

# “点亮你的智商”网页小游戏设计说明书

设计者：陈凯鑫

## 作品内容简介

本作品基于 HTML、JavaScript、CSS 等技术在网页上编写了一个名为“点亮你的智商”的益智小游戏，该游戏的目的是让玩家用最少的点击次数来点亮所有的灯泡。在该游戏中，玩家可用鼠标左键每次单击一个灯泡，单击后该灯泡以及其相邻位置的灯泡将会改变亮灭状态（由亮到灭，由灭到亮，以此循环）。游戏具有三种难度设置以及对游戏步数的统计，画面简洁美观，玩法浅显易懂，非常适合青少年玩家进行休闲娱乐。

## 1 设计方案

### 1.1 设计原理

团队在充分了解和深入分析当下青少年人群对网页小游戏的需求的前提下，制作出了一款基于 HTML、JavaScript、CSS 等技术的网页小游戏——“点亮你的智商”。

该游戏界面简洁清晰，操作简单。运用了 HTML 对网页进行编写，同时利用了 CSS3 技术对页面的布局、字体、颜色、背景和其它效果实现有效而精确的控制，相较于同类型的其他游戏，本作品画面以及布局更为美观大方，符合当下青少年的审美趋向，其次还设计了多种难度供玩家挑战。

### 1.2 HTML 部分

HTML 英语意思是 Hypertext Marked Language，即超文本标记语言，是一种用来制作超文本文档的简单标记语言。HTML 是由 WEB 的发明者 Tim Berners-Lee 和同事 Daniel Connolly 于 1990 年创立的一种标记式语言。它是标准通用化标记语言 SGML 的应用。

HTML（超文件标记语言）是一种建立网页文件的语言，透过标记式的指令(Tag)，将影像、声音、图片、文字、动画、影视等内容显示出来。因为它可以从一个文件跳转到另一个文件，与世界各地主机的文件连接。超文本传输协议规定了浏览器在运行 HTML 文档时所遵循的规则和进行的操作。HTTP 协议的制定使浏览器在运行超文本时有了统一的规则 and 标准。

所谓超文本，是因为它可以加入图片、声音、动画、影视等内容，事实上每一个 HTML 文档都是一种静态的网页文件，这个文件里面包含了 HTML 指令代码，这些指令代码并不是一种程序语言，它只是一种排版网页中资料显示位置的标记结构语言，易学易懂，非常简单。HTML 的普遍应用就是带来了超文本的技术——通过单击鼠标从一个主题跳转到另一个主题，从一个页面跳转到另一个页面与世界各地主机的文件链接。直接获取相关的主题。

本实例的 HTML 部分主要有以下三大部分组成：主界面、等级界面、游戏说明

主界面：步数统计、游戏界面、重新开始按钮、结束界面（一张表情包、评价语、再来一盘按钮）。

等级界面：三个等级按钮。

游戏说明：游戏说明按钮、关闭按钮、游戏说明文字及图片。

具体界面如下图 1、图 2 所示。



图 1 游戏主界面组成



图 2 游戏说明界面

游戏通关时界面如下图 3 所示。



图 3 游戏通关界面显示

### 1.3 CSS3 部分

CSS 即层叠样式表（Cascading Style Sheet）。在网页制作时采用层叠样式表技术，可以有效地对页面的布局、字体、颜色、背景和其它效果实现更加精确的控制。只要对相应的代码做一些简单的修改，就可以改变同一页面的不同部分，或者页数不同的网页的外观和格式。CSS3 是 CSS 技术的升级版本，CSS3 语言开发是朝着模块化发展的。以前的规范作为一个模块实在是太庞大而且比较复杂，所以，把它分解为一些小的模块，更多新的模块也被加入进来。这些模块包括：盒子模型、列表模块、超链接方式、语言模块、背景和边框、文字特效、多栏布局等。

该作品的按钮使用了 Bootstrap 框架，其中的主要代码：`<button class="btn btn-lg" id="ClickMe">游戏说明</button>`

此外，本实例中的 CSS3 里设计的主要样式为以下几种。

#### 1.3.1 3D 按钮样式

利用：active 伪类实现。

:active 伪类：向激活（在鼠标点击与释放之间发生的事件）的元素添加特殊的样式。

transition 属性：注意兼容设置。用法如图 4 所示。

语法

```
transition: property duration timing-function delay;
```

值	描述
<a href="#">transition-property</a>	规定设置过渡效果的 CSS 属性的名称。
<a href="#">transition-duration</a>	规定完成过渡效果需要多少秒或毫秒。
<a href="#">transition-timing-function</a>	规定速度效果的速度曲线。
<a href="#">transition-delay</a>	定义过渡效果何时开始。

图 4 transition 属性使用方法

鼠标未点击按钮前：按钮样式如图 5 所示，浅色为主，深色为阴影。



图 5 “重新开始”按钮样式 1

鼠标点击后，释放前：按钮样式如图 6 所示，无深色阴影。



图 6 “重新开始”按钮样式 2

box-shadow 属性：注意兼容设置。用法如图 7 所示。

#### 语法

```
box-shadow: h-shadow v-shadow blur spread color inset;
```

**注释：**box-shadow 向框添加一个或多个阴影。该属性是由逗号分隔的阴影列表，每个阴影由 2-4 个长度值、可选的颜色值以及可选的 inset 关键词来规定。省略长度的值是 0。

值	描述	测试
<i>h-shadow</i>	必需。水平阴影的位置。允许负值。	<a href="#">测试</a>
<i>v-shadow</i>	必需。垂直阴影的位置。允许负值。	<a href="#">测试</a>
<i>blur</i>	可选。模糊距离。	<a href="#">测试</a>
<i>spread</i>	可选。阴影的尺寸。	<a href="#">测试</a>
<i>color</i>	可选。阴影的颜色。请参阅 CSS 颜色值。	<a href="#">测试</a>
<i>inset</i>	可选。将外部阴影 (outset) 改为内部阴影。	<a href="#">测试</a>

图 7 box-shadow 属性使用方法

### 1.3.2 界面边框样式

伪类元素：before, :after

伪类元素：层叠样式表 (CSS) 的主要目的是给 HTML 元素添加样式，然而，在一些案例中给文档添加额外的元素是多余的或是不可能的。事实上 CSS 中有一个特性允许我们添加额外元素而不扰乱文档本身，这就是“伪元素”。

伪元素实际上在 CSS1 中就存在了，但是我们现在所讨论的:before 和:after 则发布于 CSS2.1 中。在最初，伪元素的语法是使用“:”（一个冒号），随着 web 的发展，在 CSS3 中修订后的伪元素使用“::”（两个冒号），也就是::before 和 ::after—以区分伪元素和伪类（比如:hover, :active 等）。

简而言之，伪元素将会在内容元素的前后插入额外的元素，因此当我们添加它们时，使用以下的标记方式，他们在技术上是平等的。但是这些元素实际上并不在文档中生成。它们将在外部可见，但是你将不会在文档的源代码中找到它们，因此，实际上它们是“虚假”的元素。

## 1.4 JavaScript 部分

JavaScript 语言的前身叫做 Live Script。自从 Sun 公司推出著名的 Java 语言之后，Netscape 公司引进了 Sun 公司有关 Java 的程序概念，将自己原有的 Live Script 重新进行设计，并改名为 JavaScript。JavaScript 是一种基于对象和事件驱动并具有安全性能的脚本语言，有了 JavaScript，可使网页变得生动。使用它的目的是与 HTML 超文本标识语言、Java 脚本语言一起实现在一个网页中链接多个对象，与网络客户交互作用，从而可以开发客户端的应用程序。它是通过嵌入或调入在标准的 HTML 语言中实现的。

本实例中涉及到的主要变量和函数如下所示。

### 1.4.1 变量

- 1、状态数组：定义一个  $\text{Size} \times \text{Size}$  大小的数组存储灯泡的状态，数组由 0，1 组成。
- 2、图片数组（两个）：我们只需显示两张图片，亮和灭，因此数组大小为 2。状态数组中的值即图片数组的索引值。图片数组存储亮和灭的图片地址。
- 3、IsOver 判断游戏是否结束：初始化后设置为 false，每次点击图片判断 IsOver，游戏结束设置为 true。
- 4、IsRepeat 判断状态数组是否重复：初始化设置为 false，没有初始化设置为 true。

### 1.4.2 函数

- 1、绘制游戏界面：即将 HTML 中的 table 标签补充完整（加入 tr、td），需要 Size 个 tr，Size 个 td，采取嵌套循环结合数组（存放插入内容）的方式。
- 2、初始化：状态数组清零->随机点亮->步数清零并更新->更新整个游戏界面->隐藏结束界面。
- 3、点击图片改变状态：包括自己的状态，和上下左右图片的状态，改变方式：先将状态数组的 0->1, 1->0(改变状态函数)，再将状态数组的值作为图片数组的索引(更新图片函数)，每点击一次要判断游戏是否结束。
- 4、改变状态：1-原状态值，进行置反。
- 5、更新图片：状态数组元素值 (0,1) -> 图片数组索引值->改变标签<img>的图片地址。
- 6、更新整个游戏界面：将随机生成的状态数组->图片数组。
- 7、判断游戏是否结束：先判断：计算状态矩阵中 1 个数量；设置结束界面：步数+评价，显示容器。
- 8、等级设置：点击不同按钮，设置 Size 的大小，再重新绘制游戏界面、初始化。
- 9、游戏说明弹窗：点击游戏说明按钮：游戏说明淡入、背景阴影；点击关闭按钮和背景：游戏说明隐藏、背景隐藏。

## 3 方案测试

### 3.1 Debug

问题描述：在游戏的 V1.0 版本中 4X4 方格以上大小的游戏无法正确结束。

解决：设置 IsOver 和 IsRepeat 参数。

### 3.2 兼容性测试

在游戏的设计过程中为了更好地兼容不同浏览器，使游戏界面在不同浏览器中也能够更

好地显示，我们在 style.css 文件中做了兼容性的设置，例如下图 8 所示。

```
-webkit-border-radius: 60px 5px;  
-webkit-box-shadow: 0px 0px 35px rgba(0, 0, 0, 0.1) inset;  
-moz-border-radius: 60px 5px;  
-moz-box-shadow: 0px 0px 35px rgba(0, 0, 0, 0.1) inset;  
-ms-border-radius: 60px 5px;  
-ms-box-shadow: 0px 0px 35px rgba(0, 0, 0, 0.1) inset;
```

图 8 游戏浏览器兼容性设定