

|* 学习目标

* 能够掌握JavaScript事件

- * 监听机制：事件，事件源，监听器，注册监听

- * 事件：onclick,ondblclick,onload,onmouseover,onmouseout,onmousedown
onmouseup,onmousemove,onkeydown,onkeyup,onkeypress
onchange,onselect, onfocus,onblur,...

* 能够掌握和使用BOM

- * BOM:Browser Object Model

- * Window,Navigator(cookie),Screen,History,Location

- * Window

- * 对话框 (alert , confirm,prompt)

- * open,close

- * setTimeout,setInterval,clear

- * History

- * forward().back(),go(),length

- * Location

- * href,reload

- * <meta charset="UTF-8" http-equiv="refresh" content="5;url="">

* 能够掌握和使用DOM

- * DOM:Document Object Model

- * W3C:三部分

- 核心DOM,XMLDOM,HTMLDOM

- * 核心DOM

- Document,Element,Node,Attribute,Comment,Text

* 回顾

- * js:基本数据类型，运算符，变量，常量，条件语句，循环语句

* 基本数据类型：number , string , boolean , null , undefiend

* java 是强类型语言 , js是弱类型的语言

* type of

* let , var , const

* 其他类型转换boolean

* == ===

* 条件语句：if else , switch--->表单校验

* 循环语句：while , do while,for

* 九九乘法表

* 对象

* 三种定义方式

* let person =new Object();

* let person={firstName:"xiao",lastName:"hei"};

* function Person(firstName,lastName){

 this.firstName=fristName;

 this.lastName=lastName;

 function sayHello(){

 }

}

* let person =new Person("", "");

* 对象

* Array

* let arr=new Array(3);

* let arr=new Array("xiaohei",48,true);

* let arr=["xiaohei",48,true];

* length

* push,join

* for(let index in arr){}

* Boolean , Number , String

* Date : toLocalString();getTime(); get/set

* Global:encodeURIComponent,decodeUri,parseInt,parseFloat,isNaN,eval,....

* Math:PI,round,ceil,floor,random,pow,sqrt,....

* RegExp

* 正则表达式

* let reg=new RegExp("{6,}");

let reg=/ {6,}/;

reg.test(123456);

* 常见符号

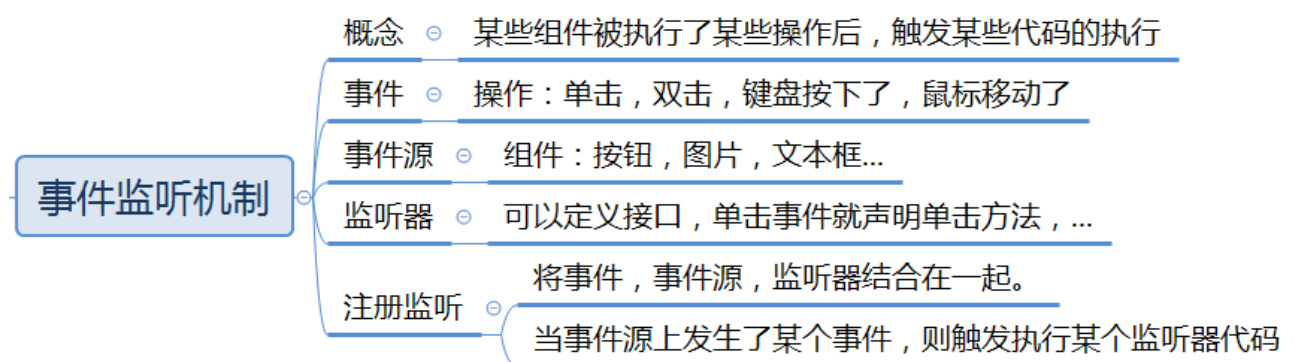
* 单个字符 : [0123],[0-9],[a-z][A-Z][a-zA-Z]\d,\w,\s,.....

* 量词 : * , ? , +

{m,n},{m,},{n}.....

* 能够掌握JavaScript事件

* 事件监听机制



* JS常见的事件

常见的事件

- 点击事件
 - onclick：单击事件
 - ondblclick：双击事件
- 焦点事件
 - onblur：失去焦点
 - onfocus:元素获得焦点
- 加载事件
 - onload：一张页面或一幅图像完成加载。
- 鼠标事件
 - onmousedown 鼠标按钮被按下
 - onmouseup 鼠标按键被松开
 - onmousemove 鼠标被移动
 - onmouseover 鼠标移到某元素之上
 - onmouseout 鼠标从某元素移开
- 键盘事件
 - onkeydown 某个键盘按键被按下
 - onkeyup 某个键盘按键被松开
 - onkeypress 某个键盘按键被按下并松开
- 选择和改变事件
 - onchange 域的内容被改变
 - onselect 文本被选中
- 表单事件
 - onsubmit 确认按钮被点击
 - onreset 重置按钮被点击

```
1 * 事件测试
2 <!DOCTYPE html>
3 <html lang="en">
4 <head>
5     <meta charset="UTF-8">
6     <title>常见事件</title>
7     <script>
8         function load() {
9             alert("页面加载完成");
10        }
11        function imgOnload() {
12            alert("图片加载完成");
13        }
14        function clickMe() {
15            alert("按钮单击事件触发");
16        }
17        function doubleClickMe() {
```

```
18         alert("按钮双击事件触发");
19     }
20     var flag=true;
21     function testFocus() {
22         if(flag){
23             alert("文本框获得焦点");
24             flag=false;
25         }
26     }
27     function testBlur() {
28         alert("文本框失去焦点");
29     }
30     function mouseDown() {
31         console.log("鼠标按钮被按下");
32     }
33     function mouseUp() {
34         console.log("鼠标按键被松开");
35     }
36     function mouseover() {
37         console.log("鼠标移到某元素之上");
38     }
39     function mouseout() {
40         console.log("鼠标从某元素移开");
41     }
42     function mousemove() {
43         console.log("鼠标从某元素移开");
44     }
45     function keydown() {
46         // console.log("某个键盘按键被按下");
47         //let code=event.code;
48         //let codeKey=event.keyCode;
49         //alert(codeKey+": "+code);
50     }
51     function keyup() {
52         // console.log("某个键盘按键被松开");
53     }
54     function keypress() {
55         // console.log("某个键盘按键被按下并松开");
56     }
57     function change() {
```

```
58         //内容改变并且失去焦点的时候
59         let username=document.getElementById("username").value;
60         console.log("内容改变:"+username);
61     }
62     function showMsg() {
63         console.log("文本被选中");
64     }
65     function checkForm() {
66         alert("表单提交");
67         return false;
68     }
69     function mreset() {
70         alert("重置表单");
71         return true;
72     }
73     </script>
74 </head>
75 <body onload="load();" onkeydown="keydown();" onkeyup="keyup();" onkeypress="ke
76     <br>
77     <input type="button" value="单击事件" onclick="clickMe();" /><br>
78     <input type="button" value="双击事件" ondblclick="doubleClickMe();" /><br>
79     <input id="username" type="text" placeholder="请输入用户名" onfocus="testFoc
80     <input type="text" placeholder="请输入用户名" onblur="testBlur();" /><br>
81     <a href="#" onmouseover="mouseover();" onmouseout="mouseout();"
82     onmouseup="mouseUp();" onmousedown="mouseDown();" onmousemove="mousemove
83     <br>
84     <input type="text" value="亮哥教育" onselect="showMsg(event);" /><br>
85
86     <form action="http://www.baidu.com/" onsubmit="return checkForm();" onreset
87         <input type="submit" value="提交" />
88         <input type="text" />
89         <input type="reset" value="重置" />
90     </form>
91 </body>
92 </html>
```



单击事件	
双击事件	
请输入用户名	
请输入用户名	
百度	
亮哥教育	
提交	<input type="text"/>
	重置

* 事件监听机制：事件，事件源，监听器，注册监听

* 事件：onclick, ondblclick, onload, onmouseover, onmouseout, onmousedown, onmouseup
onmousemove, onkeydown, onkeyup, onkeypress, onchange, onselect
onsubmit, onreset

* 能够掌握和使用BOM



* Window 对象



```

1 案例一（测试window对象）
2 <!DOCTYPE html>
3 <html lang="en">
4 <head>
5   <meta charset="UTF-8">
  
```



```
6 <title>window对象</title>
7 <script>
8     /*弹出框相关的方法*/
9     alert("警告");
10    let isDel=confirm("真的要删除嘛?");
11    if(isDel){
12        console.log("删除");
13    }else{
14        console.log("不删除");
15    }
16    let name=prompt("请输入姓名");
17    console.log(name);
18    var newWindow;
19    function mopen() {
20        newWindow= open("http://www.baidu.com/");
21    }
22    function mclose() {
23        newWindow.close();
24    }
25    var t1=setTimeout(mopen,2000);
26    function mclear() {
27        clearTimeout(t1);
28    }
29    var t2=setInterval(mopen,2000);
30    function mclear2() {
31        clearTimeout(t2);
32    }
33 </script>
34 </head>
35 <body>
36     <input type="button" value="打开新窗口" onclick="mopen();"/>
37     <input type="button" value="关闭窗口" onclick="mclose();"/>
38     <br>
39     <input type="button" value="清除定时一" onclick="mclear();"/>
40     <input type="button" value="清除定时二" onclick="mclear2();"/>
41 </body>
42 </html>
43
44 案例二（轮播图）
45 <!DOCTYPE html>
```

```

46 <html lang="en">
47 <head>
48     <meta charset="UTF-8">
49     <title>轮播图</title>
50     <script>
51         var num=1;
52         function lunbo() {
53             num++;
54             if(num>3){
55                 num=1;
56             }
57             let image=document.getElementById("image");
58             image.src="images/banner"+num+".jpg";
59         }
60         function load() {
61             setInterval(lunbo,3000);
62         }
63     </script>
64 </head>
65 <body onload="load();">
66     
67 </body>
68 </html>

```

* Navigator



1 <!DOCTYPE html>

```

2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Navigator</title>
6   <script>
7     document.write("浏览器代号:"+navigator.appCodeName+"<br>");
8     document.write("浏览器名称:"+navigator.appName+"<br>");
9     document.write("浏览器版本:"+navigator.appVersion+"<br>");
10    document.write("是否启动cookie:"+navigator.cookieEnabled+"<br>");
11    document.write("硬件平台:"+navigator.platform+"<br>");
12  </script>
13 </head>
14 <body>
15 </body>
16 </html>

```

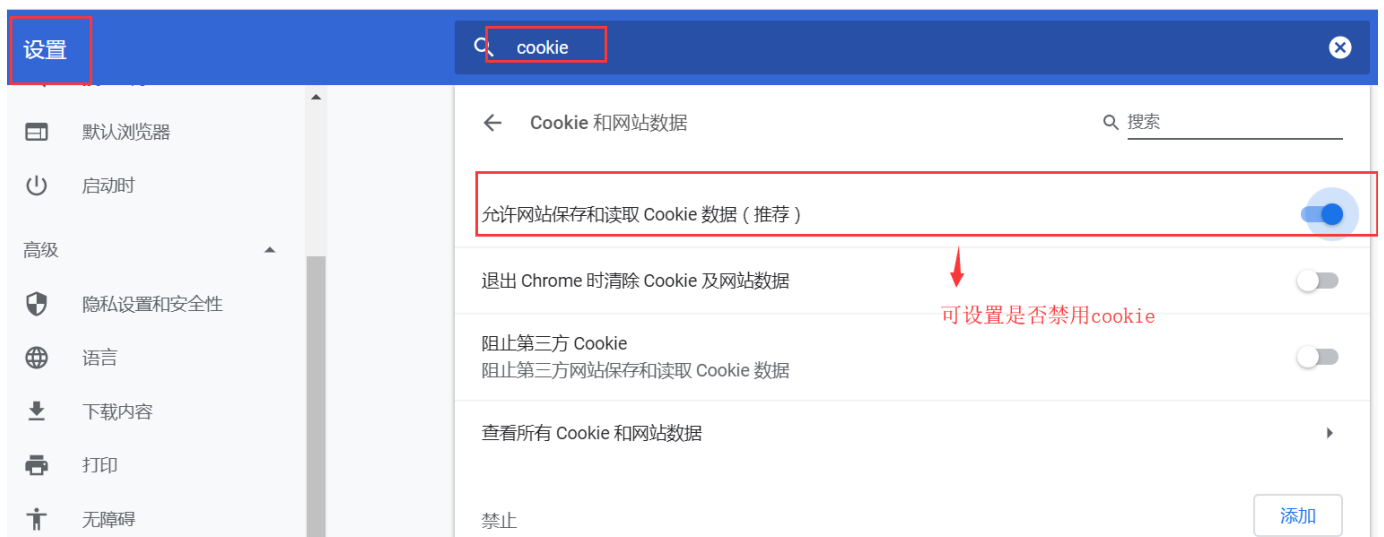
浏览器代号:Mozilla

浏览器名称:Netscape

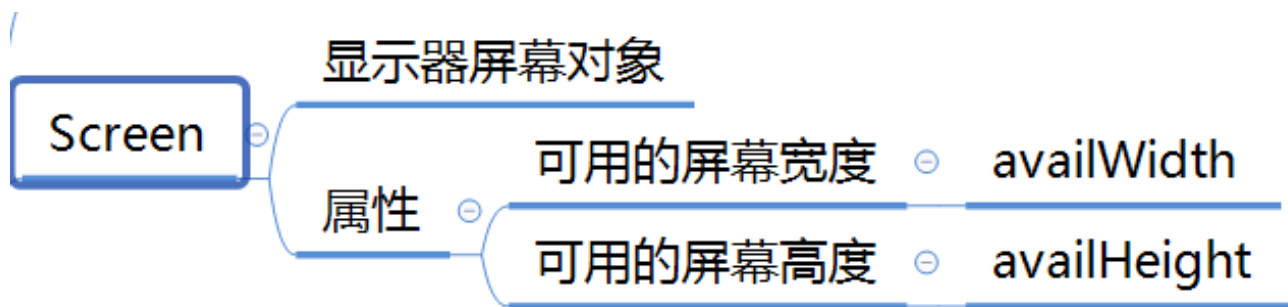
浏览器版本:5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/77.0.3865.120 Safari/537.36

是否启动cookie:true

硬件平台:Win32



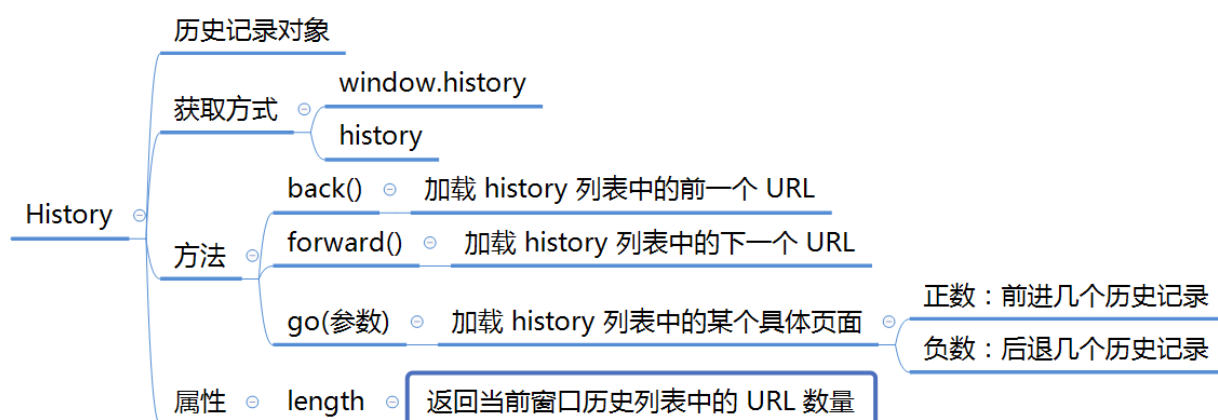
* Screen对象



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Navigator</title>
6     <script>
7         document.write("可用的屏幕宽度:"+screen.availWidth+"<br>");
8         document.write("可用的屏幕高度:"+screen.availHeight+"<br>");
9     </script>
10 </head>
11 <body>
12 </body>
13 </html>
```

可用的屏幕宽度:1280
可用的屏幕高度:680

* History 对象



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>history演示</title>
6     <script>
7         console.log(history.length);
8         function forward() {
9             history.forward();
10            // history.go(1);
11        }
12    </script>
13 </head>
14 <body>
15     <a href="test20.html">test20</a><br>
16     <input type="button" value="前进" onclick="forward();"/><br>
17 </body>
18 </html>
```

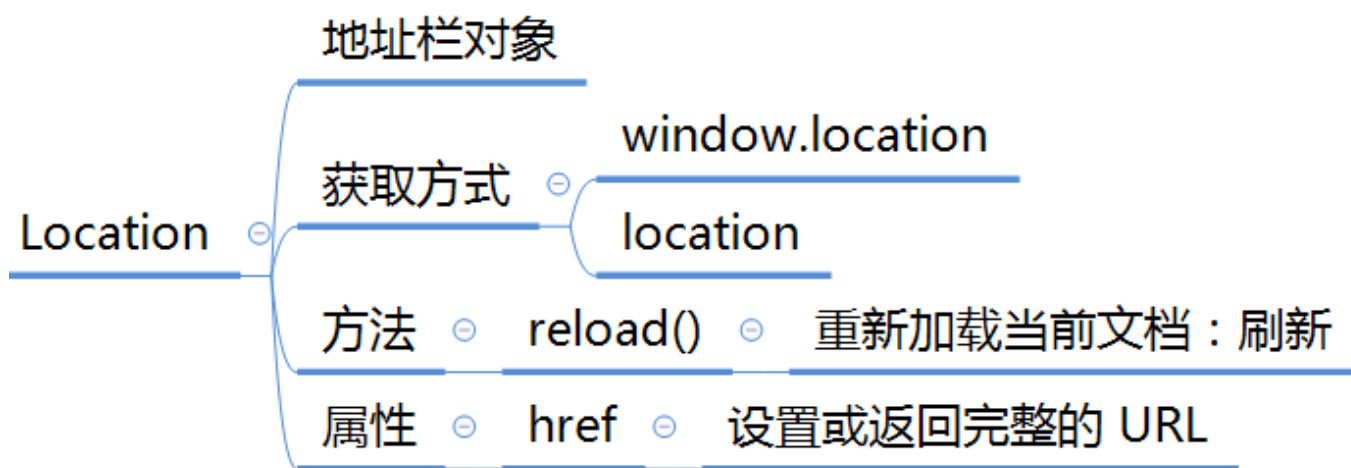
19

20 test20.html:

21

```
22 <!DOCTYPE html>
23 <html lang="en">
24 <head>
25     <meta charset="UTF-8">
26     <title>history2</title>
27     <script>
28         console.log(history.length)
29         function back() {
30             history.back();
31             // history.go(-1);
32         }
33     </script>
34 </head>
35 <body>
36     <input type="button" value="后退" onclick="back();">
37
38 </body>
39 </html>
```

* Location 对象



```
1 案例一：
2 <!DOCTYPE html>
3 <html lang="en">
4 <head>
5     <meta charset="UTF-8">
6     <title>location</title>
7     <script>
8         function clickMe() {
9             location.href="http://www.baidu.com";
10        }
11    </script>
12 </head>
13 <body>
14     <input type="button" value="去百度" onclick="clickMe();"/>
```

```
15 </body>
16 </html>
17
18 案例二：
19 注册
20 <!DOCTYPE html>
21 <html lang="en">
22 <head>
23     <meta charset="UTF-8">
24     <title>表单校验</title>
25     <script>
26         function checkForm() {
27             let username= document.getElementById("username");
28             // 校验用户名
29             if(username.value == ""){
30                 alert("用户名不能为空");
31                 return false;
32             }else if(username.value.length<6){
33                 alert("用户名长度不能小于6");
34                 return false;
35             }
36             // 校验密码
37             let password=document.getElementById("password");
38             if(password.value == ""){
39                 alert("密码不能为空");
40                 return false;
41             }else if(password.value.length<6){
42                 alert("密码长度不能小于6");
43                 return false;
44             }
45             let repassword=document.getElementById("repassword");
46             if(repassword.value == ""){
47                 alert("确认密码不能为空");
48                 return false;
49             }else if(repassword.value.length<6){
50                 alert("确认密码不能小于6");
51                 return false;
52             }
53             if(password.value!=repassword.value){
54                 alert("密码和确认密码不一致");
```

```
55         return false;
56     }
57     // 校验成功之后,返回true,表示可以提交表单
58     return true;
59 }
60 </script>
61 </head>
62 <body>
63 <h3>注册新用户</h3>
64 <hr>
65 <form action="test23.html" method="get">
66     <label for="username">账号</label>
67     <input id="username" type="text" placeholder="请输入用户名" name="username"/>
68     <label for="password">密码</label>
69     <input id="password" type="password" name="password"/><br>
70     <label for="repassword">重新输入密码</label>
71     <input id="repassword" type="password" name="password"/><br>
72     <input type="reset" value="重置"/>
73     <input type="submit" value="注册" onclick="return checkForm();"/>
74 </form>
75 </body>
76 </html>
77 第一种方式: 注册成功, 5秒跳转到首页
78 <!DOCTYPE html>
79 <html lang="en">
80 <head>
81     <meta charset="UTF-8">
82     <title>注册成功</title>
83     <style>
84         div{
85             color: red;
86             font-size: 24px;
87             text-align: center;
88         }
89     </style>
90     <script>
91         function load() {
92             var time=document.getElementById("time");
93             var second=5;
94             function showTime(){
```



```

95         second--;
96         if(second<=0){
97             location.href="http://www.baidu.com";
98             second=0;
99             clearInterval(timeId)
100         }
101         time.innerHTML=second+"";
102     }
103     var timeId=setInterval(showTime,1000);
104 }
105 </script>
106
107 </head>
108 <body onload="load();">
109     <div>注册成功,<span id="time">5</span>秒之后, 自动跳转到首页...</div>
110 </body>
111 </html>
112

```

113 第二种方式: 注册成功, 5秒跳转到首页

```

114 <!DOCTYPE html>
115 <html lang="en">
116 <head>
117     <meta charset="UTF-8"
118         http-equiv="refresh"
119         content="5;url=http://www.baidu.com">
120     <title>注册成功</title>
121     <style>
122         div {
123             color: red;
124             font-size: 24px;
125             text-align: center;
126         }
127     </style>
128     <script>
129         function load() {
130             var time = document.getElementById("time");
131             var second = 5;
132             function showTime() {
133                 second--;
134                 time.innerHTML = second + "";

```

```
135         if (second <= 0) {
136             second = 0;
137             clearInterval(timeId)
138         }
139     }
140     var timeId = setInterval(showTime, 1000);
141 }
142 </script>
143 </head>
144 <body onload="load();">
145 <div>注册成功,<span id="time">5</span>秒之后, 自动跳转到首页...</div>
146 </body>
147 </html>
```

148
149 案例三（每2秒刷新页面）

150 第一种写法:

```
151 <!DOCTYPE html>
152 <html lang="en">
153 <head>
154     <meta charset="UTF-8">
155     <title>location</title>
156     <script>
157         function mrefresh(){
158             location.reload();
159         }
160         setInterval(mrefresh,2000);
161     </script>
162 </head>
163 <body>
164
165 </body>
166 </html>
```

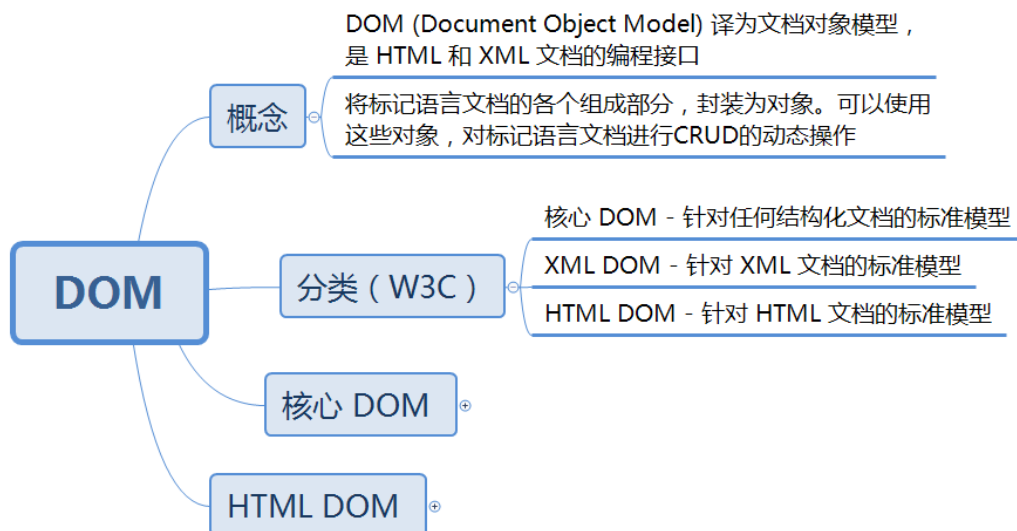
167 第二种写法:

```
168 <!DOCTYPE html>
169 <html lang="en">
170 <head>
171     <meta charset="UTF-8" http-equiv="refresh" content="2">
172     <title>location</title>
173 </head>
174 <body>
```

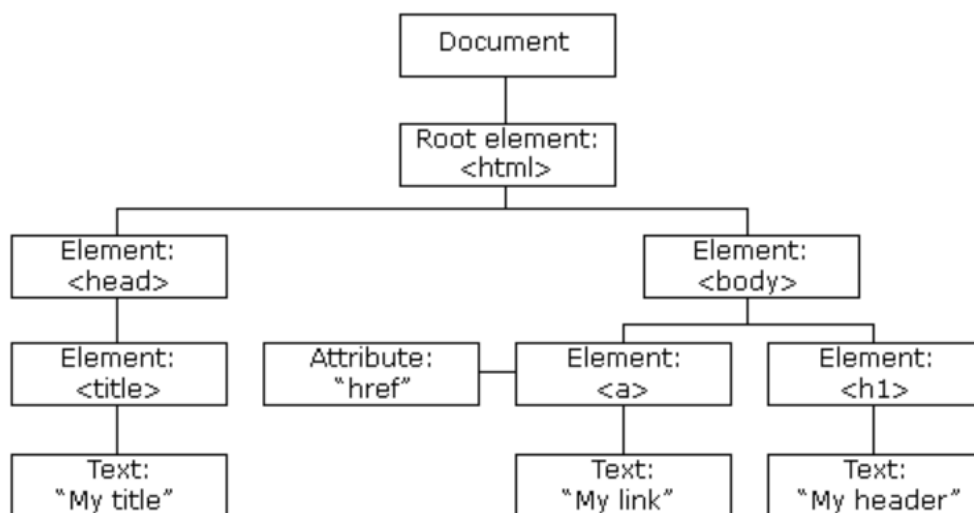
```
175 </body>
176 </html>
```

* 能够掌握和使用DOM

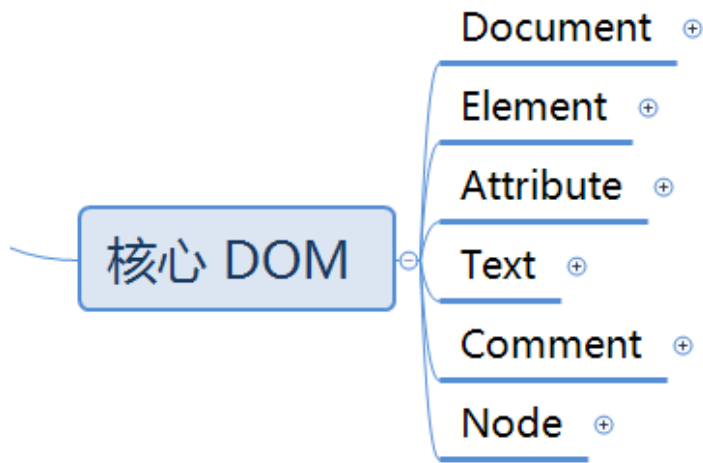
* DOM的概述



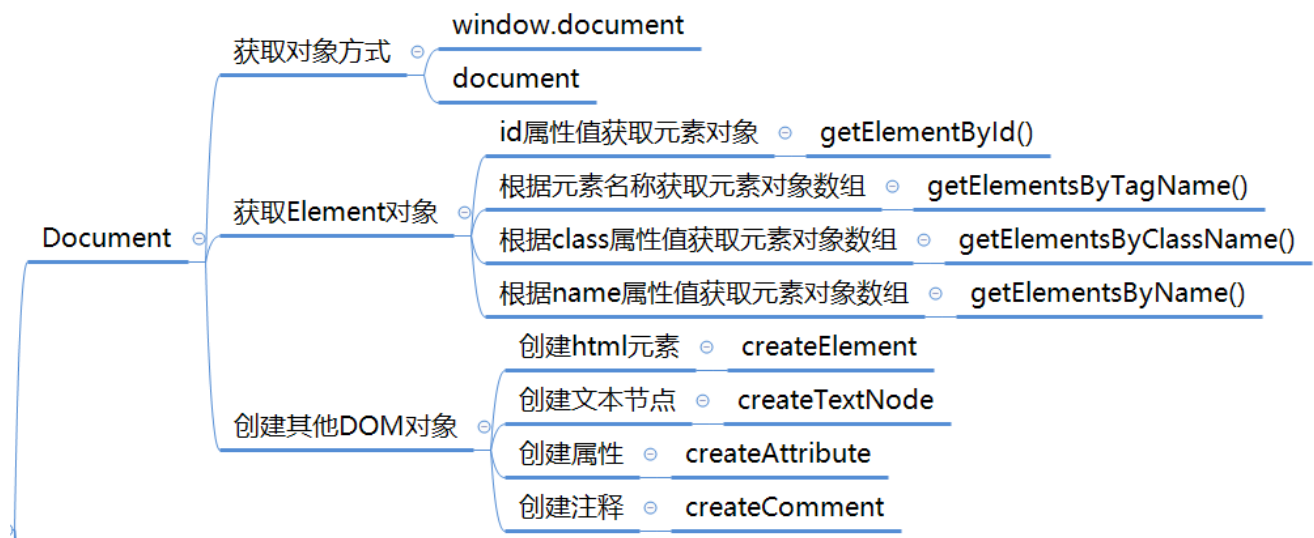
HTML DOM 树形结构:



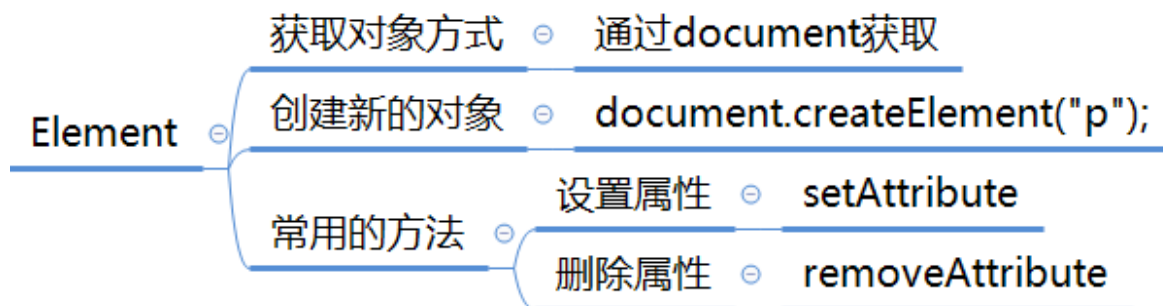
* 核心的DOM



* Document



* Element



* Attribute,Text,Comment

Attribute ○ 属性对象

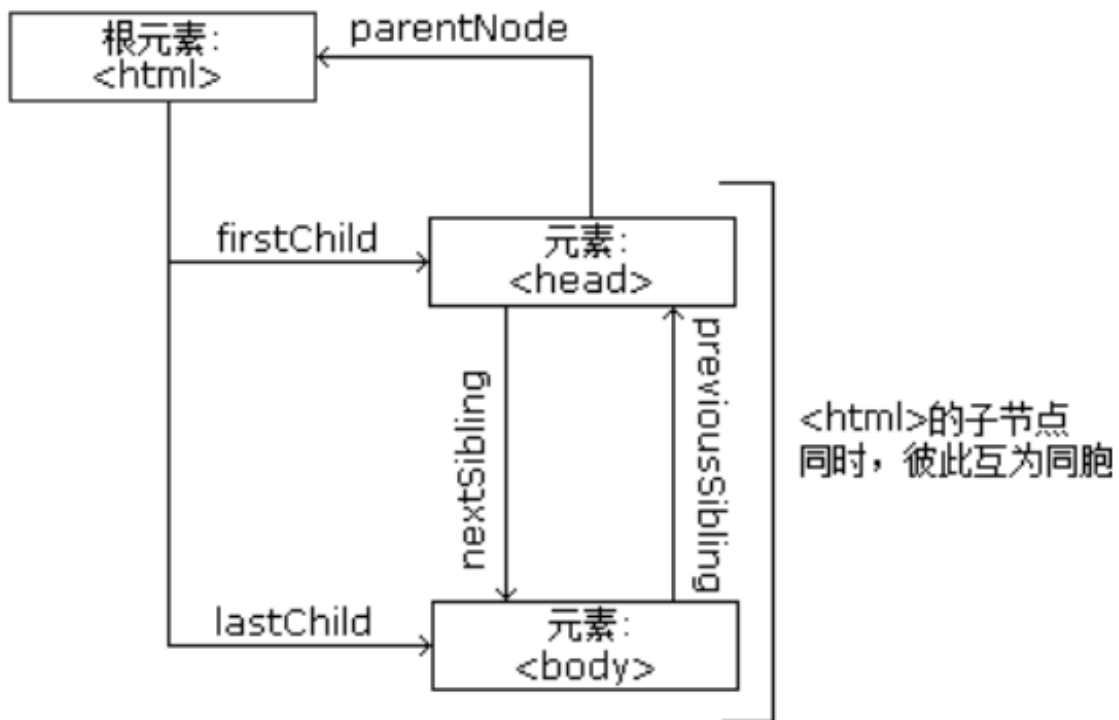
Text ○ 文本对象

Comment ○ 注释对象

* Node



* 节点与节点的关系



* HTML DOM

