学习方法

1. 建立逻辑思维
   1. 多分析
   2. 理解逻辑算法
   3. 多练、多敲、多读
2. 编程习惯
3. 大胆动手
4. 保持好奇好
5. 乐观面对错误

就业能力

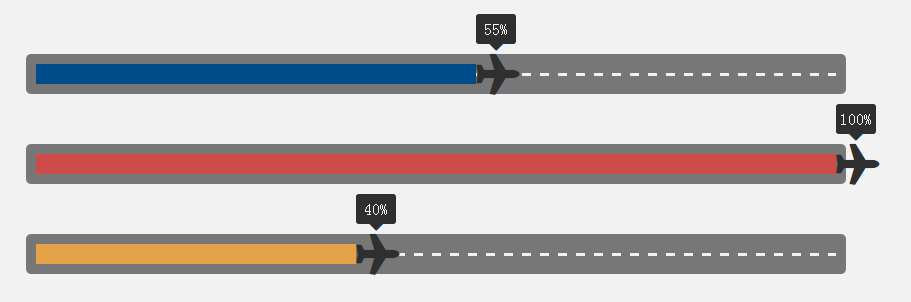
1. 分析能力
2. 独立思考能力
3. 程序阅读能力
4. 程序排错能力
5. 独立编写程序能力
6. 团队协作能力
7. 学习能力
8. 善于总结的能力

专业技能

DOM（下）

1. 自定义属性及getAttribute方法
   1. 自定义属性：由程序员自己给标签添加的属性。
   2. getAttribute():获取特定元素节点属性的值
      1. 直接使用自定义的属性名来获取自定义属性时，IE最新版本不支持，其他浏览器不支持，因此要采用这个方法来访问。
   3. 扩展：setAttribute():可以修改或添加元素节点的属性
   4. 扩展：removeAttribute():移除特定元素节点属性
2. outerHTML/innerText(非W3C)
   * 1. outerHTML:设置或获取对象及其内容的HTML形式
     2. innerText:设置或获取位于对象起始和结束标签内的文本（不包含HTML文本）
     3. innerHTML:设置或获取位于对象起始和结束标签内的文本（包含HTML文本）
3. childNodes/过滤空白节点
   1. childNodes属性：获取到指定节点中的所有子节点
      1. 返回元素节点和文本节点两种
   2. 过滤空白节点
      1. 在低版本IE中，会自动忽略掉空白的文本节点，其他浏览器不会忽略
      2. 可以编写一个手工过滤空白节点的方法
      3. 或者编写一个删除空白节点的方法
      4. 匹配空白节点可以使用正则：/^\s+$/.test(子节点.nodeValue)
      5. 忽略空白文本节点  
         function noSpace(node){  
          var arr = [];  
          //遍历所有的子节点  
          for(var i = 0; i < node.length; i++){  
          //判断节点类型是文本且文本中包含空白  
          if(node[i].nodeType == 3 && /^\s+$/.test(node[i].nodeValue)){  
          continue;  
          }else{  
          arr.push(node[i]); //将非空白节点添加到新的数组中  
          }  
          }  
          return arr;  
         }
      6. 删除空白文本节点  
         function removeSpace(node){  
          for(var i = 0; i < node.length; i++){  
          //判断节点类型是文本且文本中包含空白  
          if(node[i].nodeType == 3 && /^\s+$/.test(node[i].nodeValue)){  
          //移除文本节点中的空白  
          node[i].parentNode.removeChild(node[i]);  
          }  
          }  
          return node;  
          }
4. 高级选取firstChild/lastChild/parentNode/previousSibling/nextSibling
   1. 层次节点属性：节点的层次结构可以划分为：父节点与子节点、兄弟节点这两种。
   2. firstChild:用于获取当前元素节点的第一个子节点，相当于childNodes[0];
   3. lastChild:用于获取当前元素节点的最后一个子节点，相当于childNodes[box.childNodes.length – 1]
   4. parentNode：返回该节点的父节点。
   5. previousSibling:返回该节点的前一个同级节点。
   6. nextSibling:返回该节点的后一个同级节点。

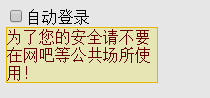
第二节：应用

1. 隔行变色
2. 进度条  
   

第三节

1. 获取非行内样式（兼容问题）
   1. function getStyle(obj,attr){  
       if(obj.currentStyle){  
       return obj.currentStyle[attr];  
       }else{  
       return getComputedStyle(obj,false)[attr];  
       }  
       }
2. createTextNode文本节点
3. 父节点.insertBefore(newNode,existNode)
4. offsetWidth/offsetHeight
   1. obj.offsetWidth 指 obj 对象自身的宽度，整型，单位像素。
      1. offsetWidth = width + padding + border
   2. obj.offsetHeight 指 obj 对象自身的高度，整型，单位像素。
      1. offsetHeight = height + padding + border
5. offsetLeft/offsetTop
   1. obj.offsetLeft 指 obj 距离左方或上层对象的位置，整型，单位像素。
   2. obj.offsetTop 指 obj 距离上方或上层对象的位置，整型，单位像素。

第四节：应用

1. 网页换肤
2. 简易年历
3. 自动登录勾选  
   

第五节：综合应用

1. 点击按钮换图片
2. Tab切换案例
3. 扩展案例：QQ延迟提示框

周六练习

1. 苹果菜单（扩展：offsetParent）  
   offsetParent属性返回一个对象的引用，这个对象是距离调用offsetParent的元素最近的（在包含层次中最靠近的），并且是已进行过CSS定位的容器元素。 如果这个容器元素未进行CSS定位, 则offsetParent属性的取值为根元素(在标准兼容模式下为html元素；在怪异呈现模式下为body元素)的引用。 当容器元素的style.display 被设置为 "none"时（译注：IE和Opera除外），offsetParent属性 返回 null。
2. 扩展案例：密码强度  
   