

全部课程 (/courses/) / 基于 JavaScript 实现打地鼠游戏 (/courses/48) / 基于 JavaScript 实现打地鼠小游戏

在线实验，请到PC端体验

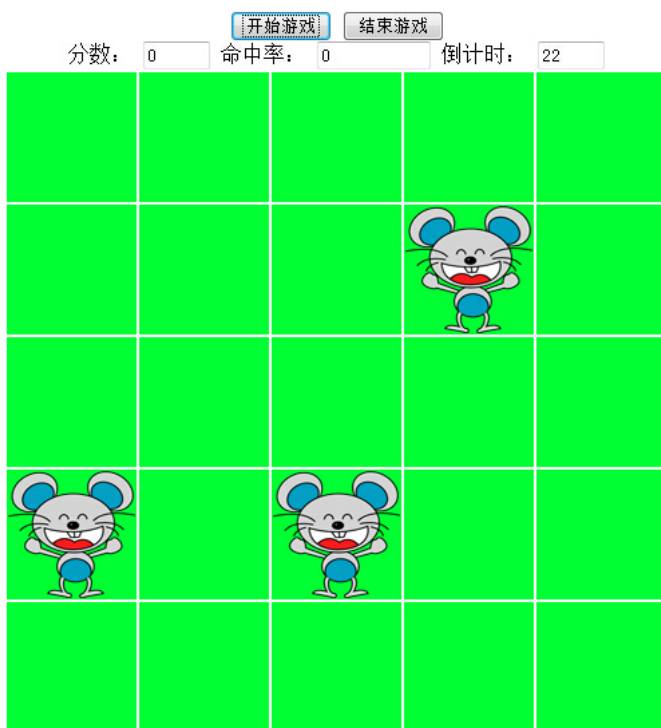
# 基于 JavaScript 实现打地鼠游戏

## 一、实验介绍

### 1.1 实验内容

本课程带领大家实现一款耳熟能详的经典小游戏——打地鼠。这是一个使用 HTML，CSS，JavaScript 实现的 Web 小游戏，需要使用 Firefox 浏览器打开才能看到游戏的最终效果。

游戏最终效果截图如下：



### 1.2 实验知识点

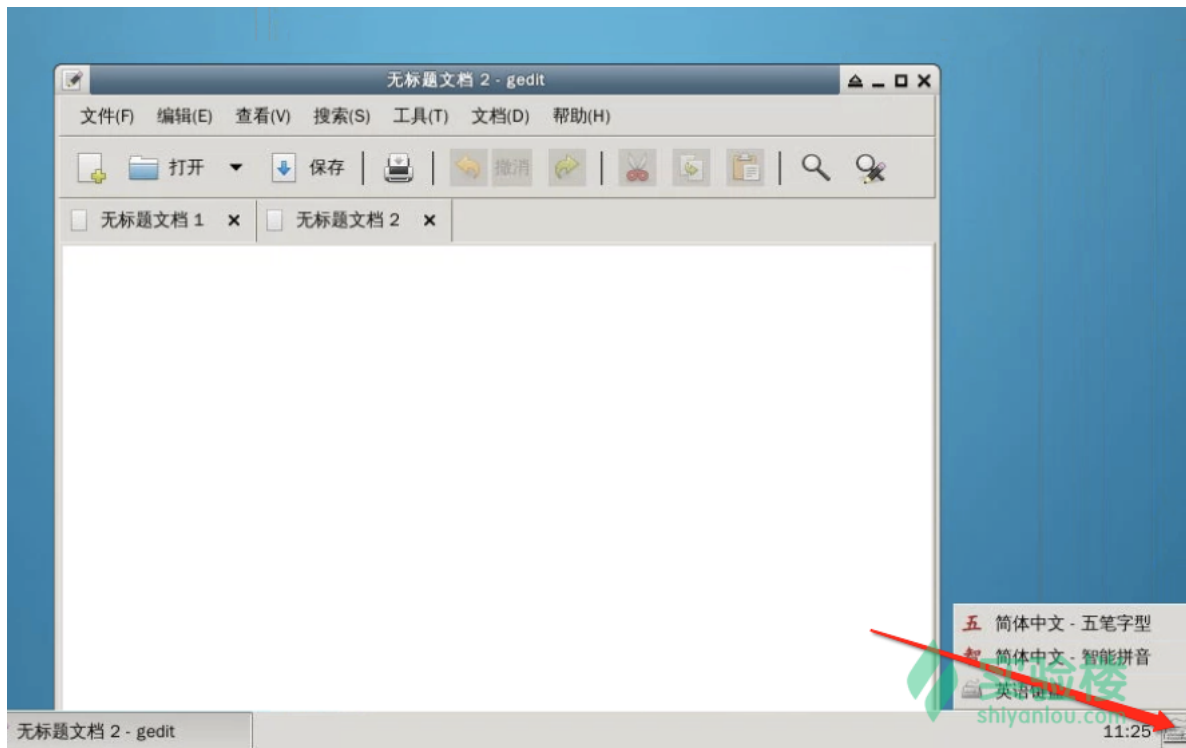
- HTML，CSS 基础
- JavaScript 计时函数，事件监听

### 1.3 实验环境

本实验环境采用 桌面环境，实验中会用到桌面上的程序：

1. Xfce 终端: Linux 命令行终端，打开后会进入 Bash 环境，可以使用 Linux 命令
2. Firefox 浏览器，可以用在需要前端界面的课程里，只需要打开 HTML 页面即可
3. gvim：非常好用的编辑器，最简单的用法可以参考课程 Vim编辑器 (<http://www.shiyanlou.com/courses/2>)
4. 其他编辑器：如果 Vim 不熟悉可以使用 gedit 或 brackets，其中 brackets 比较适合开发前端代码。

注意本实验中如果需要输入中文，可以鼠标点击右下角的键盘进行切换输入法：



## 1.4 适合人群

本实验较为简单，适合有一点前端基础（HTML 和 JavaScript）的同学，如果碰巧也懂 CSS 那就更好了。

## 1.5 代码获取

本课程中的所有源码可以通过以下方式下载：

```
$git clone https://github.com/shiyanlou/Whac-A-Mole
```

## 三、开发准备

我们先在 Code 下面创建下面的目录结构：

```
dadishu
|__index.html
|__main.js
|__img
```

img 是一个目录，我们需要在里面存放实验中要用到的老鼠和锤子的图片。进入 img 目录，通过下面的命令下载图片：

```
$wget http://labfile.oss.aliyuncs.com/courses/48/mouse.png
```

```
$wget http://labfile.oss.aliyuncs.com/courses/48/chuizi.png
```

注意：这里需要注意的是由于实验楼环境的限制，锤子图片效果可能无法显示。但并不影响游戏项目的实现和完成。

## 四、实验步骤

### 4.1 核心方法介绍

在打地鼠小游戏中将会多次用到 JavaScript 中的计时函数：

- setTimeout()
- setInterval()

这两个函数的作用很大，不仅仅在本次的小游戏中，在很多的 JavaScript 程序中都会有它们的身影。接下来将会详细的剖析这两个函数。

#### 1. setTimeout()

用于在指定的毫秒数后调用函数或计算表达式（只执行一次，可通过创建一个函数循环重复调用 `setTimeout`，来实现重复操作）

## 用法示例

- `setTimeout("functionName()",1000);`
- `setTimeout(functionName,1000)`

动手实践是学习 IT 技术最有效的方式!

开始实验

调用一个无参的方法很简单，但当我们调用一个带有参数的函数时，问题就来了。

解决的方法就是再写一个函数，该函数返回一个不带参数的函数。

示例如下：

```
function show(name){
    alert(name + " , 你好! ");
}

function returnFun(name){
    return function(){
        show(name);
    };
}

setTimeout(returnFun("小明"),1000);
```

## 2. clearTimeout()

立即终止 `setTimeout()` 方法。

示例：

```
var timeId = setTimeout(...);
clearTimeout(timeId);
```

## 3. setInterval()

可按照指定的周期（毫秒）来调用函数或计算表达式。

与 `setTimeout()` 不同，不论调用的函数或计算表达式需要多长时间才能执行完，它都只是简单的每隔一定时间就重复执行一次那个函数或计算表达式。

## 用法示例

- `setInterval("functionName()",1000);`
- `setInterval(functionName,1000);`

## clearInterval()

立即终止 `setInterval()` 方法。

示例：

```
var interId = setInterval(...);
clearInterval(interId);
```

下一节课将会具体介绍这两个方法的运用场景，以及如何完整的实现打地鼠小游戏。

## 4.2 功能逻辑设计

功能包括：

- 得分统计
- 计算成功率
- 老鼠图片的隐藏、显示
- 判断是否点中老鼠
- 最终结果显示
- 错误提示

流程设计：

- 点击“开始游戏”按钮游戏开始，否则将提示“请点击开始游戏”字样
- 分数、命中率显示重置为“0”，倒计时开始（默认为30秒）
- 老鼠图片不断显示、隐藏，玩家可点击鼠标左键进行游戏
- 当30秒倒计时结束或者玩家主动点击“结束按钮”时，游戏结束并显示游戏结果

游戏设计简单，并未添加其他丰富的游戏设计，也没有添加动画、声音等特殊效果，其目的是想带领大家从简洁的游戏逻辑设计中清楚地体会到从设计到开发、完善的乐趣，能通过简单的几行代码也能轻松实现一个属于自己制作的小游戏。

动手实践是学习IT技术最有效的方式!

开始实验

### 4.3 框架设计

Web 游戏自然离不开 HTML。常见的打地鼠游戏都是在几个固定位置上，也就是“老鼠洞”，随机出现老鼠。本次游戏设计将会采用 `<table>` 标签来布局描述“老鼠洞”，得分、命中率等结果显示将使用 `<input>` 标签。HTML 代码（包括 CSS）如下：

```

<!DOCTYPE html>
<html>
  <head>
    <title>打地鼠</title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <style type="text/css">动手实践是学习 IT 技术最有效的方式!          开始实验
      #content {
        width: 960px;
        margin: 0 auto;
        text-align: center;
      }
      table {
        margin: 0 auto;
        cursor: url('img/chuizi.png'),auto;
      }
      td {
        width: 95px;
        height: 95px;
        background-color: #00ff33;
      }
    </style>
    <script src="main.js"></script>
  </head>
  <body>
    <div id="content">
      <input type="button" value="开始游戏" onclick="GameStart()">
      <input type="button" value="结束游戏" onclick="GameOver()">
      <form name="form1">
        <label>分数: </label>
        <input type="text" name="score" size="5">
        <label>命中率: </label>
        <input type="text" name="success" size="10">
        <label>倒计时: </label>
        <input type="text" name="remtime" size="5">
      </form>
      <table>
        <tr>
          <td id="td[0]" onclick="hit(0)"></td>
          <td id="td[1]" onclick="hit(1)"></td>
          <td id="td[2]" onclick="hit(2)"></td>
          <td id="td[3]" onclick="hit(3)"></td>
          <td id="td[4]" onclick="hit(4)"></td>
        </tr>
        <tr>
          <td id="td[5]" onclick="hit(5)"></td>
          <td id="td[6]" onclick="hit(6)"></td>
          <td id="td[7]" onclick="hit(7)"></td>
          <td id="td[8]" onclick="hit(8)"></td>
          <td id="td[9]" onclick="hit(9)"></td>
        </tr>
        <tr>
          <td id="td[10]" onclick="hit(10)"></td>
          <td id="td[11]" onclick="hit(11)"></td>
          <td id="td[12]" onclick="hit(12)"></td>
          <td id="td[13]" onclick="hit(13)"></td>
          <td id="td[14]" onclick="hit(14)"></td>
        </tr>
        <tr>
          <td id="td[15]" onclick="hit(15)"></td>
          <td id="td[16]" onclick="hit(16)"></td>
          <td id="td[17]" onclick="hit(17)"></td>
          <td id="td[18]" onclick="hit(18)"></td>
          <td id="td[19]" onclick="hit(19)"></td>
        </tr>
        <tr>
          <td id="td[20]" onclick="hit(20)"></td>
          <td id="td[21]" onclick="hit(21)"></td>
          <td id="td[22]" onclick="hit(22)"></td>
          <td id="td[23]" onclick="hit(23)"></td>
          <td id="td[24]" onclick="hit(24)"></td>
        </tr>
      </table>
    </div>
  </body>
</html>

```

```
</body>  
</html>
```

这样打地鼠小游戏的结构设计就完成了，可以保存后通过 **Firefox** 浏览器查看当前效果。

## 4.4 功能逻辑实现

动手实践是学习 IT 技术最有效的方式！

[开始实验](#)

在 `main.js` 中编写游戏的逻辑，代码中已经给出了详细的注释：

```

var td = new Array(),      //保存每个格子的地鼠
    playing = false,      //游戏是否开始
    score = 0,            //分数
    beat = 0,            //鼠标点击次数
    success = 0,         //命中率
    knock = 0,          //鼠标点击次数
    countDown = 30,     //倒计时
    interId = null,     //指定 setInterval()的变量
    timeId = null;      //指定 setTimeout()的变量

//游戏结束
function GameOver(){
    timeStop();
    playing = false;
    clearMouse();
    alert("游戏结束! \n 你获得的分数为: "+score+"\n 命中率为: "+success);
    success = 0;
    score = 0;
    knock = 0;
    beat = 0;
    countDown = 30;
}

//显示当前倒计时所剩时间
function timeShow(){
    document.form1.remtime.value = countDown;
    if(countDown == 0){
        GameOver();
        return;
    }else{
        countDown = countDown-1;
        timeId = setTimeout("timeShow()",1000);
    }
}

//主动停止所有计时
function timeStop(){
    clearInterval(interId);//clearInterval()方法返回 setInterval()方法的 id
    clearTimeout(timeId);//clearTime()方法返回 setTimeout()的 id
}

//随机循环显示老鼠图片
function show(){
    if(playing)
    {
        var current = Math.floor(Math.random()*25);
        //这里的路径也需要根据自己的实际文件路径来修改
        document.getElementById("td["+current+"]").innerHTML = '';
        //使用 setTimeout()实现3秒后隐藏老鼠图片
        setTimeout("document.getElementById('td["+current+"]').innerHTML=' ',3000);
    }
}

//清除所有老鼠图片
function clearMouse(){
    for(var i=0;i<=24;i++)
    {
        document.getElementById("td["+i+"]").innerHTML="";
    }
}

//点击事件函数, 判断是否点中老鼠
function hit(id){
    if(playing==false)
    {
        alert("请点击开始游戏");
        return;
    }
    else
    {
        beat +=1;
        if(document.getElementById("td["+id+"]").innerHTML!="")
        {
            score += 1;
            knock +=1;
            success = knock/beat;
            document.form1.success.value = success;

```

```

        document.form1.score.value = score;
        document.getElementById("td["+id+"]").innerHTML="";
    }
    else
    {
        score += -1;
        success = knock/b
        document.form1.success.value = success;
        document.form1.score.value = score;
    }
}

//游戏开始
function GameStart(){
    playing = true;
    interId = setInterval("show()",1000);
    document.form1.score.value = score;
    document.form1.success.value = success;
    timeShow();
}

```

动手实践是学习 IT 技术最有效的方式!

开始实验

现在，点击开始游戏就可以让老鼠动起来。由于环境问题，你看不到锤子的效果。

## 五、实验总结

到此，整个游戏从设计到制作就完全介绍完了。运行一下自己制作的小游戏，虽然简单，但是其中的乐趣是不言而喻的。

再看看代码，也没有很多晦涩难懂的复杂算法或逻辑，这也是 JavaScript 容易入门的原因。但是 JavaScript 是一门弱类型语言，语法很灵活，所有要真真正掌握 JavaScript 还是需要花一点功夫。

最后，游戏虽然完成了，但是代码的优化也是每一个项目所必须的。例如：在 html 代码中每个 <td> 标签都绑定了 onclick 事件对象，这样繁琐的代码显然不是我们想要的，但当我们使用 js 库，比如最受人欢迎的 jquery，我们的代码会更加简洁。像这样需要优化的代码还有很多，希望同学们能有自己的创新，不断的完善、优化这个小游戏!

感谢你的学习。

[下一节 > \(/courses/48/labs/219/document\)](/courses/48/labs/219/document)

### 课程教师



**Christian**

共发布过17门课程

Web前端资深培训讲师，大型电子商务软件产品经理、交互设计师、用户体验师

[查看老师的所有课程 > \(/teacher/20407\)](/teacher/20407)

### 前置课程

[HTML基础入门 \(/courses/19\)](/courses/19)

[Javascript基础（新版） \(/courses/21\)](/courses/21)

### 进阶课程

[网页版-别踩白块游戏 \(/courses/306\)](/courses/306)

[网页版2048 \(/courses/62\)](/courses/62)



## 动手做实验，轻松学IT

