# 对话记录

**制作点击图片后的页面，在该页面中，有一扇门，门口有一个石像镇守，移动到该石像可以将其放大，点击这个石像后，出现一个方框，方框中有一道10以内随机的计算题，用户需要在文本框中输入正确答案才能跳转到下一个页面，否则就提示答案错误**

好的，我们可以创建一个新的HTML页面，页面中有一扇门，门口有一个石像镇守。点击石像后，出现一个方框，方框中有一道10以内随机的计算题，用户需要在文本框中输入正确答案才能跳转到下一个页面，否则就提示答案错误。以下是HTML，CSS和JavaScript代码：

var statue = document.getElementById('statue');

var quiz = document.getElementById('quiz');

var question = document.getElementById('question');

var answer = document.getElementById('answer');

var submit = document.getElementById('submit');

statue.addEventListener('click', function() {

  var num1 = Math.floor(Math.random() \* 10);

  var num2 = Math.floor(Math.random() \* 10);

  question.textContent = num1 + " + " + num2 + " = ?";

  quiz.classList.remove('hidden');

});

submit.addEventListener('click', function() {

  if (parseInt(answer.value) === num1 + num2) {

    window.location.href = 'next.html';

  } else {

    alert('答案错误');

  }

});

**页面出现一个角色图片，并且需要点击三次图片才能显示出石像，每点击一次向右移动，且只能移动三次**

var character = document.getElementById('character');

var clickCount = 0;

character.addEventListener('click', function() {

  clickCount++;

  if (clickCount <= 3) {

    character.classList.add('move');

    character.style.left = clickCount \* 100 + 'px';

  }

  if (clickCount === 3) {

*// 显示石像的代码*

  }

});

**页面中有一个石像图片，点击石像后，出现一个方框，方框中有一道10以内随机的计算题，用户需要在文本框中输入正确答案，否则就提示答案错误。**

这是一个涉及到HTML、CSS和JavaScript的基础交互设计问题。以下是一个简单的实现：  
首先，我们需要创建HTML结构，包括一个图片元素和一个隐藏的方框（包含问题、输入框和提示信息）：

然后，我们可以使用CSS来美化这个页面：

最后，我们使用JavaScript来添加交互功能：

document.getElementById('statue').addEventListener('click', function() {

  var num1 = Math.floor(Math.random() \* 10);

  var num2 = Math.floor(Math.random() \* 10);

  document.getElementById('question').textContent = num1 + " + " + num2 + " = ?";

  document.getElementById('questionBox').style.display = 'block';

});

document.getElementById('submit').addEventListener('click', function() {

  var answer = document.getElementById('answer').value;

  if (parseInt(answer) === num1 + num2) {

    document.getElementById('feedback').textContent = "答案正确！";

  } else {

    document.getElementById('feedback').textContent = "答案错误！";

  }

});

**设置一个倒计时，用户只能在两分钟内填写答案，否则给与一个挑战失败的反馈**

**然后，在JavaScript中添加倒计时功能：**

var countdown;

document.getElementById('statue').addEventListener('click', function() {

  num1 = Math.floor(Math.random() \* 10);

  num2 = Math.floor(Math.random() \* 10);

  document.getElementById('question').textContent = num1 + " + " + num2 + " = ?";

  document.getElementById('questionBox').style.display = 'block';

  var timeLeft = 120;

  document.getElementById('timer').textContent = "剩余时间：" + timeLeft + "秒";

  countdown = setInterval(function() {

    timeLeft--;

    document.getElementById('timer').textContent = "剩余时间：" + timeLeft + "秒";

    if (timeLeft <= 0) {

      clearInterval(countdown);

      document.getElementById('feedback').textContent = "挑战失败！";

    }

  }, 1000);

});

document.getElementById('submit').addEventListener('click', function() {

  var answer = document.getElementById('answer').value;

  if (parseInt(answer) === num1 + num2) {

    document.getElementById('feedback').textContent = "答案正确！";

  } else {

    document.getElementById('feedback').textContent = "答案错误！";

  }

  clearInterval(countdown);

});

**用户在两分钟需要答对五题才能显示挑战成功**

首先，我们需要在全局范围内定义一个变量correctAnswers来记录答对的题目数量：

document.getElementById('submit').addEventListener('click', function() {

  var answer = document.getElementById('answer').value;

  if (parseInt(answer) === num1 + num2) {

    correctAnswers++;

    if (correctAnswers >= 5) {

      document.getElementById('feedback').textContent = "挑战成功！";

      clearInterval(countdown);

    } else {

*// Generate a new question*

      num1 = Math.floor(Math.random() \* 10);

      num2 = Math.floor(Math.random() \* 10);

      document.getElementById('question').textContent = num1 + " + " + num2 + " = ?";

    }

  } else {

    document.getElementById('feedback').textContent = "答案错误！";

  }

});

**蝴蝶随机左右移动**

要让蝴蝶随机左右移动，你可以使用JavaScript的Math.random()函数生成随机数，然后根据这个随机数来改变蝴蝶的位置。以下是一个基本的实现：  
  
JavaScript:

function moveButterfly(*butterfly*) {

  var left = Math.random() \* window.innerWidth;

  butterfly.style.left = left + 'px';

}

var butterflies = document.getElementsByClassName('butterfly');

for (var i = 0; i < butterflies.length; i++) {

  setInterval(moveButterfly, 1000, butterflies[i]);  *// 每秒移动一次*

}

# 修改思路

1. 刚开始做的网页与用户的交互不多，于是添加了一个角色移动的click事件，一是为了防止图片未刷新出来时，角色已经移动，也是为了增加用户与网页之间的交互
2. 线索提示方面，用户不知道下一步该做什么，之后在页面上给出了更清晰的提示
3. 在第二关的计算题设置倒计时，并显示答题情况，优化页面的交互功能
4. 优化第二关蝴蝶的移动，令其随机在指定方框位置移动