097 字节的文本数据给服务端, 可以采用那些方式?

正确答案: E 你的答案: 空 (错误)

存入 IndexedDB
写入 COOKIE
放在 URL 参数
写入 Session
使用 POST
放在 Local Storage

下面哪个属性不会让 div 脱离文档流 (normal flow) ?

正确答案: C 你的答案: 空 (错误)

position: absolute;
position: fixed;
position: relative;
float: left;

javascript 语言特性中, 有很多方面和我们接触的其他编程语言不太一样, 比如说, javascript 语言实现继承机制的核心就是 , 而不是 Java 语言那样的类式继承。Javascript 解析引擎在读取一个 Object 的属性值时, 会沿着 向上寻找, 如果最终没有找到, 则该属性值为 ; 如果最终找到该属性值, 则返回结果。与这个过程不同的是, 当 javascript 解析引擎执行“给一个 Object 的某个属性赋值”的时候, 如果当前 Object 存在该属性, 则改写该属性的值, 如果当前的 Object 本身并不存在该属性, 则赋值该属性的值。

你的答案 (错误)

参考答案 prototype

参考答案 原型链

参考答案 undefined

填写内容让下面代码支持 a.name = "name1"; b.name = "name2";

```
function obj(name){
```

```
}
```

```
  obj.  = "name2";
```

```
  var a = obj("name1");
```

```
  var b = new obj;
```

你的答案 (错误)

参考答案 if(name){ this.name = name;}return this;

参考答案 prototype.name

输出对象中值大于 2 的 key 的数组

```
var data = {a: 1, b: 2, c: 3, d: 4};
```

```
Object.keys(data).filter(function(x) { return  ;})
```

期待输出: ["c","d"]

你的答案 (错误)

参考答案 data[x]>2

请实现一个 fibonacci 函数，要求其参数和返回值如下所示：

```
1  /**
2   * @desc: fibonacci
3   * @param: count {Number}
4   * @return: result {Number} 第 count 个 fibonacci 值，计数从 0 开始
5   fibonacci 数列为：[1, 1, 2, 3, 5, 8, 13, 21, 34 ...]
6   则 getNthFibonacci(0) 返回值为 1
7   则 getNthFibonacci(4) 返回值为 5
8   */
9   function getNthFibonacci(count) {
10 }
```

你的答案

• 添加笔记

•

•

• 收藏

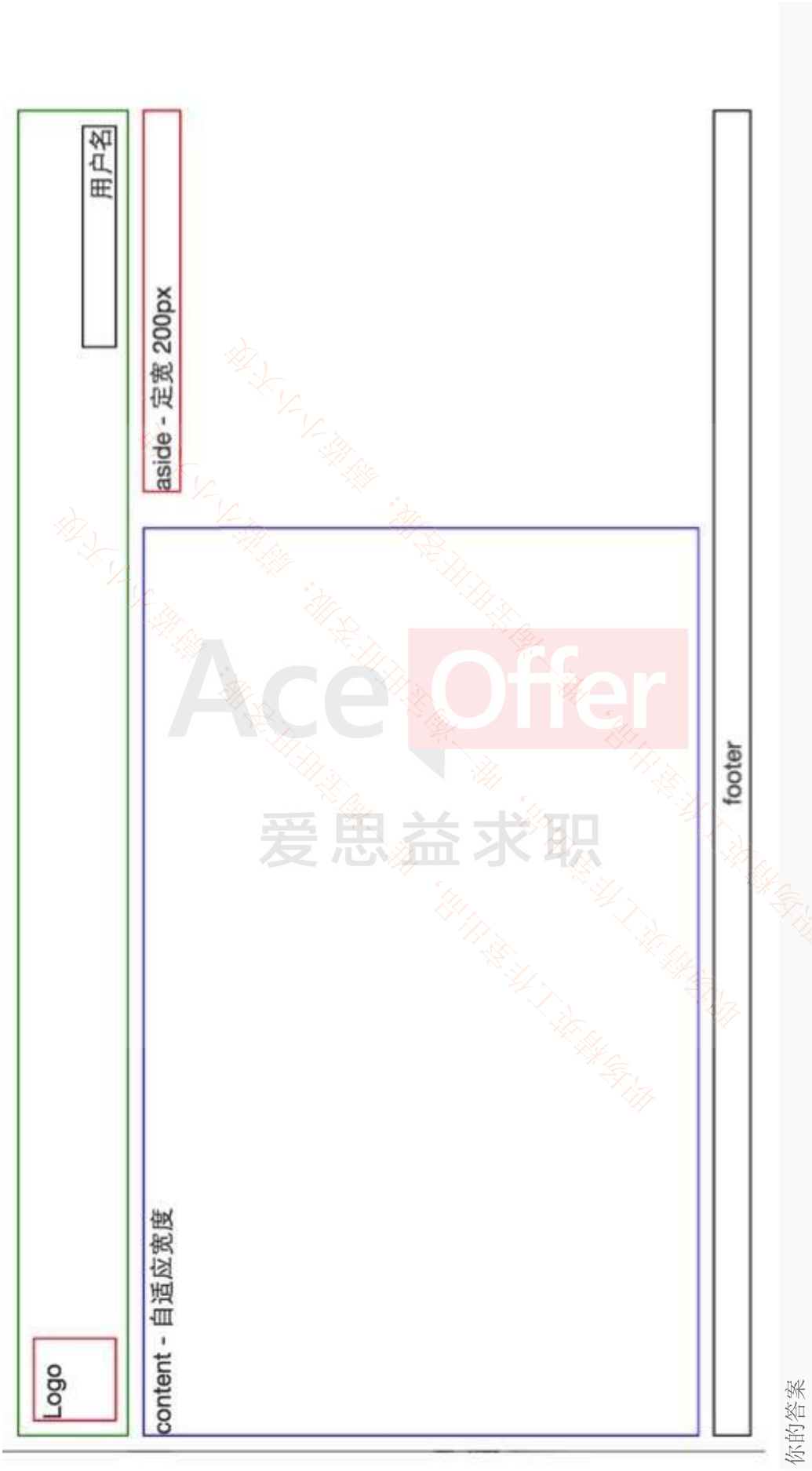
• 纠错

参考答案

此题应该避免使用递归的方法，因为当 count 较大时，递归的方法耗时较长。
故考虑使用迭代法，可以使用数组记录每一项。
但此题只需要用到前面两项，从节约空间的角度讲不需要开辟数组。

```
1 function getNthFibonacci(count) {
2     if(count<0) return 0;
3     if(count<=1) return 1;
4     var first = 1;
5     var second = 1;
6     var third = 0;
7     for(var i = 2; i <= count; i++) {
8         third = first + second;
9         first = second;
10        second = third;
11    }
12    return third;
13 }
```

实现如下页面布局。核心区域左侧自适应，右侧固定宽度 200px



- 添加笔记

-

- 求解答(7)

- 收藏

- 纠错

参考答案

css:

```
1  body {
2      margin: 0;
3  }
4  .fn-clear:after {
5      content: " ";
6      clear: both;
7      display: block;
8      font-size: 0;
9      visibility: hidden;
10     height: 0;
11 }
12 .fn-clear {
13     zoom: 1;
14 }
15 .container {
16     padding: 10px;
17 }
18 .header {
```

```
19 background: #eee;
20 position: relative;
21 margin-bottom: 10px;
22 }
23 .logo {
24     width: 100px;
25     height: 100px;
26     float: left;
27     background: #ff60;
28 }
29 .username {
30     position: absolute;
31     right: 10px;
32     bottom: 10px;
33 }
34 .main {
35     margin-bottom: 10px;
36 }
37 .side-bar {
38     width: 200px;
39     float: right;
40     background: #ccc;
41 }
42 .content {
43     margin-right: 200px;
44     background: #f6f6f6;
```



```
45 }  
46 .footer {  
47     background: #999;  
48 }
```

```
html:  
1 <div class="container">  
2     <div class="header fn-clear">  
3         <div class="logo">logo</div>  
4         <div  
5             class="username">zhoumingXXX@163.com</div>  
6         </div>  
7         <div class="main">  
8             <div  
9                 class="side-bar">menu</div>  
10            <div class="content">左侧内容</div>  
11        </div>  
12        <div class="footer">  
13            footer  
14        </div>  
15 </div>
```