网页设计与制作课程设计网页小游戏任务报告

报告人：麦合木提·麦麦提

学号：21332129

班级：21级传播一班

2023年7月10日

一、游戏设计思路：该游戏的灵感干来源于钢琴块儿和打地鼠小游戏。

游戏的基本玩法逻辑：该游戏主要考验玩家的反应能力。

二、游戏流程：游戏玩家在游戏开始之前有10分，在游戏开始时，每隔3秒都有一个盒子从网页顶端掉落，盒子在网页顶端出现的位置是随机的，盒子到达网页低端所用时间为两秒，盒子掉落轨迹为水平垂直直线，盒子最终不会在网页底端停留，而会直接消失，玩家在盒子消失之前击中盒子得分，未击中盒子失分，最终玩家的分数为0分，则游戏宣告失败，玩家的分数为60分，则游戏胜利。

三、游戏操作互动方式：玩家通过鼠标来击中网页中掉落的盒子。

写代码的思路：首先我自己写好了网页的HTML、css部分，

<!DOCTYPE html>

<html>

<head>

  <title>打击盒子游戏</title>

  <style>

    #game-area {

      position: relative;

      width: 400px;

      height: 600px;

      border: 1px solid black;

      margin: 20px auto;

    }

    .box {

      position: absolute;

      width: 50px;

      height: 50px;

      background-color: red;

    }

    #start-button {

      display: block;

      margin: 0 auto;

      margin-top: 100px;

    }

    #end-button{

      margin: 20px auto;

    }

    #result-text {

      position: absolute;

      top: 50%;

      left: 50%;

      transform: translate(-50%, -50%);

      text-align: center;

      font-size: 24px;

      font-weight: bold;

      visibility: hidden;

    }

    .startMenu{

      margin: 0 auto;

      margin-top: 200px;

      padding-top: 25px;

      width: 200px;

      height: 200px;

      background-color: red;

      text-align: center;

      box-sizing: border-box;

    }

  </style>

</head>

<body>

  <div id="game-area">

    <div id="score">Score: 10</div>

  </div>

第二步：完成js部分

 <script>

    var score = 10;

    var gameArea = document.getElementById('game-area');

    var scoreElement = document.getElementById('score');

    var resultText = document.getElementById('result-text');

    var boxInterval;

让cursor帮我写一个菜单代码

function createMenu(){

      let startMenu = document.createElement("div")

      startMenu.className = "startMenu"

      gameArea.appendChild(startMenu)

      resultText = document.createElement("div")

      resultText.id = "result-text"

      gameArea.appendChild(resultText)

      let startButton = document.createElement("button")

      startButton.id = "start-button"

      startButton.innerHTML = "开始游戏"

      startButton.onclick = function(){

        startGame()

      }

      let endButton = document.createElement("button")

      endButton.id = "end-button"

      endButton.innerHTML = "结束游戏"

      endButton.onclick = function(){

        endGame1()

      }

      startMenu.appendChild(startButton)

      startMenu.appendChild(endButton)

    }

    createMenu()

    function endGame1(){

      window.close()

然后写产生盒子的代码、盒子的速度

function createBox() {

      if(score > 0){

        var box = document.createElement('div');

        box.className = 'box';

        var randomPosition = Math.floor(Math.random() \* (gameArea.offsetWidth - 50));

        box.style.left = randomPosition + 'px';

        gameArea.appendChild(box);

        var topPosition = 0;

        box.boxInterval = setInterval(function() {

          topPosition += 2;

          box.style.top = topPosition + 'px';

          if (topPosition >= gameArea.offsetHeight) {

            clearInterval(box.boxInterval);

            box.remove();

            score -= 2;

            scoreElement.textContent = 'Score: ' + score;

            if (score === 0) {

              endGame(false);

            }

          }

        }, 10);

      }

    }

  var createBoxInterval;

  function startGame() {

    var startButton = document.getElementById('start-button');

    startButton.disabled = true; // Disable the button once the game starts

    resultText.textContent = ''; // Clear any previous result text

    resultText.style.visibility = 'hidden'; // Hide the result text

    createBoxInterval = setInterval(createBox, 3000);

    var startMenu = document.querySelector('.startMenu')

    gameArea.removeChild(startMenu)

  }

最后写游戏的胜负规则

  gameArea.addEventListener('click', function(event) {

    var clickedElement = event.target;

    if (clickedElement.className === 'box') {

      clearInterval(clickedElement.boxInterval);

      clickedElement.remove();

      score += 1;

      scoreElement.textContent = 'Score: ' + score;

      if (score >= 60) {

        endGame(true);

      }

    }

  });

      function endGame(isVictory) {

        let boxs = document.querySelectorAll(".box")

        for(let i = 0; i < boxs.length; i++){

          boxs[i].remove();

        }

        if (isVictory) {

          createMenu()

          resultText.textContent = '恭喜，你赢得了游戏！';

        } else {

          clearInterval(createBoxInterval);

          createMenu()

          resultText.innerHTML = '很遗憾，你输掉了游戏！';

        }

        resultText.style.visibility = 'visible'; // Show the result text

        var startButton = document.getElementById('start-button');

        startButton.disabled = false; // Enable the start button to play again

      }

  </script>

</body>

</html>

写完这些以后发现当网页中同时出现两个盒子时游戏宣告失败，并疯狂失分？

Cursor的答案：

这个问题的原因在于boxInterval变量。在createBox()函数中，每创建一个新的盒子，都会用新的setInterval函数覆盖boxInterval。因此，当你点击一个盒子时，clearInterval(boxInterval);只会清除最新的盒子的计时器，而不是被点击的盒子的计时器。  
  
解决这个问题的一种方法是将每个盒子的计时器存储在盒子自身中，这样在点击盒子时，可以清除该盒子的计时器。以下是修改后的代码：

function createBox() {

  var box = document.createElement('div');

  box.className = 'box';

  var randomPosition = Math.floor(Math.random() \* (gameArea.offsetWidth - 50));

  box.style.left = randomPosition + 'px';

  gameArea.appendChild(box);

  var topPosition = 0;

  box.boxInterval = setInterval(function() {

    topPosition += 5;

    box.style.top = topPosition + 'px';

    if (topPosition >= gameArea.offsetHeight) {

      clearInterval(box.boxInterval);

      box.remove();

      score -= 2;

      scoreElement.textContent = 'Score: ' + score;

      if (score <= 0) {

        endGame(false);

      }

    }

  }, 10);

}

gameArea.addEventListener('click', function(*event*) {

  var clickedElement = event.target;

  if (clickedElement.className === 'box') {

    clearInterval(clickedElement.boxInterval);

    clickedElement.remove();

    score += 1;

    scoreElement.textContent = 'Score: ' + score;

    if (score >= 60) {

      endGame(true);

    }

  }

});

这样，每个盒子都有自己的计时器，点击盒子时会清除该盒子的计时器，不会影响其他盒子。

修改前的代码：

|  |
| --- |
| function createBox() { |
|  | var box = document.createElement('div'); |
|  | box.className = 'box'; |
|  | var randomPosition = Math.floor(Math.random() \* (gameArea.offsetWidth - 50)); |
|  | box.style.left = randomPosition + 'px'; |
|  | gameArea.appendChild(box); |
|  |  |
|  | var topPosition = 0; |
|  | boxInterval = setInterval(function() { |
|  | topPosition += 2; |
|  | box.style.top = topPosition + 'px'; |
|  |  |
|  | if (topPosition >= gameArea.offsetHeight) { |
|  | clearInterval(boxInterval); |
|  | box.remove(); |
|  | score -= 2; |
|  | scoreElement.textContent = 'Score: ' + score; |
|  | if (score <= 0) { |
|  | endGame(false); |
|  | } |
|  | } |
|  | }, 10); |
|  | } |
|  |  |
|  | function startGame() { |
|  | var startButton = document.getElementById('start-button'); |
|  | startButton.disabled = true; // Disable the button once the game starts |
|  | resultText.textContent = ''; // Clear any previous result text |
|  | resultText.style.visibility = 'hidden'; // Hide the result text |
|  | setInterval(createBox, 2000); |
|  | } |
|  |  |
|  | gameArea.addEventListener('click', function(event) { |
|  | var clickedElement = event.target; |
|  | if (clickedElement.className === 'box') { |
|  | clearInterval(boxInterval); |
|  | clickedElement.remove(); |
|  | score += 1; |
|  | scoreElement.textContent = 'Score: ' + score; |
|  | if (score >= 60) { |
|  | endGame(true); |
|  | } |
|  | } |
|  | }); |