



院系 数据科学与计算机学院 学号 17341213 姓名 郑康泽

【实验题目】HTML&CSS 实验(6)

【实验目的】学习网页的版块设计和菜单设计。

【实验工具】

采用记事本编辑，也可以采用 Visual Studio、Dreamweaver 等具有网页编辑功能的软件打开并编辑页面，或者直接在 Eclipse 或 netbeans 或 intellij idea 下编辑运行。

或者使用在线调试：<http://103.26.79.35:8080/lab/html6/>

【安全设置】

如果网页中有 xss 安全问题，在 Chrome 下运行在线调试会出现 xss 审计错误。解决方法是，先关闭所有 Chrome 窗口，然后找到 Chrome 程序，并使用以下命令启动 Chrome：

```
C:\Users\jsszym\AppData\Local\Google\Chrome\Application\chrome.exe --args
```

```
--disable-xss-auditor
```

也可以在快捷方式中加入参数。

【兼容问题】

少数题目有兼容性问题，例如，做立体按钮一题 Safari 显示有点问题。

【实验内容】

- 1、(board1.html) 参考源码中给出的部分样式，并在只修改样式的情况下按照下图设计一个类似的版块。要求：多个 div 采用 inline-block 放于一行，通过增加前面元素的宽度进行对齐。列表图标通过:after 加上。 * 这里的修改表示增改删。



* .bulletin 加上背景色(WhiteSmoke)参考一下：



特别的样式属性提示：vertical-align

完成后截图：



样式源代码（board1.html）：

```
<style>
* {
    margin: 0;
    padding: 0;
    line-height: 1.2em;
}
*, a:link, a:visited, a:hover, a:active
{
    text-decoration: none;
    font-size: 12px;
    font-family: 宋体;
}
.clear {clear: both;}
li {list-style-type: none;}
#board1 {
    width: 310px;
    margin: 50px auto;
    border: 1px solid black;
    background-color: whitesmoke;
}
.boardmenu {
    margin-bottom: 10px;
    padding: 5px;
    height: 18px;
    border-bottom: dotted #a17525;
}
.boardmenu .title {
    display: inline-block;
    width: 250px;
}
.boardmenu .title img {vertical-align: text-top;}
.boardmenu .title span {
    color: #a17525;
    font-weight: bold;
}
```



```
font-family: 黑体;
margin-left: 5px;
}
.boardmenu .more {display: inline-block;}
.boardmenu .more a {
    font-size: 13px;
    font-family: sans-serif;
    color: #a17525;
}

.boardcontent li {
    height: 26px;
    margin: auto 5px;
}

.boardcontent a {
    font-size: 12px;
    color: #6f6f6f;
    display: inline-block;
    width: 260px;
}

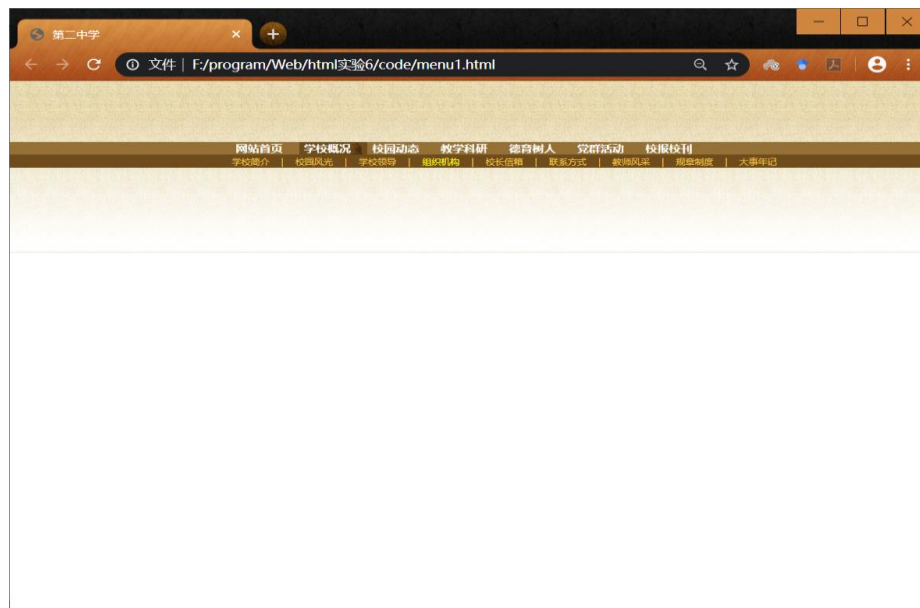
.boardcontent li:before {
    content: url('./images/list_icon.gif')
}

span {text-align: right;}
</style>
```

- 2、(menu1.html) 参考源码中给出的部分样式，并在只修改样式的情况下按照下图设计一个类似的版块，动图见 menu1.mp4。要求：菜单项(li)采用浮动方式布局，整个菜单栏#nav 采用相对定位，选中的主菜单背景采用过渡设计(background-position-y)，子菜单.subnav 采用绝对定位。



特别的样式属性提示：background-position-x, background-position-y, background-clip, vertical-align, transition-property, transition-duration
颜色：#6f4c1b, #f1c149 字体：微软雅黑
完成后截图：



样式源代码:

```
<style>
    * {
        margin: 0;
        padding: 0;
        line-height: 1.2em;
    }
    *,
    a:link,
    a:visited,
    a:hover,
    a:active {
        text-decoration: none;
        font-size: 12px;
        font-family: 宋体;
    }
    .clear {
        clear: both;
    }
    li {
        list-style-type: none;
    }
    #board1 {
        width: 310px;
        margin: 50px auto;
        background-color: whitesmoke;
    }
    .boardmenu {
        margin-bottom: 10px;
        padding: 5px;
        height: 18px;
    }
```



```
border-bottom: dotted #a17525;
}
.boardmenu .title {
    display: inline-block;
    width: 250px;
}
.boardmenu .title img {
    vertical-align: text-top;
}
.boardmenu .title span {
    color: #a17525;
    font-weight: bold;
    font-family: 黑体;
    margin-left: 5px;
}
.boardmenu .more {
    float: right
}
.boardmenu .more a {
    font-size: 13px;
    font-family: sans-serif;
    color: #a17525;
}
.boardcontent li {
    height: 26px;
    margin: auto 5px;
}
.boardcontent a {
    font-size: 12px;
    color: #6f6f6f;
    display: inline-block;
    width: 260px;
}
.boardcontent li:before {
    content: url('./images/list_icon.gif')
}
.boardcontent span {
    float: right;
}
</style>
```

- 3、(menu2.html) 在 menu1.html 的基础上，并在只修改样式的情况下设计子菜单为下拉菜单，动图见 menu2.mp4，下拉增加了 opacity 的过渡(transition)。提示：每个主菜单项(li)要改为相对定位。
参考截图：



中山大學
SUN YAT-SEN UNIVERSITY

实验报告



完成后截图：



样式源代码：

```
<style>
    * {
        margin: 0;
        padding: 0;
        font-size: 12px;
        font-family: 宋体;
        line-height: 1.2em
    }
    a:link,
    a:visited,
    a:hover,
    a:active {
        text-decoration: none;
    }
    .wrapper {
        margin: 0 auto;
        width: 980px;
        position: relative;
    }
    li {
        list-style-type: none;
    }
```

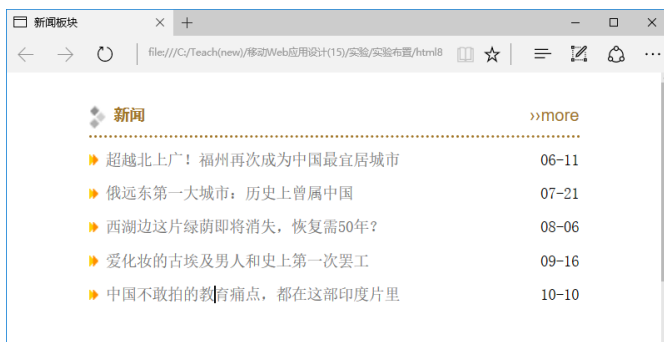


```
#header_outer {  
    height: 140px;  
    background: url("images/header_bg.png") repeat-x;  
}  
  
#header {overflow: hidden; }  
  
#nav_outer {  
    height: 30px;  
    background-image: url('images/nav_bg.jpg');  
}  
  
#nav {position: relative; }  
  
#nav span, #nav a {  
    font-size: 14px;  
    font-family: 'Microsoft YaHei';  
}  
  
#nav>li {  
    width: 140px;  
    height: 30px;  
    background-image: url('../images/nav_bg2.gif');  
    background-position-y: -64px;  
    transition: background-position-y;  
    transition-duration: 0.5s;  
    float: left;  
    position: relative;  
    padding-top: 2px;  
}  
  
#nav>li>a {  
    width: 140px;  
    color: white;  
    font-weight: bold;  
    font-size: 1.5em;  
    padding-left: 30px;  
}  
  
#nav>li:hover {  
    background-position-y: -32px;  
}  
  
.subnav>li {  
    background-image: url('../images/nav_bg.jpg');  
    width: 140px;  
    height: 30px;  
    text-align: center;  
    padding-top: 10px;  
}  
  
.subnav>li:hover {opacity: 0.7}  
  
.subnav a {  
    color: white;  
    margin: 12px;
```



```
}  
  
#nav .subnav {  
    height: 26px;  
    width: 1160px;  
    padding-top: 8px;  
    position: absolute;  
    top: 21px;  
    opacity: 0;  
    transition: opacity 0.8s;  
}  
  
#nav>li: hover .subnav {  
    opacity: 1;  
}  
  
#nav .subnav>li: hover a {  
    color: yellow;  
}  
  
#main_outer {  
    height: 500px;  
    background: url("../images/header_bg.png") repeat-x;  
    background-position: 0 -180px;  
}  
  
</style>
```

4、(board2.html) 为了在版块宽度(#board1)变化(可以手动修改)时版面自动对齐, 见下图, 尝试把 board1.html 中的 inline-block 改为浮动方式。



完成后截图:



样式源代码: *只需写一个宽度

```
<style>
* {
    margin: 0;
    padding: 0;
    line-height: 1.2em;
}
*, a:link, a:visited, a:hover, a:active
{
    text-decoration: none;
    font-size: 12px;
    font-family: 宋体;
}
.clear {clear: both;}
li {list-style-type: none;}
#board1 {
    width: 310px;
    margin: 50px auto;
    border: 1px solid black;
    background-color: whitesmoke;
}
.boardmenu {
    margin-bottom: 10px;
    padding: 5px;
    height: 18px;
    border-bottom: dotted #a17525;
```



```
}  
.boardmenu .title {  
    display: inline-block;  
    width: 250px;  
}  
.boardmenu .title img {vertical-align: text-top;}  
.boardmenu .title span {  
    color: #a17525;  
    font-weight: bold;  
    font-family: 黑体;  
    margin-left: 5px;  
}  
.boardmenu .more {display: inline-block;}  
.boardmenu .more a {  
    font-size: 13px;  
    font-family: sans-serif;  
    color: #a17525;  
}  
.boardcontent li {  
    height: 26px;  
    margin: auto 5px;  
}  
.boardcontent a {  
    font-size: 12px;  
    color: #6f6f6f;  
    display: inline-block;  
    width: 260px;  
}  
.boardcontent li:before {  
    content: url('./images/list_icon.gif')  
}  
span {text-align: right;}  
</style>
```

5、(menu3.html) 尝试把 menu1.html 中的浮动元素改为 display:inline-block，也可以得到类似的页面。

完成后截图：



样式源代码：

```
<style>
```



```
* {
    margin: 0;
    padding: 0;
    font-size: 12px;
    font-family: 宋体;
    line-height: 1.2em
}
a:link,
a:visited,
a:hover,
a:active {
    text-decoration: none;
}
.wrapper {
    margin: 0 auto;
    width: 1160px;
    position: relative;
}
li {
    list-style-type: none;
    display: inline-block
}
#header_outer {
    height: 140px;
    background: url("images/header_bg.png") repeat-x;
}
#header {
    overflow: hidden;
}
#nav_outer {
    height: 60px;
    background-image: linear-gradient(to bottom, #957137 50%, #6f4c1b 50%);
}
#nav {
    position: relative;
}
#nav span,
#nav a {
    font-size: 14px;
    font-family: 'Microsoft YaHei';
}
#nav>li {
    width: 140px;
    height: 30px;
    background-image: url('./images/nav_bg2.gif');
    background-position-y: -64px;
```



```
transition: background-position-y;  
transition-duration: 0.5s;  
padding-top: 2px;
```

```
#nav>li>a {  
    width: 140px;  
    color: white;  
    font-weight: bold;  
    font-size: 1.5rem;  
    padding-left: 30px;  
  
#nav>li:hover {  
    background-position-y: -32px;  
}  
#nav>li:hover .subnav {  
    visibility: visible  
}  
.subnav a, .subnav span {  
    color: #f1c149;  
    margin: 12px;  
}  
#nav .subnav {  
    height: 26px;  
    width: 1160px;  
    position: absolute;  
    top: 30px;  
    left: -10px;  
    padding-top: 8px;  
    visibility: hidden;  
}  
#nav .subnav>li:hover a {  
    color: yellow;  
}  
#main_outer {  
    height: 500px;  
    background: url("../images/header_bg.png") repeat-x;  
    background-position: 0 -180px;  
}  
</style>
```

6、（diamond.html）利用边框(border)透明色（transparent）做一个钻石形状：

- （1） 利用类 diamond1 做一个梯形(div1)，再利用类 diamond2 做一个三角形(div2)，让它们上下靠在一起；

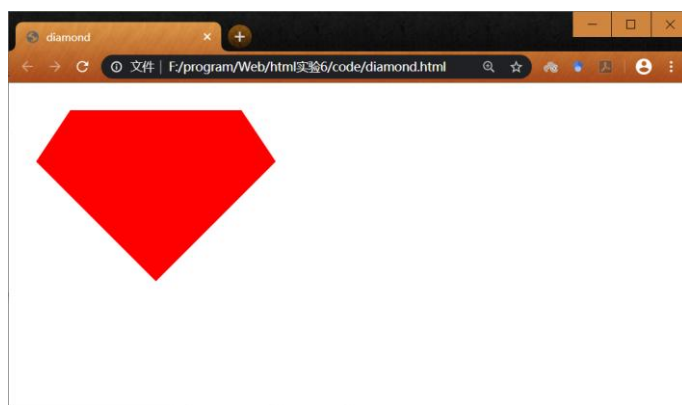




- (2) 类 diamond1 定义为 relative，把类 diamond2 改为 diamond1:before，加入 content: ""，并定义为 absolute，用 left 和 top 移动它显示出钻石形状。



完成后截屏：



样式源代码 (diamond.html)：

```
<!DOCTYPE HTML>
<html>
<head>
  <meta content="text/html; charset=utf-8">
  <title>diamond</title>
  <style>
    .diamond1 {
      height: 0px;
      width: 50px;
      background-color: white;
      border-bottom: 15px red solid;
      border-left: 10px transparent solid;
      border-right: 10px transparent solid;
      position: relative;
    }
    .diamond1::before {
      content: "";
      width: 0px;
      height: 0px;
      border-top: 35px red solid;
      border-left: 35px transparent solid;
      border-right: 35px transparent solid;
      position: absolute;
      top: 15px;
      left: -10px;
    }
  </style>
</head>
</html>
```



```
}  
</style>  
</head>  
<body>  
    <div class="diamond1"></div>  
</body>  
</html>
```

7、(love.html) 利用边框(border)的圆角(border-radius)做一个爱心形状:

(1) 通过类 love1 和 love2 用 border-radius 得到如下形状:



(2) 利用 transform 的 rotate 得到如下形状:



(3) 利用绝对定位调整.love1 和.love2 的位置, 得到下图。然后把.love1 和.love2 分别改为.love:before 和.love:after, 并加入 content: "", 依然得到下图。



完成后截图:



样式:



```
<style>

.love {
    position: relative;
    left: 100px;
}

.love::before {
    content: "";
    width: 50px;
    height: 85px;
    background-color: red;
    border-top-left-radius: 50%;
    border-top-right-radius: 50%;
    transform: rotate(45deg);
    position: absolute;
    left: 25px;
}

.love::after {
    content: "";
    width: 50px;
    height: 85px;
    background-color: red;
    border-top-left-radius: 50%;
    border-top-right-radius: 50%;
    transform: rotate(-45deg);
    position: absolute;
}

</style>
```

8、通过多重投影实现文字阴影(text-shadow.html)。参考实现：

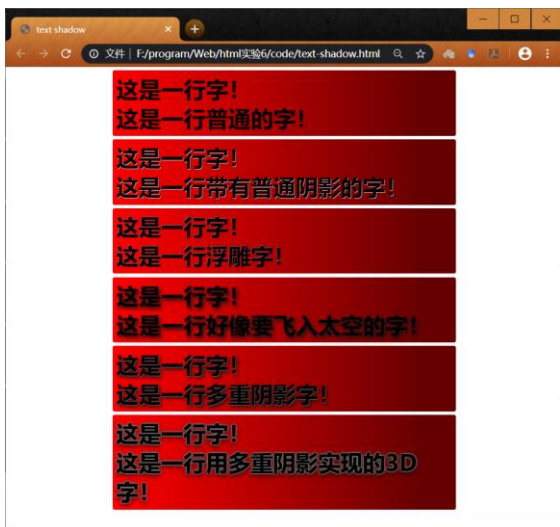


其中，box1~box3 采用 1 重投影，box4 采用 2 重投影，box5 采用多重投影。

box5 的阴影:

- ✚ 用原地投影增加边框，2px 的黑色阴影，四个灰色阴影，颜色分别为 #aaa, #999, #888, #666, 模糊度分别为 2px、4px、6px、8px。
- ✚ 做两个透明度为 0.5 的黑色投影，下移 6px 和 8px，模糊度分别为 6px 和 20px。

完成后截图:



样式:

```
<style>
    html {
        font-size: 16px;
    }
    div {
        font-family: 微软雅黑;
        background: linear-gradient(to right, #F00, #600 90%);
        margin: 10px auto;
```




```
padding: 10px;
border-radius: 0.3333rem;
width: 60rem;
font-size: 4em;
font-weight: 900;
}
#box1 {
  text-shadow: 1px 1px 1px rgb(200, 200, 200)
}
#box2 {
  text-shadow: -1px -1px 1px rgb(100, 100, 100)
}
#box3 {
  text-shadow: 5px 6px 10px rgb(20, 20, 20)
}
#box4 {
  text-shadow: 1px 1px 1px rgb(100, 100, 100), 5px 6px 10px rgb(20, 20, 20)
}
#box5 {
  text-shadow: 0px 0px 2px black, 0px 0px 2px #aaa, 0px 0px 4px #999, 0px 0px
6px #888, 0px 0px 8px #666, 0px 6px 6px rgba(0, 0, 0, 0.5), 0px 8px 20px rgba(0, 0,
0, 0.5)
}
</style>
```

9、(box-shadow.html)通过 inset 阴影形成凸出和凹陷 3D 盒子。原来的两个 2D 盒子如下面第一个图所示。下图(a)到(e)显示了把它们改造为两个 3D 框架(box-shadow.html)的前五个步骤的结果。除非指出，前五个步骤的投影偏移度和模糊度参考下图进行设置。



2D 盒子



(a)



(b)



(c)



(d)



(e)

(a) 先变为渐进颜色:



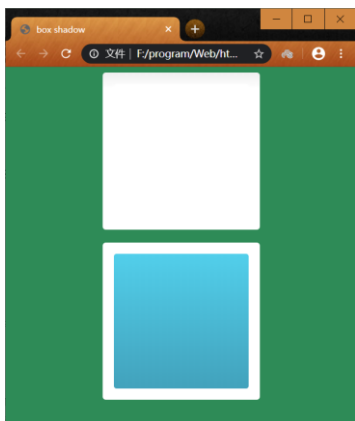
box1, box3: 从浅灰色 rgb(240, 240, 240) 到白色 (rgb(255, 255, 255)) 的 10% 处, 后面全部白色。

box2: 不变。白色, 可能会有点看不清。

box4: 从 rgb(82, 207, 235) 到 rgb(66, 162, 188)。

然后变为圆角, 参考效果见上面的图。

完成后截屏:



增加的样式:

```
<style>
```

```
#box1, #box2, #box3, #box4 {border-radius: 2%;}
```

```
#box1 {
```

```
width: 12rem;
```

```
height: 12rem;
```

```
margin: 0 auto;
```

```
padding: 1rem;
```

```
background: linear-gradient(to bottom, rgb(240,240,240) 0%, white 10%);
```

```
}
```

```
#box2 {
```

```
height: 100%;
```

```
width: 100%;
```

```
background: white;
```

```
}
```

```
#box3 {
```

```
width: 12rem;
```

```
height: 12rem;
```

```
margin: 0 auto;
```

```
padding: 1rem;
```

```
background: white;
```

```
}
```

```
#box4 {
```

```
height: 100%;
```

```
width: 100%;
```

```
background: linear-gradient(to bottom, rgb(82, 207, 235) 0%, rgb(66, 162, 188) 100%);
```

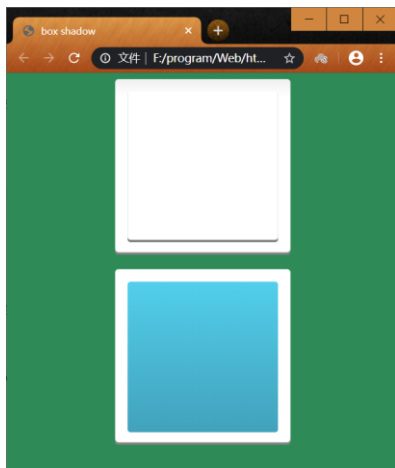
```
}
```

```
</style>
```



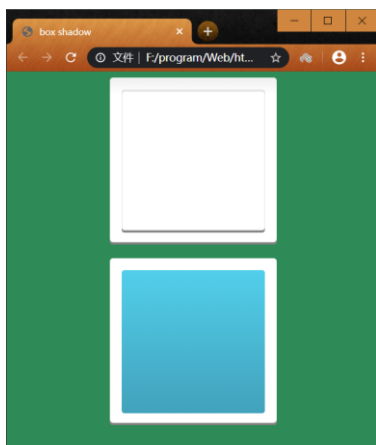
- (b) 三个盒子 (box1, box2, box3) 通过增加一个内嵌 (inset) 上移 (y 方向取负值) 的阴影变为凸出盒子。上移多少决定了凸出的程度。阴影采用 0.5 透明度的黑色，其目的是颜色浅一些，并让盒子的颜色透过。参考效果见前面的图。

完成后截屏：



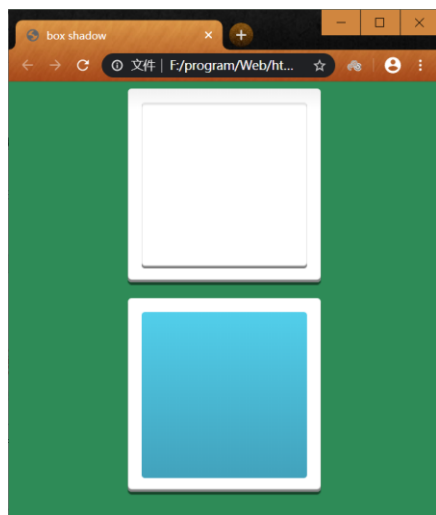
- (c) 内盒子 (box2) 通过增加一个内嵌 (inset) 下移 (y 方向取正值) 阴影变为凹陷盒子。下移多少决定了凹陷的程度。阴影采用 0.1 透明度的黑色 (颜色更浅)。参考效果见前面的图。

完成后截屏：



- (d) 外盒子 (box1, box3) 增加一个下移 (y 方向取正值) 的阴影。阴影采用 0.5 透明度的黑色。参考效果见前面的图。

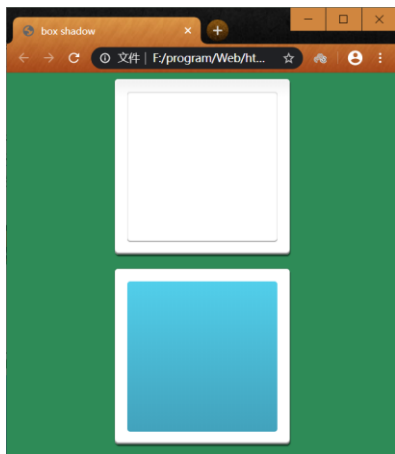
完成后截屏：



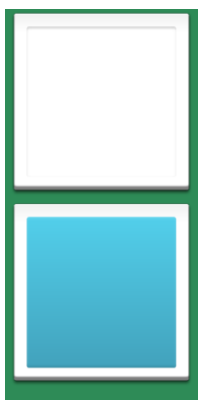


- (e) 外盒子 (box1, box3) 都增加一个下移 1px 的 inset 白色阴影和一个原位置的 1px 的黑色阴影。参考效果见上面的图，上面的盒子红色边 (左和上) 就是 inset 造成的 (本来是白色)，上面的盒子蓝色边 (右和下) 就是另一个阴影造成的 (本来是黑色)。这个主要用于增加一个框，凸出 3D 效果。

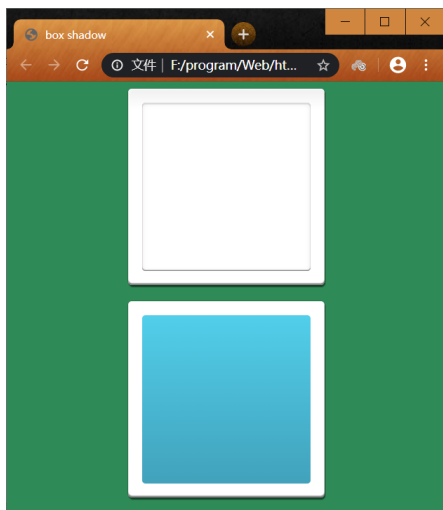
完成后截屏：



- (f) 内盒子 (box2, box4) 也增加一个边框。box2 采用下移 1px 的模糊度为 5px 的 inset 黑色阴影 (透明度 0.1)。box4 采用下移 1px 的模糊度为 5px 的 inset 白色阴影 (透明度 0.2)。最后的效果如下图。



完成后截屏：

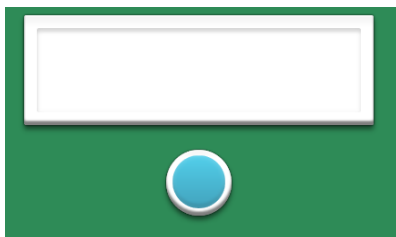


在 (1) 的基础上增加的样式：

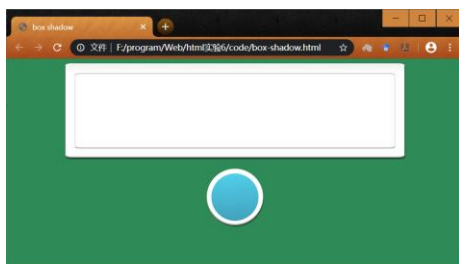


```
<style>
  #box1 {
    width: 12rem;
    height: 12rem;
    margin: 0 auto;
    padding: 1rem;
    background: linear-gradient(to bottom, rgb(240,240,240) 0%, white 10%, white
100%);
    box-shadow: inset 0px -2px 1px rgba(0, 0, 0, 0.5), 1px 2px 1px rgba(0, 0, 0, 0.5),
inset 1px 1px 2px white, 0px 0px 1px black;
  }
  #box2 {
    height: 100%;
    width: 100%;
    background: white;
    box-shadow: inset 0px 2px 1px rgba(0, 0, 0, 0.1), inset 0px -1px 1px rgba(0, 0, 0, 0.5),
inset 0px 1px 5px rgba(0, 0, 0, 0.1);
  }
  #box3 {
    width: 12rem;
    height: 12rem;
    margin: 0 auto;
    padding: 1rem;
    background: white;
    box-shadow: inset 0px -2px 1px rgba(0, 0, 0, 0.5), 1px 2px 1px rgba(0, 0, 0, 0.5),
inset 1px 1px 2px white, 0px 0px 1px black;
  }
  #box4 {
    height: 100%;
    width: 100%;
    background: linear-gradient(to bottom, rgb(82, 207, 235) 0%, rgb(66, 162, 188)
100%);
    box-shadow: inset 0px 1px 5px rgba(255, 255, 255, 0.2);
  }
</style>
```

(g)用通配符加上 `box-sizing: border-box`。把这两个框通过修改高度宽度、padding、border-radius和立体感的厚度变为下图：



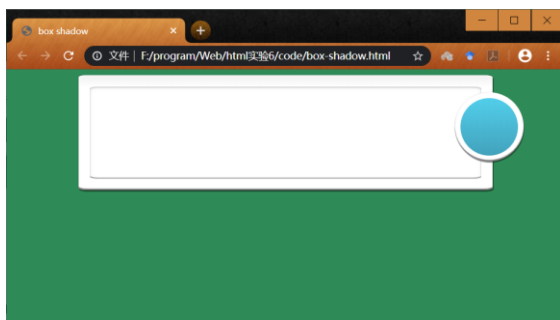
完成后截屏：



(h) 把 box1 改为相对定位，并把 box3 放到 box1 里面去，并得到下图：



完成后截屏：

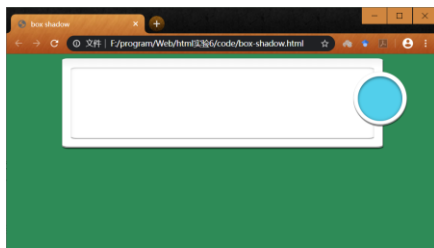


在 (6) 的基础上增加的样式：

- (i) 在鼠标悬浮在 box4 上时背景颜色在 0.3 秒内用 ease-out 过渡到 `rgb(82, 207, 235)`（原来是渐变色），`cursor` 取值 `pointer`，此时按钮会闪一下，见 `shadow-box.mp4`
- (j) 当点击 box4 时，增加两个嵌入阴影：`inset 0px 0px 5px rgba(0, 0, 0, 0.3)`，`inset 0px 3px 4px rgba(0, 0, 0, 0.3)`。目的是让按钮上部出现黑色阴影，有按键的感觉，见 `shadow-box.mp4`



完成后截图：



样式(box-shadow.html):

```
<style>
  * {
    box-sizing: border-box;
  }
```



```
body {
    background-color: seagreen;
}

html {
    font-size: 16px;
}

#box1 {
    width: 36rem;
    height: 10rem;
    margin: 0 auto;
    padding: 1rem;
    background: linear-gradient(to bottom, rgb(240, 240, 240) 0%, white 10%, white
100%);
    box-shadow: inset 0px -2px 1px rgba(0, 0, 0, 0.5), 1px 2px 1px rgba(0, 0, 0,
0.5), inset 1px 1px 2px white, 0px 0px 1px black;
    border-radius: 2%;
    position: relative;
}

#box2 {
    height: 100%;
    width: 100%;
    background: white;
    box-shadow: inset 0px 2px 1px rgba(0, 0, 0, 0.1), inset 0px -1px 1px rgba(0,
0, 0, 0.5), inset 0px 1px 5px rgba(0, 0, 0, 0.1);
    border-radius: 2%;
}

#box3 {
    width: 6rem;
    height: 6rem;
    margin: 0 auto;
    padding: 0.5rem;
    background: white;
    box-shadow: inset 0px -2px 1px rgba(0, 0, 0, 0.5), 1px 2px 1px rgba(0, 0, 0,
0.5), inset 1px 1px 2px white, 0px 0px 1px black;
    border-radius: 50%;
    position: absolute;
    top: 2rem;
    right: 3.5rem;
    z-index: 2;
}

#box4 {
    height: 100%;
    width: 100%;
    background: linear-gradient(to bottom, rgb(82, 207, 235), rgb(66, 162, 188));
    box-shadow: inset 0px 1px 5px rgba(255, 255, 255, 0.2);
    background-color: white;
```



```
border-radius: 50%;  
transition-property: background-color;  
transition-duration: 0.3s;  
transition-timing-function: ease-out;  
}  
#box4:hover {  
    background: rgb(82, 207, 235);  
    cursor: pointer;  
}  
#box4:active {  
    box-shadow: inset 0px 0px 5px rgba(0, 0, 0, 0.3), inset 0px 3px 4px rgba(0,  
0, 0, 0.3);  
}  
</style>
```

【完成情况】

是否完成了下面实验题目? (√完成 ×未做, r%完成率)

1 [√] 2 [√] 3 [√] 4 [√] 5 [√] 6 [√] 7 [√] 8 [√] 9 [√]

【交实验报告】

(a) 每位同学单独完成本实验内容并填写实验报告。

(b) 截止时间: 2020 年 6 月 20 日 23:00 (周六)

上交作业网站: <http://103.26.79.35/netdisk/default.aspx?vm=17web>

文件夹: /实验上交/html6

上传文件: (1) 学号_姓名_htm6.doc (实验报告) (不要交 PDF 文件)

(2) 学号_姓名_html6.rar (包含完成最后步骤的 html 及其相关文件)