面试

**1.常见的布局**

弹性布局

响应式布局

百分比布局

自适应布局

定位布局position

**2.对齐居中的方式**

设置父元素的display:table-cell,和vertical-align:middle

将父元素设置为display：flex，并且设置align-items:center;justify-content:center

position:absolute;

margin: auto;

width: 20px;

height: 20px;

top:0;

left: 0;

right: 0;

bottom: 0;

­­­­­­\_\_\_

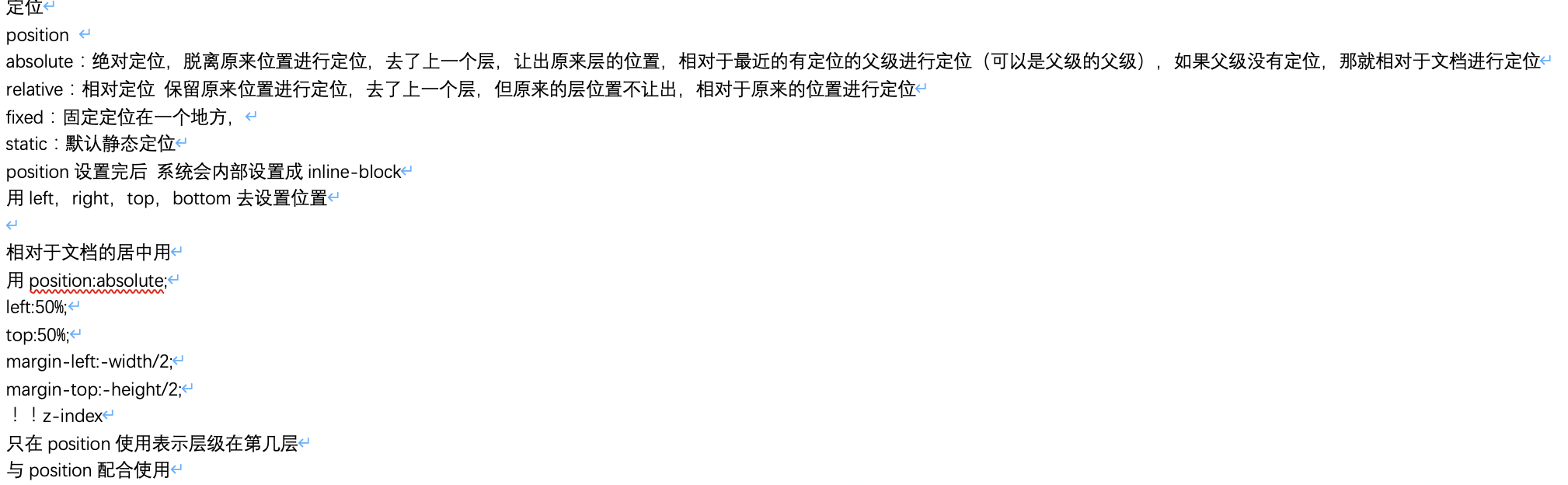
line-height=height

text-align = center

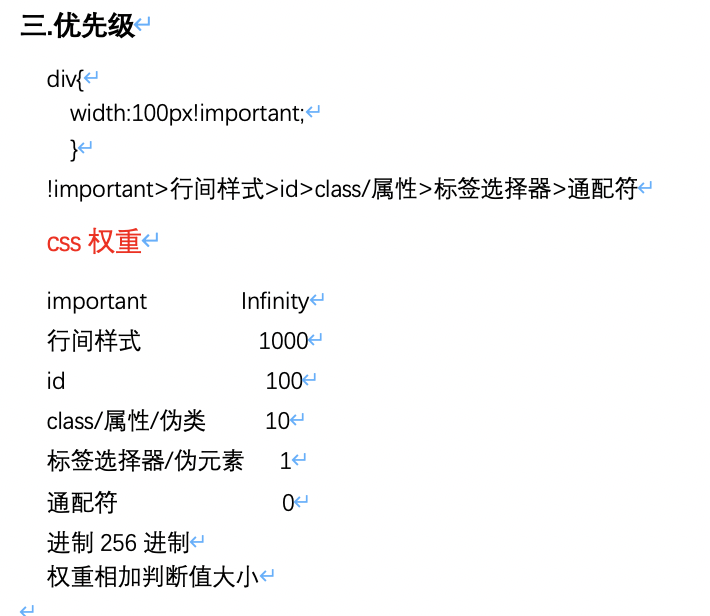
**3.BFC**



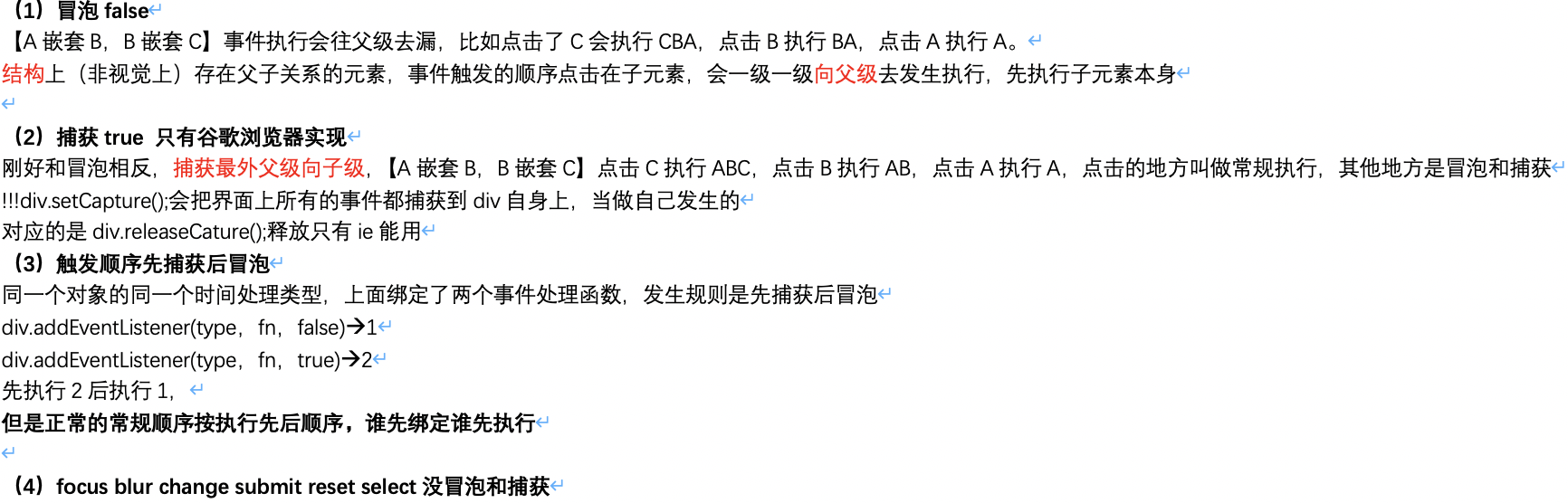
**4.各种定位**



**5.css的优先级**



**6.冒泡和捕获**



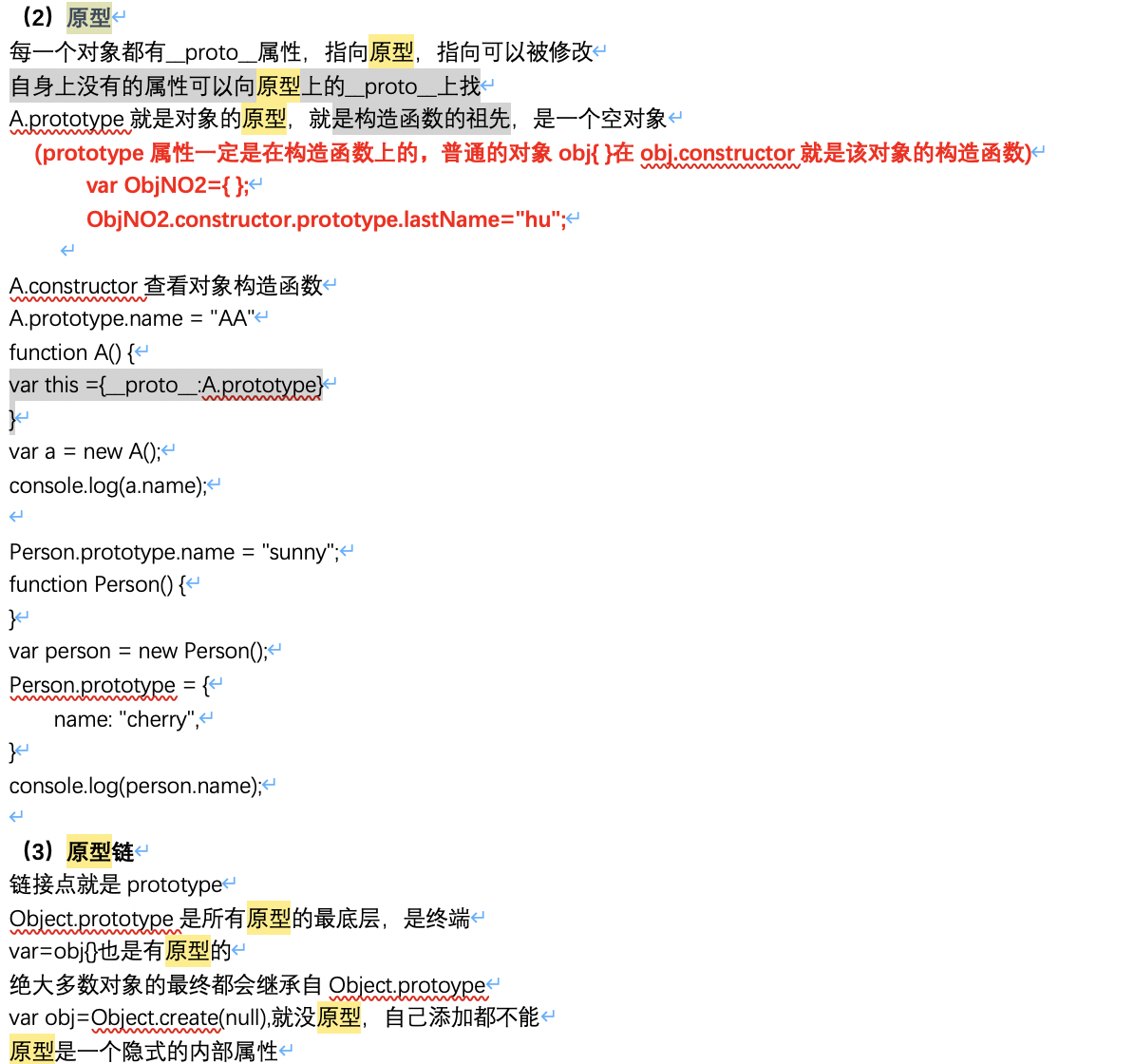
**7.js数据类型和储存方式**



**8.预编译**



**9.原型**



**10.闭包**



**11.**

**建立 TCP 连接**

在HTTP工作开始之前，客户端首先要通过网络与服务器建立连接，该连接是通过 TCP 来完成的，该协议与 IP 协议共同构建 Internet，即著名的 TCP/IP 协议族，因此 Internet 又被称作是 TCP/IP 网络。HTTP 是比 TCP 更高层次的应用层协议，根据规则，只有低层协议建立之后，才能进行高层协议的连接，因此，首先要建立 TCP 连接，一般 TCP 连接的端口号是80；

**客户端向服务器发送请求命令**

一旦建立了TCP连接，客户端就会向服务器发送请求命令；

例如：GET/sample/hello.jsp HTTP/1.1

**客户端发送请求头信息**

客户端发送其请求命令之后，还要以头信息的形式向服务器发送一些别的信息，之后客户端发送了一空白行来通知服务器，它已经结束了该头信息的发送；

**服务器应答**

客户端向服务器发出请求后，服务器会客户端返回响应；

例如： HTTP/1.1 200 OK

响应的第一部分是协议的版本号和响应状态码

**服务器返回响应头信息**

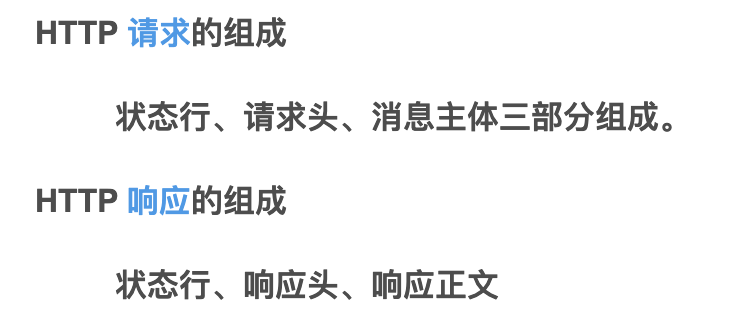
正如客户端会随同请求发送关于自身的信息一样，服务器也会随同响应向用户发送关于它自己的数据及被请求的文档；

**服务器向客户端发送数据**

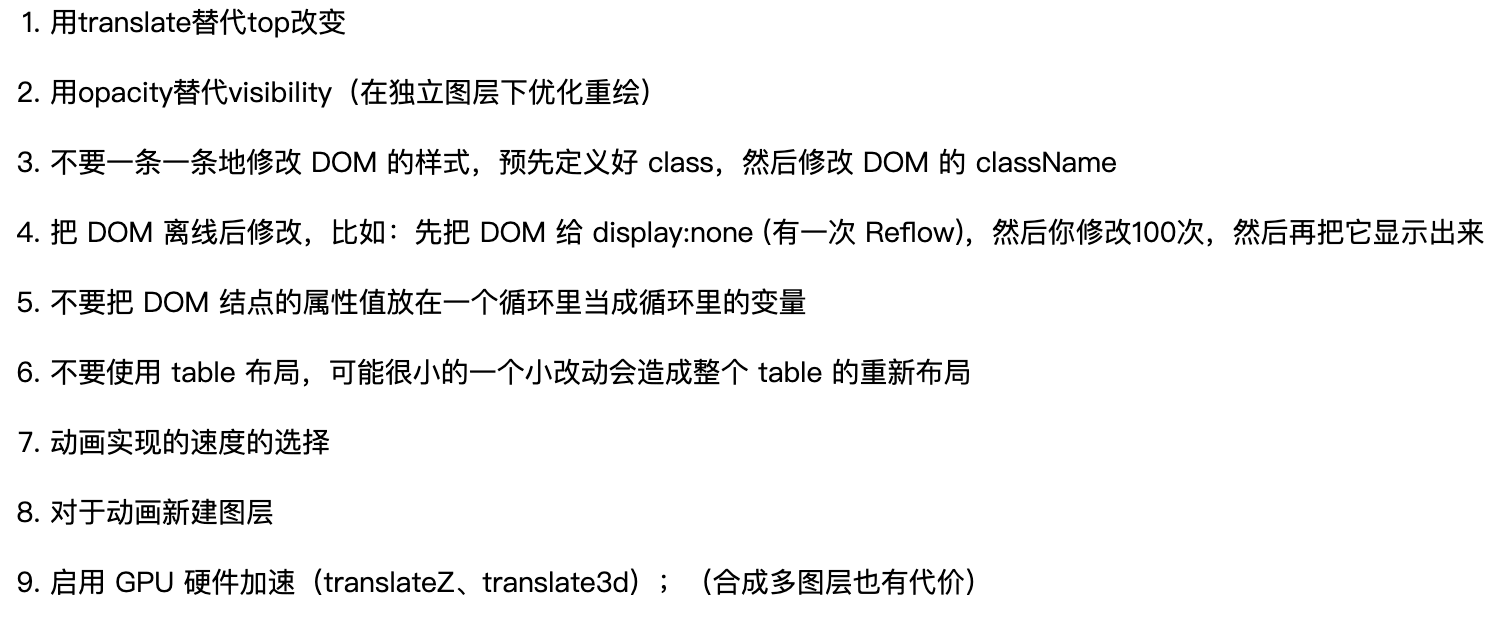
服务器向客户端发送头信息后，它会发送一个空白行来表示头信息的发送到此为结束，接着，它就以 Content-Type 响应头信息所描述的格式发送用户所请求的实际数据；

**服务器关闭 TCP 连接**

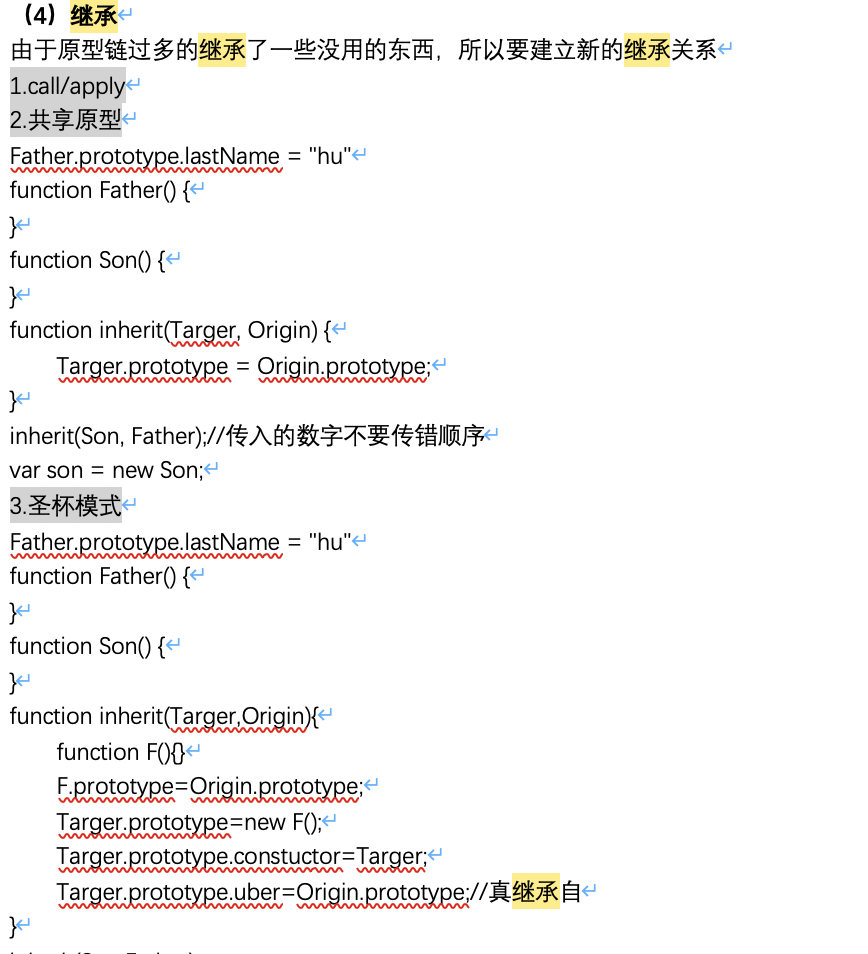
一般情况下，一旦服务器向客户端返回了请求数据，它就要关闭 TCP 连接，然后如果客户端或者服务器在其头信息加入了这行代码 Connection:keep-alive ，TCP 连接在发送后将仍然保持打开状态，于是，客户端可以继续通过相同的连接发送请求。保持连接节省了为每个请求建立新连接所需的时间，还节约了网络带宽。



**12如何避免重排和重绘**



**13.继承方式**



4.es6-extends

**14.react的优化方案**

懒加载

减少ajax的请求

**15.请求头的常见字段**

Accept-Charset 设置接受的字符编码

Accept-Charset: utf-8

Accept-Encoding 设置接受的编码格式

Accept-Encoding: gzip, deflate

Accept-Datetime 设置接受的版本时间

Accept-Datetime: Thu, 31 May 2007 20:35:00 GMT

Accept-Language 设置接受的语言

react的优化，

router-hoc

flex应用场景和原理

this指向