



【实验题目】JS 实验 2

【实验目的】学习 Javascript 高级编程。

【调试方法】

方法 1: chrome 调试 js 的方法: [参考](#)

方法 2:

```
try{  
    要调试的代码  
}  
catch(e){  
    alert(e);  
}
```

【实验环境】

- (1) 自己安装的 tomcat
- (2) 自己安装的 Eclipse
- (3) 在线调试: <http://172.18.187.233:8080/testjsp/>
要用的图像在 <http://172.18.187.233:8080/testjsp/temp/images> 中

【实验内容】

1、设计两个数字计时器(timer1.html, timer2.html)。

(1) **简单计时器**: 按启动键开始一直计时。

提示: `setTimeout(timer, 1000);` //ms

参考资料: JavaScript.ppt (下) 定时器

参考截屏:



视频演示见 timer1.mp4:

完成后截屏:



源码(timer1.html):

```
<!DOCTYPE html>
<html lang="zh-cn">
<head>
<title>计时器 1</title>
<style>
    span {text-decoration:underline;}
    span:hover {cursor:pointer}
</style>
<script>
    var t = 0.00;
    var start = false;
    function times() {
        t = t + 0.01;
        document.getElementById("timer").innerHTML = t.toFixed(2);
        window.setTimeout(times, 10);
    }
    function onstart() {
        if(!start) {
            times();
            start=true;
        }
    }
</script>
</head>
<body>
    <h1>计时器 1</h1>
    <div><span id="start" onClick="onstart()">启动</span></div>
    <div id="timer">0.00</div>
</body>
</html>
```

(2) **复杂计时器**: 按启动开始计时,“启动变”为“暂停”,再按继续启动。再暂停时按复位按钮将回到初始状态。

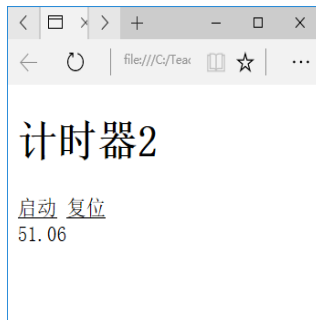
参考截屏:



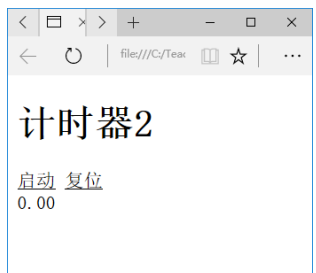
点击启动后:



点击暂停后



点击复位后



视频演示见 timer2.mp4:

完成后截屏:



启动后:



暂停后:



复位后:



源码(timer2.html):

```
<!DOCTYPE html>
<html lang="zh-cn">
<head>
  <title>计时器 2</title>
  <style>
    span {text-decoration:underline;}
    span:hover {cursor:pointer}
  </style>
  <script>
    var t = 0.00;
    var start = false;
    function toggleTimer() {
      if(!start){
        start = true;
        document.getElementById("start").innerHTML = "暂停";
      }else {
        start = false;
        document.getElementById("start").innerHTML = "启动";
      }
    }
    function times() {
      if(start) {
        t = t + 0.01;
        document.getElementById('timer').innerHTML = t.toFixed(2);
      }
      window.setTimeout(times, 10);
    }
    function resetTimer() {
      t = 0.00;
```



```
document.getElementById("timer").innerHTML = "0.00";
start = false;
document.getElementById("start").innerHTML = "启动";
}
</script>
</head>
<body>
<h1>计时器 2</h1>
<div><span id="start" onclick="toggleTimer()">启动</span>
<span onclick="resetTimer()">复位</span></div>
<div id="timer">0.00</div>
</body>
</html>
<script>times();</script>
```

2、(validate.html)验证密码，输入两个密码，如果不相同，则弹出出错信息，否则，提交到 OK.jsp (没有该文件时会出错)。

参考资料：JavaScript.ppt（下）事件对象-通过事件提交表单

提示：onclick="return submit1();" submit1()返回 false 会取消默认行为。

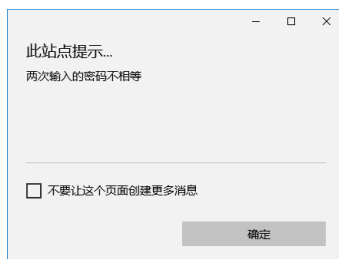
参考截屏：



输入 123 和 1234:



按提交按钮弹出信息，并取消了提交：



如果输入相同，则会提交，提交可能会出错（OK.jsp 不存在）。





完成后截屏：

输入 123 和 1234:

按提交按钮弹出信息，并取消了提交：

如果输入相同，则会提交，提交可能会出错（OK.jsp 不存在）。

此时没有使用 jsp 环境所以正常失败。



源码(validate.html):

```
<!DOCTYPE html>
<html lang="zh-cn">
<head>
  <title>验证正确性</title>
  <script>
    function submit1() {

if(document.getElementById("pass1").value!=document.getElementById("pass2").value) {
    alert("两次输入密码不相等");
    return false;
}
    return true;
}
  </script>
</head>
<body>
  <h1 style="text-align:center">验证正确性</h1>
  <form action="OK.jsp" method="post">
    <p style="text-align:center">    密码: <input type="password"
id="pass1"></p>
    <p style="text-align:center">重输入密码: <input type="password"
id="pass2"></p>
    <p style="text-align:center"><input type="submit" value="提交"
onclick="return submit1()"></p>
  </form>
</body>
</html>
```

3、制作幻灯片显示（参考 html 实验 8 的第 4 题）



功能说明:

- (1) 用 `setTimeout` 制作幻灯片动画。装载网页后自动播放幻灯片。具体见 `slider.mp4`
- (2) 当 `mouseover` 图片时，显示左右箭头，并停止动画。当 `mouseout` 画面时，隐藏左右箭头，重新启动动画。
- (3) 每次点击左箭头将移动一副图像。
- (4) 右下角的三个点每个点对应一幅图像，当前图像的变化时所对应的点也会变化。点击它们时会显示对应的图像。

参考资料: Javascript(下)v2.pdf 事件

参考视频: `slider.mp4`

编程参考:

- (1) 图片采用三个 `img`，并用一个 `div` 包含，样式 `white-space` 要加上 `nowrap`。
- (2) 外层再用一个 `div`，设置宽度为一个图片的宽度，并设置 `overflow` 为 `hidden`。



- (3) 上面的两个 div 均为 relative。
- (4) 每个点和箭头采用 div 和背景图实现，三个点再采用一个 div 封装，该 div 和两个箭头的 div 均采用绝对定位。
- (5) 通过增加 class 或删除 class 实现三个点中活动点的切换。

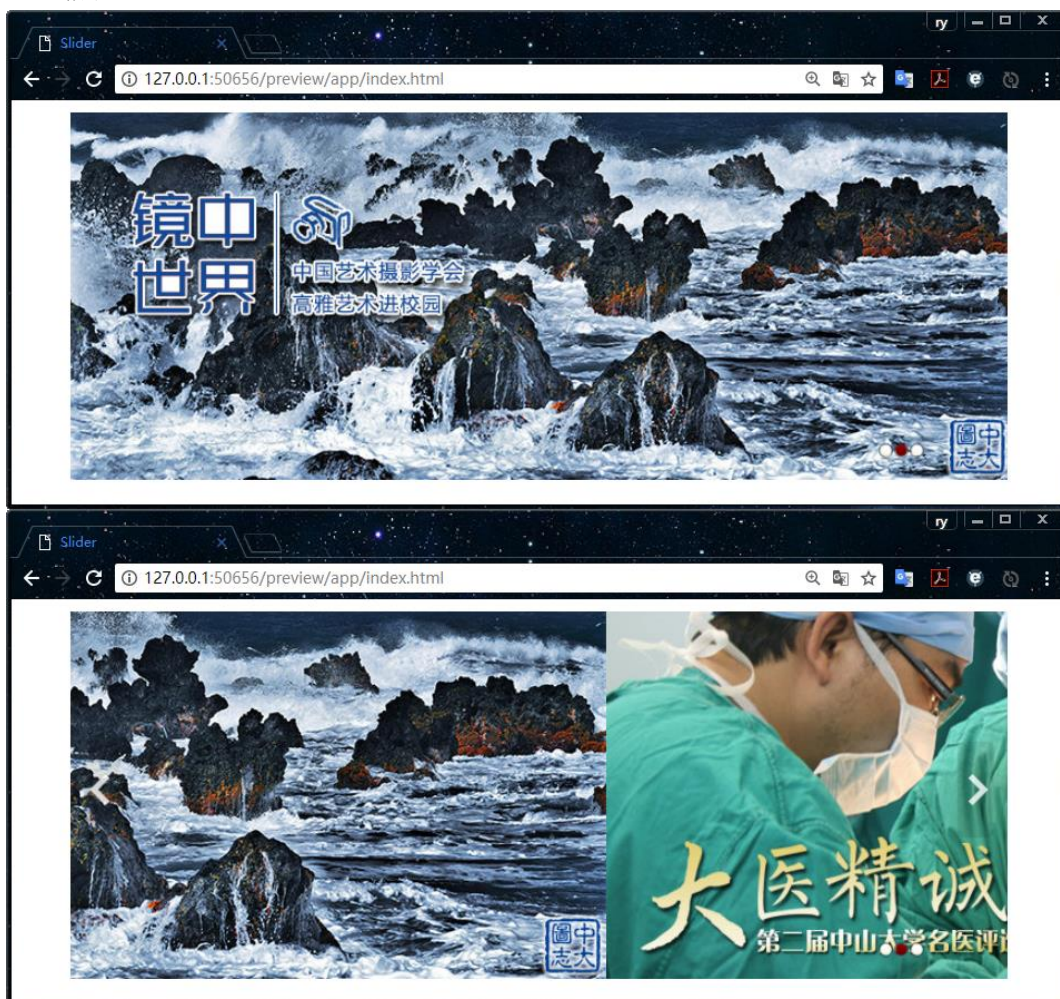
* 所用方法: `querySelector()`, `querySelectorAll()`, `classList.add()`, `classList.remove()`

完成后的屏幕截图:

鼠标移动到上方



播放:



完成后的 html 源码(slider.html 或 slider.jsp):



```
<!DOCTYPE html>
<html>
<head>
<title>Slider</title>
<style>
    #imgs_out{
        height: 300px;
        width: 766px;
        overflow: hidden;
        position: relative;
        margin: 10px auto;
    }
    #imgs{position: relative;white-space: nowrap;
    }
    .img{margin: 0 -2px;}
    #points{
        position: absolute;
        top: 265px;
        right: 70px;
        white-space: nowrap;
    }
    .point{
        display: inline-block;
        background: url(images/navigator.png) no-repeat;
        width: 12px;
        height: 12px;
        margin: 0px -2px;
    }
    .point1{background-position: 0px -24px;}
    .arrow{
        position: absolute;
        background: url(images/arrows-30.png);
        width: 32px;
        height: 80px;
        top:105px;
        display: none;
    }
    #arrow1{left: 10px;background-position: 0px 0px;}
    #arrow2{right: 10px;background-position: 0px 80px;}
</style>

</head>
<body>
    <div id="imgs_out" onMouseMove="movearrow(0)" onMouseOut="movearrow(1)">
        <div id="imgs">
            
            
            
            
        </div>

        <div id="points">
            <div class="point point1" onClick="onclickPoint(0)"></div>
            <div class="point" onClick="onclickPoint(1)"></div>
            <div class="point" onClick="onclickPoint(2)"></div>
        </div>

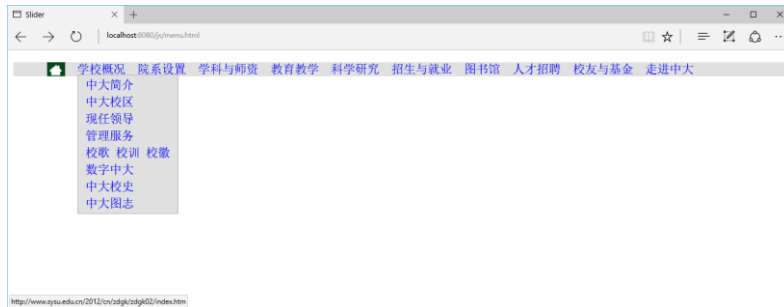
        <div id="arrow1" class="arrow" onClick="arrowonclick(0)"></div>
        <div id="arrow2" class="arrow" onClick="arrowonclick(1)"></div>
    </div>
</body>
</html>
<script>
    var pos=0;
    var cnt=0;
    var flag=true;
    var ObjImg=document.querySelectorAll(".img");
    var ObjPoints=document.querySelectorAll(".point");
    var ObjArrow=document.querySelectorAll(".arrow");
    var ObjImgout=document.querySelector("#imgs")
    function moveImg() { //移动图片
```



```
if(flag){ //鼠标没有移上去
    pos=pos-7.7;
    cnt++;
    if(pos<=-2309)pos=0;
    ObjImgout.style.left=pos+"px";
    if(pos<=-2309)changePoint(0);
    else if(pos<=-1539)changePoint(2);
    else if(pos<=-769)changePoint(1);
    else if(pos<=0)changePoint(0);
}
if(cnt<100)//移动中
    setTimeout("moveImg()",20);
else if(cnt==100){//移动完成
    cnt=0;
    setTimeout("moveImg()",2000);
}
}
function onclickPoint(i){//点击圆点
    if(i==0)pos=0;
    else if(i==1)pos=-770;
    else pos=-1540;
    cnt=0;
    ObjImgout.style.left=pos+"px";
    changePoint(i);
}
function changePoint(i){ //改变圆点背景位置
    for(j=0;j<3;j++){
        if(i==j)ObjPoints[j].classList.add("point1");
        else ObjPoints[j].classList.remove("point1");
    }
}
function movearrow(i){ //显示 隐藏 箭头
    if(i==0){
        flag=false;
        ObjArrow[0].style.display="block";
        ObjArrow[1].style.display="block";
    }else{
        flag=true;
        ObjArrow[0].style.display="none";
        ObjArrow[1].style.display="none";
    }
}
function arrowonclick(i){ //点击箭头
    if(i==0){
        if(pos<=-2310)pos=-1540;
        else if(pos<=-1540)pos=-770;
        else if(pos<=-770)pos=0;
        else if(pos<=0)pos=-1540;
    }else{
        if(pos<=-2310)pos=-770;
        else if(pos<=-1540)pos=0;
        else if(pos<=-770)pos=-1540;
        else if(pos<=0)pos=-770;
    }
    cnt=0;
    ObjImgout.style.left=pos+"px";
    if(pos<=-2309)changePoint(0);
    else if(pos<=-1539)changePoint(2);
    else if(pos<=-769)changePoint(1);
    else if(pos<=0)changePoint(0);
}
}
</script>
<script>moveImg();</script>
```



4、用 JavaScript 制作菜单（参考 html 实验 8 的第 3 题）：



参考资料：javascript(下)v2.pdf 事件

参考视频：menu.mp4

要求和说明：

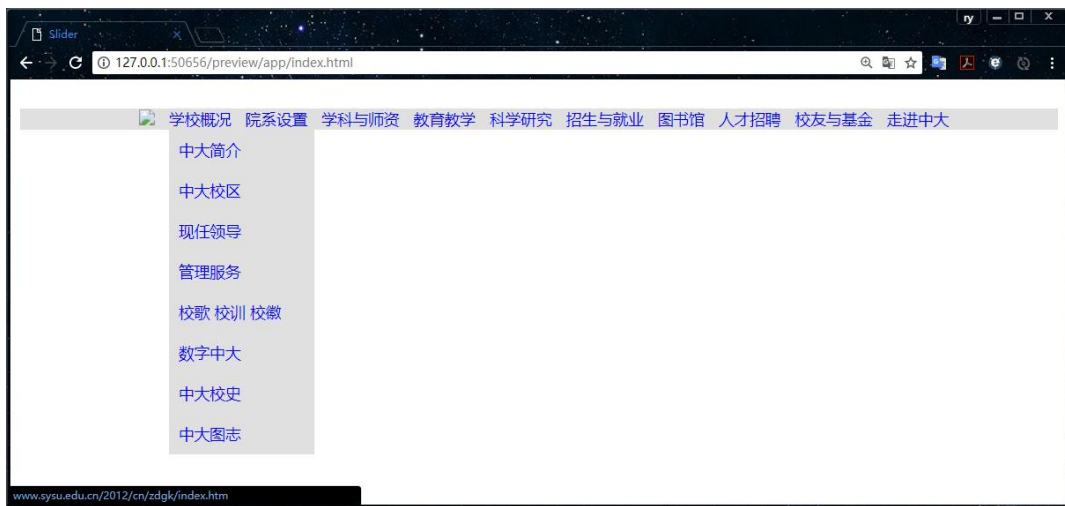
在不改动 html 的 body 内的已设计元素的情况下，如，不能为元素增加类和 id，实现下面功能：

- (1) 先用 css 制作好下拉菜单。
- (2) 然后取消 css 的 hover 功能，并用 mouseover 和 mouseout 进行实现。

编程参考：

- (1) home 图片采用背景图实现
- (2) 通过 document.querySelectorAll("#nav>li")取到主菜单（数组）。
- (3) 通过闭包和 addEventListener 设置鼠标 mouseover 和 mouseout 事件：
循环执行函数(function(ind){})(i)来绑定每个菜单项的事件

完成后的屏幕截图：



这里图片略带毒性怎么都不显示。。。

完成后的 html 源码(menu.html 或 menu.jsp)：

```
<!DOCTYPE html>
<html>
<head>
<title>menu</title>
<base href="http://www.sysu.edu.cn/2012/cn/">
<style>
a:link,a:visited {text-decoration:none;color:blue}
*{padding: 0;}
body{margin: 30px 10px;}
#nav{margin: auto; text-align: center;}
#nav>li ul li{display:inline-block;padding:10px;}
#nav,#nav>li>ul {background: rgba(224, 224, 224, 1);}
#nav>li{position:relative;display:inline-block;}
#nav>li{margin-left:10px;}
```



```
#nav>li>ul{position:absolute;width:140px;list-style-type:none;left:0;text-align:
left;display: none}
</style>
<script>
</script>
</head>
<body>
<ul id="nav">
<li><a href="index.htm"><span class="home">&nbsp;&nbsp;&nbsp;</span></a>
</li>
<li><a href="zdgk/index.htm"><span>学校概况</span></a>
<ul> <li><a href="zdgk/zdgk01/index.htm">中大简介</a></li>
<li><a href="zdgk/zdgk02/index.htm">中大校区</a></li>
<li><a href="zdgk/zdgk03/index.htm">现任领导</a></li>
<li><a href="zdgk/zdgk04/index.htm">管理服务</a></li>
<li><a href="zdgk/zdgk06/index.htm">校歌 校训 校徽</a></li>
<li><a href="zdgk/zdgk07/index.htm">数字中大</a></li>
<li><a href="zdgk/zdgk05/index.htm">中大校史</a></li>
<li><a href="zdgk/zdgk08/index.htm">中大图志</a></li>
</ul>
</li>
<li><a href="yx/index.htm"><span>院系设置</span></a>
</li>
<li><a href="xksz/index.htm"><span>学科与师资</span></a>
<ul>
<li><a href="xksz/xksz01/index.htm">学科建设</a></li>
<li><a href="xksz/xksz02/index.htm">人才名录</a></li>
<li><a href="xksz/xksz03/index.htm">名师垂范</a></li>
</ul>
</li>
<li><a href="jyjsx/index.htm"><span>教育教学</span></a>
<ul>
<li><a href="jyjsx/jyjsx01/index.htm">本科教育</a></li>
<li><a href="jyjsx/jyjsx02/index.htm">研究生教育</a></li>
<li><a href="jyjsx/jyjsx03/index.htm">国际教育</a></li>
<li><a href="jyjsx/jyjsx04/index.htm">继续教育</a></li>
</ul>
</li>
<li><a href="kxyj/index.htm"><span>科学研究</span></a>
<ul>
<li><a href="kxyj/kxyj01/index.htm">重点研究机构</a></li>
<li><a href="kxyj/kxyj04/index.htm">学术期刊</a></li>
<li><a href="kxyj/kxyj05/index.htm">博士后科研流动站</a></li>
</ul>
</li>
<li><a href="zszy/index.htm"><span>招生与就业</span></a>
<ul>
<li><a href="zszy/zszy01/index.htm">本科招生</a></li>
<li><a href="zszy/zszy02/index.htm">研究生招生</a></li>
<li><a href="zszy/zszy03/index.htm">留学生招生</a></li>
<li><a href="jyjsx/jyjsx04/index.htm">继续教育招生</a></li>
<li><a href="zszy/zszy04/index.htm">就业指导与服务</a></li>
</ul>
</li>
<li><a href="http://library.sysu.edu.cn"><span>图书馆</span></a>
</li>
<li><a href="rczp/index.htm"><span>人才招聘</span></a>
</li>
<li><a href="xyjj/index.htm"><span>校友与基金</span></a>
</li>
<li><a href="zjzd/index.htm"><span>走进中大</span></a>
<ul>
<li><a href="http://myspace.sysu.edu.cn">5D 空间</a></li>
<li><a href="zjzd/zjzd02/index.htm">校区地图</a></li>
<li><a href="zjzd/zjzd03/index.htm">生活服务</a></li>
<li><a href="zjzd/zjzd04/index.htm">中大博物馆</a></li>
<li><a href="zjzd/zjzd05/index.htm">校园文化</a></li>
</ul>
</li>
</ul>
</li>
</ul>
```



```
<script>
var t=document.querySelectorAll("#nav>li>ul");
var home=document.querySelector("#nav>li");
home.innerHTML="<a href='index.htm'><span><img style='height=20px'
src='images/home.gif'></span></a>";
for(var i=0;i<t.length;i++){
(function(i){

t[i].parentNode.onmouseover=function(){t[i].style.display="inline-block";}
t[i].parentNode.onmouseout=function(){t[i].style.display="none";}
})(i);
}
</script>
</body>
</html>
```

5、DOM 用正文为空的 dom.html，通过 javascript 的 DOM 操作，实现以下 html 网页(jys.html)：



参考资料：javascript(下).ppt 文档对象模型

实验要求：

- 1、所有节点的创建采用 `document.createElement()`、`document.createTextNode()`和 `e.cloneNode()`实现；加入元素采用 `e.appendChild()`实现。* 不可以使用 `innerHTML`。
可以用 `document.body.appendChild(e)`;把元素加入当前网页。
- 2、先设计第一行，除了第一个上标，第二上标采用克隆第一个上标实现。
- 3、上标中的属性采用 `e.style.textAlign="super"`和 `e.style.fontSize="12px"`实现
- 4、第二行采用克隆第一行实现，利用 `childNodes` 和 `nodeValue` 修改文本节点的内容。
* 其中的 e 表示某个元素节点。

完成后的屏幕截图：





完成后的源码(dom.html 或 dom.jsp):

```
<!DOCTYPE html>
<html>
<head>
  <title>DOM</title>
  <style>
    p {font-size:20px}
  </style>
</head>
<body>
<script>
  var h1 = document.createElement('h1');
  var title=document.createTextNode("静夜思");
  h1.appendChild(title);
  var p = document.createElement('p');
  var t1=document.createTextNode("床前明月光");
  var t2=document.createTextNode(", 疑是地上霜");
  var t3=document.createTextNode("。");
  var t4=document.createTextNode("(1)");
  var note = document.createElement('span');

  document.body.appendChild(h1);
  document.body.appendChild(p);

  note.appendChild(t4);
  var style=document.createAttribute("style");
  style.value="vertical-align:super;font-size:12px";
  note.setAttributeNode(style);

  p.appendChild(t1);
  p.appendChild(note);
  p.appendChild(t2);

  var note2=note.cloneNode(true);
  note2.childNodes[0].nodeValue="(2)";
  p.appendChild(note2);
  p.appendChild(t3);

  var p2=p.cloneNode(true);
  p2.childNodes[0].nodeValue="举头望明月";
  p2.childNodes[1].childNodes[0].nodeValue="(3)";
  p2.childNodes[2].nodeValue="， 低头思故乡";
  p2.childNodes[3].childNodes[0].nodeValue="(4)";
  document.body.appendChild(p2);
</script>
</body>
</html>
```

6、用 ajax-get 技术实现查询功能。

参考资料: javascript(下)v2.pdf AJAX 技术-GET

参考例程: JSP 实验(2)

实验要求:

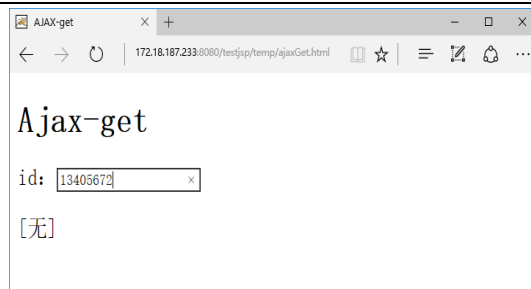
用 ajax-get 技术(ajaxGet.html 和 get.jsp)实现以下功能:

(1)在下面画面中输入 id:

Ajax-get

id:

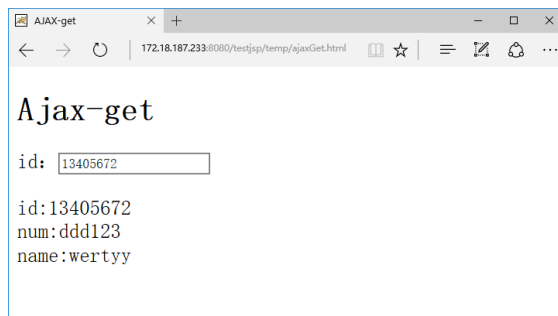
[无]



- (2) 离开 id 输入域后利用 ajax-get 技术调用 get.jsp 从 stu 表中取得 num 和 name 并显示出来，即 alert(xmlhttp.responseText)，返回的格式见下图：



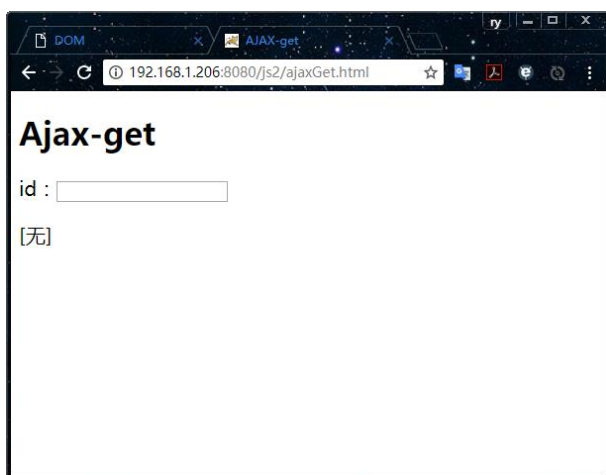
- (3) 关闭弹出框后：



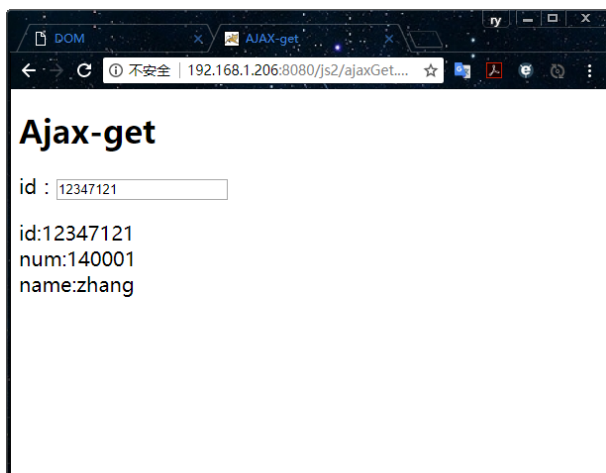
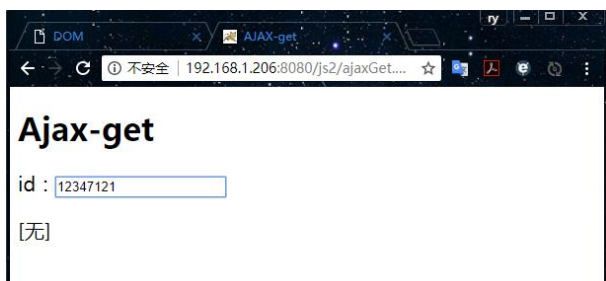
实验说明：

- (1) 用 <http://172.18.187.233:8080/jsp/browseStu.jsp> 查询数据。
MySQL 服务器：172.18.187.233:53306 用户名:user 密码:123
数据库：teaching 表：stu 字段:id,num,name
- (2) 编写 get.jsp 程序可参考 JSP 实验(2)的 browseStu2.jsp，在线调试时可以在文件名加上学号 get15000001.jsp，以免冲突。
- (3) get.jsp 修改后有时要在浏览器中直接运行并刷新才生效。
例如：<http://172.18.187.233:8080/testjsp/temp/get.jsp>
- (4) 离开输入域的事件为 blur，可以采用属性 onblur 触发。

完成后截屏：



查询:



查询失败:



源码 ajaxGet.html 或 ajaxGet.jsp:

```
<!DOCTYPE html>
<html>
<head>
  <title>AJAX-get</title>
  <style>
    p {font-size:20px}
  </style>
</head>
<body>
<h1>Ajax-get</h1>
<p>id: <input id="t1" type="text" onblur="change()" /></p>
<p id="t2">[无]</p>

</body>
</html>
<script language="JavaScript" type="text/javascript">
  function change() {
    var t2=document.getElementById("t2");
    t2.innerHTML="正在查询...";
    setTimeout(show,1);
  }
  function show() {
    var t1=document.getElementById("t1");
    var t2=document.getElementById("t2");
    var xmlhttp = new XMLHttpRequest(); // 创建 http 请求
    xmlhttp.onreadystatechange = function() { // 当 http 请求的状态变化时执行
      if(xmlhttp.readyState == 4) { // 4-已收到 http 响应数据
        if(xmlhttp.status >= 200 && xmlhttp.status < 300 || xmlhttp.status
== 304) { // 200~299-OK 304-unmodified
          alert(xmlhttp.responseText); // http 响应的正文
          t2.innerHTML = xmlhttp.responseText;
        } else{
          alert("error");
        }
      }
    };
    var id="get.jsp?id="+t1.value;
    xmlhttp.open("get",id,true);
    xmlhttp.send(null);
  }
</script>
```

源码 get.jsp:

```
<%@ page language="java" import="java.util.*, java.sql.*"
  contentType="text/html; charset=utf-8"
%>
<%
  request.setCharacterEncoding("utf-8");
  String msg = "";
  String info="";
```



```
String id=request.getParameter("id");
String connectionString = "jdbc:mysql://172.18.187.231:53306/teaching"
+ "?autoReconnect=true&useUnicode=true"
+ "&characterEncoding=UTF-8";
if(id!=null) {
    try {
        Class.forName("com.mysql.jdbc.Driver");
        Connection con = DriverManager.getConnection(connectionString, "user",
"123");
        Statement stmt = con.createStatement();
        ResultSet rs = stmt.executeQuery("select * from stu where id= '" + id+"'");
        if (rs.next()) {
            info+="id:"+rs.getString("id")+"<br>\n";
            info+="num:"+rs.getString("num")+"<br>\n";
            info+="name:"+rs.getString("name")+"<br>\n";
        }
        else info="null";
        rs.close();
        stmt.close();
        con.close();
    } catch (Exception e) {
        msg = e.getMessage();
    }
}
else info="null";
%>
<%=info%>
```

7、用 ajax-post 技术(ajaxPost.html, post.jsp)实现修改数据的功能。

参考资料: Javascript(下)v2.pdf AJAX 技术-POST

实验要求:

(1) 先浏览记录:



浏览学生名单

id	学号	姓名	-
13405651	174391	16791	修改 删除
13405656	111112	56884	修改 删除
13405672	ddd123	aaaaa	修改 删除
13405673	1236ss	qqqqq	修改 删除

[新增](#) [上一页](#) [下一页](#)

(2) 选择输入某个 id 后:

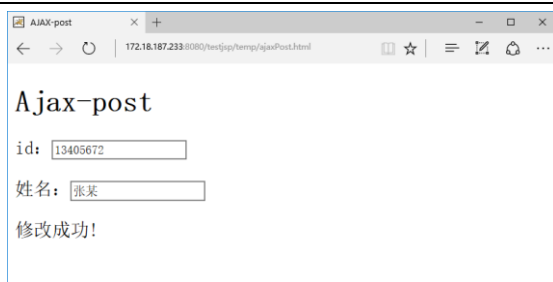


Ajax-post

id:

姓名:

(3) 当输入新的姓名并离开(onblur)时用 ajax 技术调用 post.jsp 修改具有该 id 记录的姓名的值。



(4) 查看修改后的情况:



实验说明:

- (1) 编写 post.jsp 程序可参考 JSP 实验 updateStu.jsp, 在线调试时可以在文件名加上学号 post13000001.jsp, 以避免重名冲突。
- (2) 在线调试用 <http://172.18.187.233:8080/jsp/browseStu.jsp> 查看记录数据和修改结果

完成后截屏:

- (1) 先浏览记录:



- (2) 选择输入某个 id 后:





- (3) 当输入新的姓名并离开(onblur)时用 ajax 技术调用 post.jsp 修改具有该 id 记录的姓名的值。



- (4) 查看修改后的情况:



源码(ajaxPost.html 或 ajaxPost.jsp):

```
<!DOCTYPE html>
<html>
<head>
  <title>AJAX-post</title>
  <style>
    p {font-size:20px}
  </style>
</head>
<body>
  <h1>Ajax-post</h1>
  <p>id: <input id="t1" type="text"></p>
  <p>姓名: <input id="t2" type="text" onBlur="change()"></p>
  <p><span id="t4"></span></p>
  <script language="JavaScript" type="text/javascript">
    function change() {
      var t1=document.getElementById("t1");
      var t2=document.getElementById("t2");
      var t4=document.getElementById("t4");
      var xmlhttp = new XMLHttpRequest(); // 创建 http 请求
      xmlhttp.onreadystatechange = function() { // 当 http 请求的状态变化时执行
        if(xmlhttp.readyState == 4) { // 4-已收到 http 响应数据
          if(xmlhttp.status >= 200 && xmlhttp.status < 300 || xmlhttp.status
== 304) { // 200~299-OK 304-unmodified
            t4.innerHTML = xmlhttp.responseText;
          } else {
            alert("error");
          }
        }
      };
      var v1 = t1.value;
      var v2 = t2.value;
      var param = "id="+ encodeURIComponent(v1) + "&name="+encodeURIComponent(v2); //
汉字需要编码
      xmlhttp.open("post", "post.jsp", true);
      xmlhttp.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
    }
  </script>
</body>
</html>
```




```
xmlhttp.send(param);  
}  
</script>  
</body>  
</html>
```

源码(post.jsp):

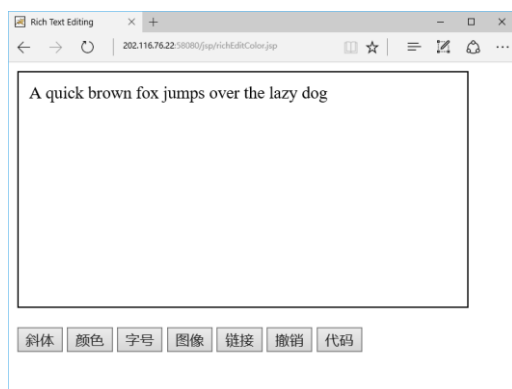
```
<%@ page language="java" import="java.util.*, java.sql.*"  
    contentType="text/html; charset=utf-8" %>  
<%request.setCharacterEncoding("utf-8");  
    String msg = "";  
    String connectString = "jdbc:mysql://172.18.187.231:53306/teaching"  
        + "?autoReconnect=true&useUnicode=true"  
        + "&characterEncoding=UTF-8";  
  
    Class.forName("com.mysql.jdbc.Driver");  
    Connection con=DriverManager.getConnection(connectString, "user", "123");  
    Statement stmt = con.createStatement();  
    if(request.getMethod().equalsIgnoreCase("post")){  
        String id=request.getParameter("id");  
        String name = request.getParameter("name");  
        try{  
            String fmt="update stu set name='%s' where id='%s'";  
            String sql = String.format(fmt,name,id);  
            int cnt = stmt.executeUpdate(sql);  
            if(cnt>0) msg = "修改成功!";  
            else msg="修改失败!";  
            stmt.close();  
            con.close();  
        } catch (Exception e) {  
            msg="修改失败!";  
        }  
    }  
<%>  
<%=msg%>
```

8、采用 div 实现以下富文本编辑。

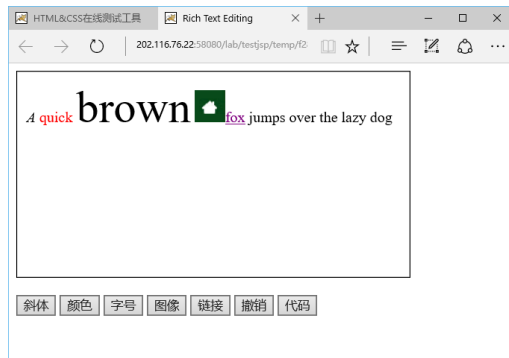
参考资料：Javascript(下)v2.pdf p49-p51 和附录 4

实验要求：

(1) 运行开始：



(2) 依次选中一个词进行设置格式（图像加当前词在后面）：（*字号为 7）

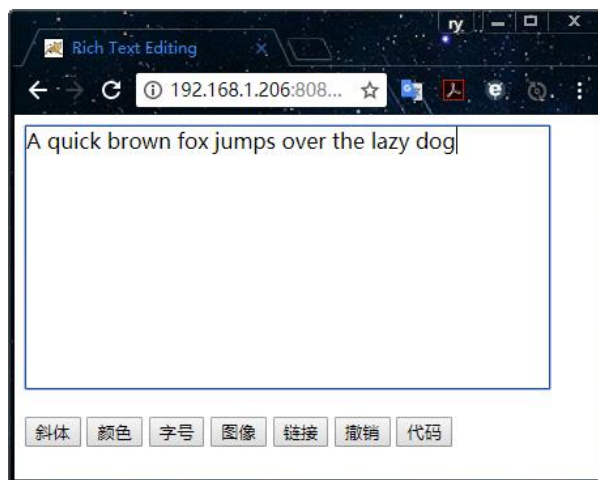


(3) 按了撤销按钮后可以撤销前面的操作，按了代码按钮后会显示编辑框中实际的代码：



完成后截屏：

(1) 运行开始：



(2) 依次选中一个词进行设置格式：



(3) 按了撤销按钮后可以撤销前面的操作，按了代码按钮后会显示编辑框中实际的代码：



源码 richEdit.jsp:

```
<%@ page language="java" import="java.util.*, java.io.*"
    contentType="text/html; charset=utf-8"%>
<!DOCTYPE html>
<html><head><title>Rich Text Editing</title></head>
<body>
<div id="cs" name="edit" style="height:200px; width:400px; border:solid 1px black"
    contenteditable="true">
</div><br>
<input type="button" value="斜体" onclick="italic()">
<input type="button" value="颜色" onclick="color()">
<input type="button" value="字号" onclick="textsize()">
<input type="button" value="图像" onclick="imgs()">
<input type="button" value="链接" onclick="srcs()">
<input type="button" value="撤销" onclick="clears()">
<input type="button" value="代码" onclick="show()">
<script>
    function italic() {
        document.execCommand("italic", false, null);
    }
    function color() {
        document.execCommand("forecolor", false, "red");
    }
    function textsize() {
        document.execCommand("fontsize", false, 7);
    }
    function imgs() {
        document.execCommand("insertimage", false, "images/home.gif");
    }
    function srcs() {
        document.execCommand("createlink", false, "http://www.sysu.edu.cn");
    }
    function clears() {
        var unuse = document.execCommand('undo');
    }
    function show() {
        var cs=document.getElementById("cs");
        alert(cs.innerHTML);
    }
</script>
</body>
</html>
```

- 9、(message.html)编写一个消息框弹出程序，当无输入值时显示提示信息，否则显示输入值。要求消息框定位在浏览器客户区中央（参见 position.html），并且整个窗口(包括被遮蔽的网页部分)变灰且禁止操作（参见 opacity.html），直到点击信息框的 OK 按钮才变正常。消息框要求有标题栏和关闭按钮。只改样式和脚本部分。

* 采用 querySelector()选择元素。

说明：

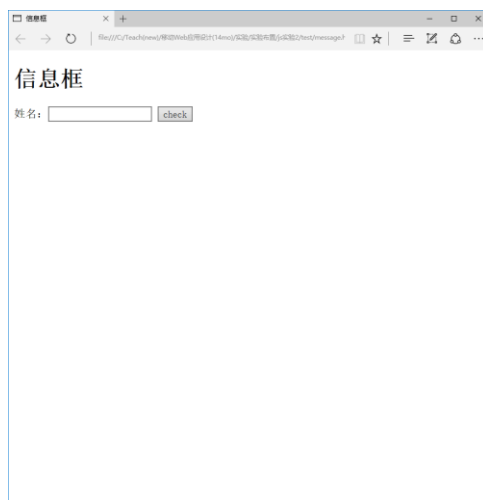
document.documentElement.clientHeight 浏览器客户区的高度



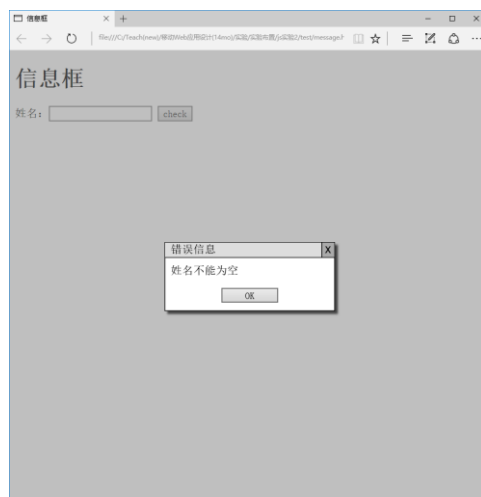
`document.documentElement.scrollHeight` 整个网页的高度（包含滚出客户区的部分）



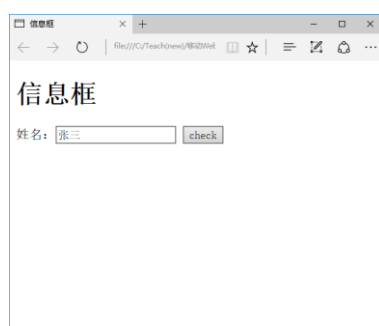
参考显示:



点击 check 之后



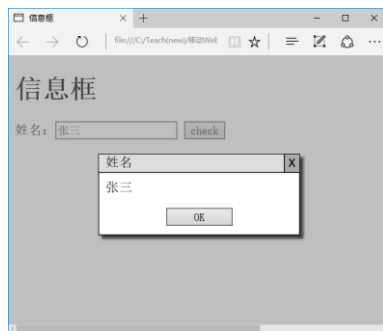
点击 OK, 输入姓名:





(窗口变小了)

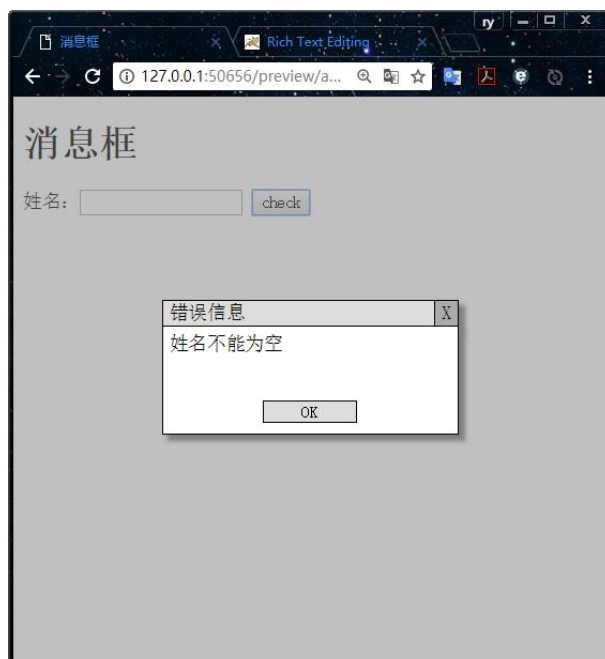
点击 check 后:



完成实验后截屏:



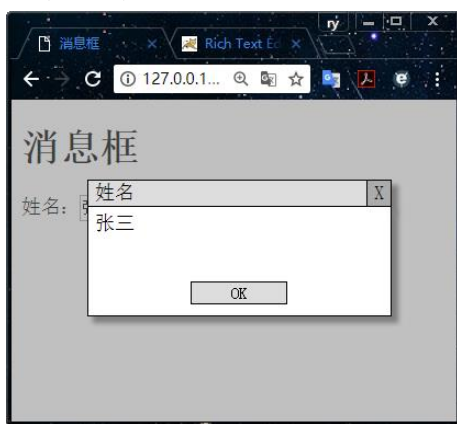
点击 check 之后



点击 OK, 改变窗口大小, 输入姓名:



点击 check 后:



源程序 (message.html):

```
<!DOCTYPE html>
<html lang="zh-cn">
<head>
  <title>消息框</title>
  <style>
    * {font-family:"宋体"}
    #shadow {
      position:absolute;
      display:block;
      top:0;
      left:0;
      z-index:998;
      background:gray;
      opacity: 0.5;
    }
    #msgbox1{
      z-index:999;
      position:absolute;
      margin: auto;
      display:none;
    }
    .msgheader{
      position:relative;
      width: 250px;
      height: 22px;
      border: 1px solid black;
      background-color:rgb(221, 221, 221);
      box-shadow:5px 5px 5px gray;
    }
    .msgtitle {
      position:absolute;
      height:20px;
      padding: 1px 5px;
    }
    .msgclose{
      position:absolute;
```




```
        right:-1px;
        top:-1px;
        border: 1px solid black;
        height:20px;
        width:9px;
        padding: 1px 5px;
        background-color:rgb(172,172,172);
    }
    .msgcontent{
        position: absolute;
        background: #FFF;
        top:22px;
        width: 240px;
        height: 80px;
        border: 1px solid black;
        padding: 5px;
        box-shadow:5px 5px 5px gray;
    }
    .btns{
        position: absolute;
        width:80px;
        left: 85px;
        top: 85px;
    }
    .ok{
        width:80px;
        border:1px solid black;
        text-align: center;
    }
</style>
</head>
<body>
<h1>消息框</h1>
<script>
    function showMsgBox() {
        pos();
        var name=document.querySelector("#name");
        var msgtitle=document.querySelector(".msgtitle");
        var msg=document.querySelector(".msgcontent");
        if(name.value==""){
            msgtitle.innerHTML="错误信息";
            msg.innerHTML="姓名不能为空";
        }
        else{
            msgtitle.innerHTML="姓名";
            msg.innerHTML=name.value;
        }
    }
    function hideMsgBox(boxId) {
        var box =document.querySelector("#msgbox1");
        box.style.display="none";
        hideShadow();
    }

    function showShadow() {
        var shadow=document.querySelector("#shadow");
        shadow.style.width="" +document.documentElement.scrollWidth+"px";

if(document.documentElement.clientHeight>document.documentElement.scrollHeight)
    shadow.style.height=document.documentElement.clientHeight+"px";
    else
        shadow.style.height=document.documentElement.scrollHeight+"px";
        shadow.style.display="block";
    }

    function hideShadow() {
        var shadow=document.querySelector("#shadow");
        shadow.style.display="none";
    }

    function pos() {
```



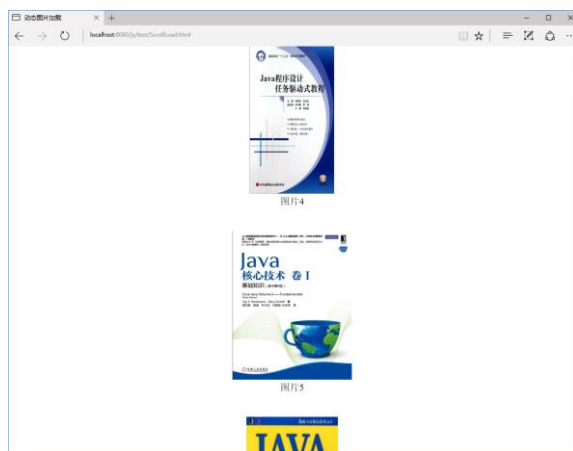
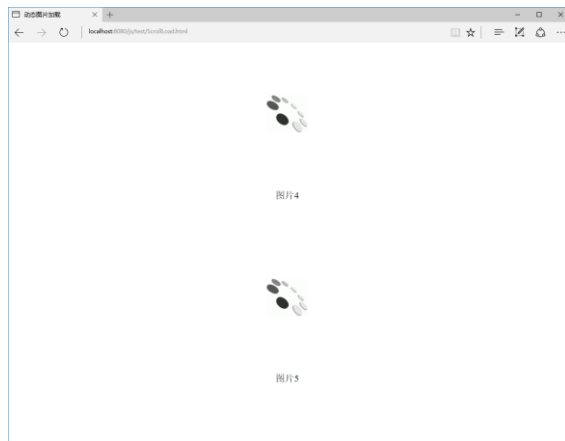
```
var box=document.querySelector("#msgbox1");
box.style.display="inline-block";
box.style.top=document.documentElement.clientHeight/2-60+"px";
box.style.left=document.documentElement.scrollWidth/2-126+"px";
}
function check(){
    showShadow();
    showMsgBox();
}
</script>
<div id="shadow">
</div>
<div id="msgbox1" class="msg msgbox">
    <div class="msgheader msg">
        <div class="msgtitle msg">这是标题</div>
        <div class="msgclose msg" onclick="hideMsgBox('msgbox1')">X</div>
    </div>
    <div class="msgcontent msg">这是内容
    </div>
    <div class="btns"> <!--弹出框的 OK 按钮，按了就触发 hideMsgBox 函数 -->
        <input type="button" class="ok" value="OK"
onclick="hideMsgBox('msgbox1')">
    </div>
</div>
姓名: <input type="text" id="name">
<input type="button" onclick="check()" value="check">
</body>
</html>
```

10、(ScrollLoad.html)为了节约流量，让网页中的图像只有出现在浏览器客户区时才显示，见 ScrollLoad.mp4。方法是用自定义属性 xSrc 保存图像的 URL，在网页滚动时对网页中所有图像(image 元素)进行判断，看看它们的上边沿或下边沿是否出现在浏览器客户区，如果是，则把 xSrc 的值拷贝到属性 Src 中（用 `getAttribute` 和 `setAttribute` 方法），此时浏览器才去下载并显示图像。 参考：Javascript(下)v2.pdf p53 p28-29

参考显示：



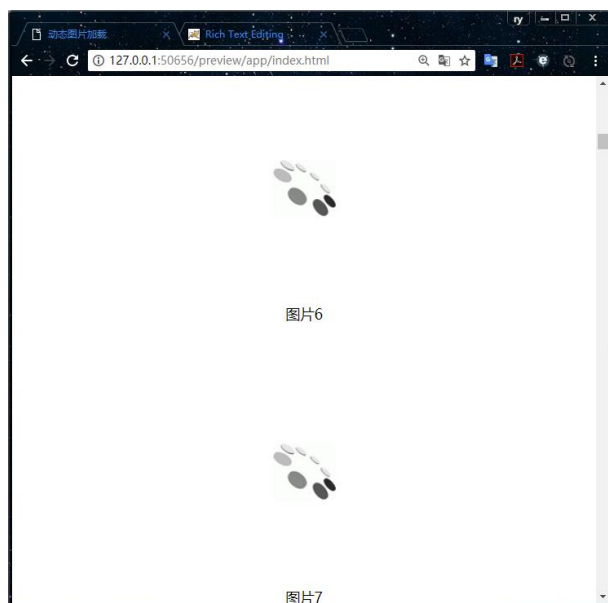
上滚之后：



完成实验后截屏：



上滚之后：



源程序（ScrollLoad.html）:

```
<!DOCTYPE html>
<html>
<head>
  <meta content="text/html;charset=utf-8" /><title>动态图片加载</title>
  <style>
    body {text-align: center;}
    .list {margin-bottom: 40px;}
  </style>
</head>
<body>
<h1>动态图片加载</h1>
<div id="content">
</div>
<script type="text/javascript">
  var allImgs = "";
  var imgPath="http://172.18.187.11:8080/imgs/";
  for (var i = 1; i < 60; i++) {
    allImgs += '<div class="list"><br />图片' + i + ' </div>' ;
  }
  document.getElementById("content").innerHTML = allImgs;
```



```
function windowHeight() { //获取页面浏览器客户区的高度
    return
    (document.compatMode=="CSS1Compat"?document.documentElement.clientHeight
    :document.body.clientHeight;
}
function showImages() {
    var docImg = document.images;
    var cnt = docImg.length;
    if (!cnt) return false;
    for (var i = 0; i < cnt; i++) {
        var o = docImg[i], imgHeight = 250;
        if(docImg[i].getBoundingClientRect().top > 0 &&
docImg[i].getBoundingClientRect().top < windowHeight()){
            docImg[i].src=docImg[i].getAttribute("xSrc");
        }
    }
};
window.onload = function() {
    setTimeout(function() {showImages()}, 1200);
}
showImages();
</script>
</body>
</html>
```