|  |
| --- |
| let isDesktop = navigator['userAgent'].match(/(ipad|iphone|ipod|android|windows phone)/i) ? false : true; |
|  |  | let fontunit = isDesktop ? 20 : ((window.innerWidth > window.innerHeight ? window.innerHeight : window.innerWidth) / 320) \* 10; |
|  |  | document.write('<style type="text/css">' + |
|  |  | 'html,body {font-size:' + (fontunit < 30 ? fontunit : '30') + 'px;}' + |
|  |  | (isDesktop ? '#welcome,#GameTimeLayer,#GameLayerBG,#GameScoreLayer.SHADE{position: absolute;}' : |
|  |  | '#welcome,#GameTimeLayer,#GameLayerBG,#GameScoreLayer.SHADE{position:fixed;}@media screen and (orientation:landscape) {#landscape {display: box; display: -webkit-box; display: -moz-box; display: -ms-flexbox;}}') + |
|  |  | '</style>'); |
|  |  | let map = {'d': 1, 'f': 2, 'j': 3, 'k': 4}; |
|  |  | if (isDesktop){ |
|  |  | document.write('<div id="gameBody">'); |
|  |  | document.onkeydown = function (e) { |
|  |  | let key = e.key.toLowerCase(); |
|  |  | if (Object.keys(map).indexOf(key) !== -1) { |
|  |  | click(map[key]) |
|  |  | (function(w) { |
|  |  | let isDesktop = !navigator['userAgent'].match(/(ipad|iphone|ipod|android|windows phone)/i); |
|  |  | let fontunit = isDesktop ? 20 : ((window.innerWidth > window.innerHeight ? window.innerHeight : window.innerWidth) / 320) \* 10; |
|  |  | document.write('<style type="text/css">' + |
|  |  | 'html,body {font-size:' + (fontunit < 30 ? fontunit : '30') + 'px;}' + |
|  |  | (isDesktop ? '#welcome,#GameTimeLayer,#GameLayerBG,#GameScoreLayer.SHADE{position: absolute;}' : |
|  |  | '#welcome,#GameTimeLayer,#GameLayerBG,#GameScoreLayer.SHADE{position:fixed;}@media screen and (orientation:landscape) {#landscape {display: box; display: -webkit-box; display: -moz-box; display: -ms-flexbox;}}') + |
|  |  | '</style>'); |
|  |  | let map = {'d': 1, 'f': 2, 'j': 3, 'k': 4}; |
|  |  | if (isDesktop) { |
|  |  | document.write('<div id="gameBody">'); |
|  |  | document.onkeydown = function (e) { |
|  |  | let key = e.key.toLowerCase(); |
|  |  | if (Object.keys(map).indexOf(key) !== -1) { |
|  |  | click(map[key]) |
|  |  | } |
|  |  | } |
|  |  | } |
|  |  | let body, blockSize, GameLayer = [], |
|  |  | GameLayerBG, touchArea = [], |
|  |  | GameTimeLayer; |
|  |  | let transform, transitionDuration; |
|  |  |  |
|  |  | w.init = function() { |
|  |  | showWelcomeLayer(); |
|  |  | body = document.getElementById('gameBody') || document.body; |
|  |  | body.style.height = window.innerHeight + 'px'; |
|  |  | transform = typeof (body.style.webkitTransform) != 'undefined' ? 'webkitTransform' : (typeof (body.style.msTransform) != |
|  |  | 'undefined' ? 'msTransform' : 'transform'); |
|  |  | transitionDuration = transform.replace(/ransform/g, 'ransitionDuration'); |
|  |  | GameTimeLayer = document.getElementById('GameTimeLayer'); |
|  |  | GameLayer.push(document.getElementById('GameLayer1')); |
|  |  | GameLayer[0].children = GameLayer[0].querySelectorAll('div'); |
|  |  | GameLayer.push(document.getElementById('GameLayer2')); |
|  |  | GameLayer[1].children = GameLayer[1].querySelectorAll('div'); |
|  |  | GameLayerBG = document.getElementById('GameLayerBG'); |
|  |  | if (GameLayerBG.ontouchstart === null) { |
|  |  | GameLayerBG.ontouchstart = gameTapEvent; |
|  |  | } else { |
|  |  | GameLayerBG.onmousedown = gameTapEvent; |
|  |  | } |
|  |  | gameInit(); |
|  |  | initSetting(); |
|  |  | window.addEventListener('resize', refreshSize, false); |
|  |  | let btn = document.getElementById('ready-btn'); |
|  |  | btn.className = 'btn btn-primary btn-lg'; |
|  |  | btn.onclick = function () { |
|  |  | closeWelcomeLayer(); |
|  |  | } |
|  |  | } |
|  |  |  |
|  |  | w.winOpen = function() { |
|  |  | window.open(location.href + '?r=' + Math.random(), 'nWin', 'height=500,width=320,toolbar=no,menubar=no,scrollbars=no'); |
|  |  | let opened = window.open('about:blank', '\_self'); |
|  |  | opened.opener = null; |
|  |  | opened.close(); |
|  |  | } |
|  |  | } |
|  |  | let body, blockSize, GameLayer = [], |
|  |  | GameLayerBG, touchArea = [], |
|  |  | GameTimeLayer; |
|  |  | let transform, transitionDuration; |
|  |  |  |
|  |  | function init() { |
|  |  | showWelcomeLayer(); |
|  |  | body = document.getElementById('gameBody') || document.body; |
|  |  | body.style.height = window.innerHeight + 'px'; |
|  |  | transform = typeof (body.style.webkitTransform) != 'undefined' ? 'webkitTransform' : (typeof (body.style.msTransform) != |
|  |  | 'undefined' ? 'msTransform' : 'transform'); |
|  |  | transitionDuration = transform.replace(/ransform/g, 'ransitionDuration'); |
|  |  | GameTimeLayer = document.getElementById('GameTimeLayer'); |
|  |  | GameLayer.push(document.getElementById('GameLayer1')); |
|  |  | GameLayer[0].children = GameLayer[0].querySelectorAll('div'); |
|  |  | GameLayer.push(document.getElementById('GameLayer2')); |
|  |  | GameLayer[1].children = GameLayer[1].querySelectorAll('div'); |
|  |  | GameLayerBG = document.getElementById('GameLayerBG'); |
|  |  | if (GameLayerBG.ontouchstart === null) { |
|  |  | GameLayerBG.ontouchstart = gameTapEvent; |
|  |  | } else { |
|  |  | GameLayerBG.onmousedown = gameTapEvent; |
|  |  | } |
|  |  | gameInit(); |
|  |  | initSetting(); |
|  |  | window.addEventListener('resize', refreshSize, false); |
|  |  | let btn = document.getElementById('ready-btn'); |
|  |  | btn.className = 'btn btn-primary btn-lg'; |
|  |  | btn.onclick = function () { |
|  |  | closeWelcomeLayer(); |
|  |  | } |
|  |  | } |
|  |  |  |
|  |  | function winOpen() { |
|  |  | window.open(location.href + '?r=' + Math.random(), 'nWin', 'height=500,width=320,toolbar=no,menubar=no,scrollbars=no'); |
|  |  | let opened = window.open('about:blank', '\_self'); |
|  |  | opened.opener = null; |
|  |  | opened.close(); |
|  |  | } |
|  |  | let refreshSizeTime; |
|  |  |  |
|  |  | function refreshSize() { |
|  |  | clearTimeout(refreshSizeTime); |
|  |  | refreshSizeTime = setTimeout(\_refreshSize, 200); |
|  |  | } |
|  |  |  |
|  |  | function \_refreshSize() { |
|  |  | countBlockSize(); |
|  |  | for (let i = 0; i < GameLayer.length; i++) { |
|  |  | let box = GameLayer[i]; |
|  |  |  |
|  |  | let refreshSizeTime; |
|  |  |  |
|  |  | w.refreshSize = function() { |
|  |  | clearTimeout(refreshSizeTime); |
|  |  | refreshSizeTime = setTimeout(\_refreshSize, 200); |
|  |  | } |
|  |  |  |
|  |  | w.\_refreshSize = function() { |
|  |  | countBlockSize(); |
|  |  | for (let i = 0; i < GameLayer.length; i++) { |
|  |  | let box = GameLayer[i]; |
|  |  | for (let j = 0; j < box.children.length; j++) { |
|  |  | let r = box.children[j], |
|  |  | rstyle = r.style; |
|  |  | rstyle.left = (j % 4) \* blockSize + 'px'; |
|  |  | rstyle.bottom = Math.floor(j / 4) \* blockSize + 'px'; |
|  |  | rstyle.width = blockSize + 'px'; |
|  |  | rstyle.height = blockSize + 'px'; |
|  |  | } |
|  |  | } |
|  |  | let f, a; |
|  |  | if (GameLayer[0].y > GameLayer[1].y) { |
|  |  | f = GameLayer[0]; |
|  |  | a = GameLayer[1]; |
|  |  | } else { |
|  |  | f = GameLayer[1]; |
|  |  | a = GameLayer[0]; |
|  |  | } |
|  |  | let y = ((\_gameBBListIndex) % 10) \* blockSize; |
|  |  | f.y = y; |
|  |  | f.style[transform] = 'translate3D(0,' + f.y + 'px,0)'; |
|  |  | a.y = -blockSize \* Math.floor(f.children.length / 4) + y; |
|  |  | a.style[transform] = 'translate3D(0,' + a.y + 'px,0)'; |
|  |  | } |
|  |  |  |
|  |  | w.countBlockSize = function() { |
|  |  | blockSize = body.offsetWidth / 4; |
|  |  | body.style.height = window.innerHeight + 'px'; |
|  |  | GameLayerBG.style.height = window.innerHeight + 'px'; |
|  |  | touchArea[0] = window.innerHeight; |
|  |  | touchArea[1] = window.innerHeight - blockSize \* 3; |
|  |  | } |
|  |  |  |
|  |  | let \_gameBBList = [], |
|  |  | \_gameBBListIndex = 0, |
|  |  | \_gameOver = false, |
|  |  | \_gameStart = false, |
|  |  | \_gameTime, \_gameTimeNum, \_gameScore, \_date1, deviation\_time; |
|  |  |  |
|  |  | w.gameInit = function() { |
|  |  | createjs.Sound.registerSound({ |
|  |  | src: "./static/music/err.mp3", |
|  |  | id: "err" |
|  |  | }); |
|  |  | createjs.Sound.registerSound({ |
|  |  | src: "./static/music/end.mp3", |
|  |  | id: "end" |
|  |  | }); |
|  |  | createjs.Sound.registerSound({ |
|  |  | src: "./static/music/tap.mp3", |
|  |  | id: "tap" |
|  |  | }); |
|  |  | gameRestart(); |
|  |  | } |
|  |  |  |
|  |  | w.gameRestart = function() { |
|  |  | \_gameBBList = []; |
|  |  | \_gameBBListIndex = 0; |
|  |  | \_gameScore = 0; |
|  |  | \_gameOver = false; |
|  |  | \_gameStart = false; |
|  |  | \_gameTimeNum = 20; |
|  |  | GameTimeLayer.innerHTML = creatTimeText(\_gameTimeNum); |
|  |  | countBlockSize(); |
|  |  | refreshGameLayer(GameLayer[0]); |
|  |  | refreshGameLayer(GameLayer[1], 1); |
|  |  | } |
|  |  |  |
|  |  | w.gameStart = function() { |
|  |  | \_date1 = new Date(); |
|  |  | \_gameStart = true; |
|  |  | \_gameTime = setInterval(gameTime, 1000); |
|  |  | } |
|  |  |  |
|  |  | w.gameOver = function() { |
|  |  | \_gameOver = true; |
|  |  | clearInterval(\_gameTime); |
|  |  | setTimeout(function () { |
|  |  | GameLayerBG.className = ''; |
|  |  | showGameScoreLayer(); |
|  |  | }, 1500); |
|  |  | } |
|  |  |  |
|  |  |  |
|  |  | w.encrypt = function(text) { |
|  |  | let encrypt = new JSEncrypt(); |
|  |  | encrypt.setPublicKey("MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDTzGwX6FVKc7rDiyF3H+jKpBlRCV4jOiJ4JR33qZPVXx8ahW6brdBF9H1vdHBAyO6AeYBumKIyunXP9xzvs1qJdRNhNoVwHCwGDu7TA+U4M7G9FArDG0Y6k4LbS0Ks9zeRBMiWkW53yQlPshhtOxXCuZZOMLqk1vEvTCODYYqX5QIDAQAB"); |
|  |  | return encrypt.encrypt(text); |
|  |  | } |
|  |  |  |
|  |  | w.SubmitResults = function() { |
|  |  | let system = "其他操作系统"; |
|  |  | let area = "异世界"; |
|  |  | if (document.getElementById("username").value) { |
|  |  | if (navigator.appVersion.indexOf("Win") !== -1) system = "Windows"; |
|  |  | if (navigator.appVersion.indexOf("Mac") !== -1) system = "Macintosh"; |
|  |  | if (navigator.appVersion.indexOf("Linux") !== -1) system = "Linux"; |
|  |  | if (navigator.appVersion.indexOf("Android") !== -1) system = "Android"; |
|  |  | if (navigator.appVersion.indexOf("like Mac") !== -1) system = "iOS"; |
|  |  | if (returnCitySN['cname']) { |
|  |  | area = returnCitySN['cname'] |
|  |  | } |
|  |  | let httpRequest = new XMLHttpRequest(); |
|  |  | httpRequest.open('POST', './SubmitResults.php', true); |
|  |  | httpRequest.setRequestHeader("Content-type", "application/json"); |
|  |  | let name = document.getElementById("username").value; |
|  |  | let message = document.getElementById("message").value; |
|  |  | let test = "|\_|"; |
|  |  | httpRequest.send(encrypt(\_gameScore + test + name + test + tj + test + system + test + area + test + message)); |
|  |  | } |
|  |  | } |
|  |  |  |
|  |  | w.gameTime = function() { |
|  |  | \_gameTimeNum--; |
|  |  | if (\_gameTimeNum <= 0) { |
|  |  | GameTimeLayer.innerHTML = '&nbsp;&nbsp;&nbsp;&nbsp;时间到！'; |
|  |  | gameOver(); |
|  |  | GameLayerBG.className += ' flash'; |
|  |  | createjs.Sound.play("end"); |
|  |  | } else { |
|  |  | GameTimeLayer.innerHTML = creatTimeText(\_gameTimeNum); |
|  |  | } |
|  |  | } |
|  |  |  |
|  |  | w.creatTimeText = function(n) { |
|  |  | return '&nbsp;TIME:' + n; |
|  |  | } |
|  |  |  |
|  |  | let \_ttreg = / t{1,2}(\d+)/, |
|  |  | \_clearttClsReg = / t{1,2}\d+| bad/; |
|  |  |  |
|  |  | w.refreshGameLayer = function(box, loop, offset) { |
|  |  | let i = Math.floor(Math.random() \* 1000) % 4 + (loop ? 0 : 4); |
|  |  | for (let j = 0; j < box.children.length; j++) { |
|  |  | let r = box.children[j], |
|  |  | rstyle = r.style; |
|  |  | rstyle.left = (j % 4) \* blockSize + 'px'; |
|  |  | rstyle.bottom = Math.floor(j / 4) \* blockSize + 'px'; |
|  |  | rstyle.width = blockSize + 'px'; |
|  |  | rstyle.height = blockSize + 'px'; |
|  |  | r.className = r.className.replace(\_clearttClsReg, ''); |
|  |  | if (i === j) { |
|  |  | \_gameBBList.push({ |
|  |  | cell: i % 4, |
|  |  | id: r.id |
|  |  | }); |
|  |  | r.className += ' t' + (Math.floor(Math.random() \* 1000) % 5 + 1); |
|  |  | r.notEmpty = true; |
|  |  | i = (Math.floor(j / 4) + 1) \* 4 + Math.floor(Math.random() \* 1000) % 4; |
|  |  | } else { |
|  |  | r.notEmpty = false; |
|  |  | } |
|  |  | } |
|  |  | } |
|  |  | let f, a; |
|  |  | if (GameLayer[0].y > GameLayer[1].y) { |
|  |  | f = GameLayer[0]; |
|  |  | a = GameLayer[1]; |
|  |  | } else { |
|  |  | f = GameLayer[1]; |
|  |  | a = GameLayer[0]; |
|  |  | } |
|  |  | let y = ((\_gameBBListIndex) % 10) \* blockSize; |
|  |  | f.y = y; |
|  |  | f.style[transform] = 'translate3D(0,' + f.y + 'px,0)'; |
|  |  | a.y = -blockSize \* Math.floor(f.children.length / 4) + y; |
|  |  | a.style[transform] = 'translate3D(0,' + a.y + 'px,0)'; |
|  |  | } |
|  |  |  |
|  |  | function countBlockSize() { |
|  |  | blockSize = body.offsetWidth / 4; |
|  |  | body.style.height = window.innerHeight + 'px'; |
|  |  | GameLayerBG.style.height = window.innerHeight + 'px'; |
|  |  | touchArea[0] = window.innerHeight - blockSize \* 0; |
|  |  | touchArea[1] = window.innerHeight - blockSize \* 3; |
|  |  | } |
|  |  | let \_gameBBList = [], |
|  |  | \_gameBBListIndex = 0, |
|  |  | \_gameOver = false, |
|  |  | \_gameStart = false, |
|  |  | \_gameTime, \_gameTimeNum, \_gameScore, \_date1, deviation\_time; |
|  |  |  |
|  |  | function gameInit() { |
|  |  | createjs.Sound.registerSound({ |
|  |  | src: "./static/music/err.mp3", |
|  |  | id: "err" |
|  |  | }); |
|  |  | createjs.Sound.registerSound({ |
|  |  | src: "./static/music/end.mp3", |
|  |  | id: "end" |
|  |  | }); |
|  |  | createjs.Sound.registerSound({ |
|  |  | src: "./static/music/tap.mp3", |
|  |  | id: "tap" |
|  |  | }); |
|  |  | gameRestart(); |
|  |  | } |
|  |  |  |
|  |  | function gameRestart() { |
|  |  | \_gameBBList = []; |
|  |  | \_gameBBListIndex = 0; |
|  |  | \_gameScore = 0; |
|  |  | \_gameOver = false; |
|  |  | \_gameStart = false; |
|  |  | \_gameTimeNum = 20; |
|  |  | GameTimeLayer.innerHTML = creatTimeText(\_gameTimeNum); |
|  |  | countBlockSize(); |
|  |  | refreshGameLayer(GameLayer[0]); |
|  |  | refreshGameLayer(GameLayer[1], 1); |
|  |  | } |
|  |  |  |
|  |  | function gameStart() { |
|  |  | \_date1 = new Date(); |
|  |  | \_gameStart = true; |
|  |  | \_gameTime = setInterval(gameTime, 1000); |
|  |  | } |
|  |  |  |
|  |  | function gameOver() { |
|  |  | \_gameOver = true; |
|  |  | clearInterval(\_gameTime); |
|  |  | setTimeout(function () { |
|  |  | GameLayerBG.className = ''; |
|  |  | showGameScoreLayer(); |
|  |  | }, 1500); |
|  |  | } |
|  |  |  |
|  |  |  |
|  |  | function encrypt(text) { |
|  |  | let encrypt = new JSEncrypt(); |
|  |  | encrypt.setPublicKey("MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDTzGwX6FVKc7rDiyF3H+jKpBlRCV4jOiJ4JR33qZPVXx8ahW6brdBF9H1vdHBAyO6AeYBumKIyunXP9xzvs1qJdRNhNoVwHCwGDu7TA+U4M7G9FArDG0Y6k4LbS0Ks9zeRBMiWkW53yQlPshhtOxXCuZZOMLqk1vEvTCODYYqX5QIDAQAB"); |
|  |  | let data = encrypt.encrypt(text); |
|  |  | return data; |
|  |  | } |
|  |  |  |
|  |  | function SubmitResults() { |
|  |  | let system = "其他操作系统"; |
|  |  | let area = "异世界"; |
|  |  | if (document.getElementById("username").value) { |
|  |  | if (navigator.appVersion.indexOf("Win") != -1) system = "Windows"; |
|  |  | if (navigator.appVersion.indexOf("Mac") != -1) system = "Macintosh"; |
|  |  | if (navigator.appVersion.indexOf("Linux") != -1) system = "Linux"; |
|  |  | if (navigator.appVersion.indexOf("Android") != -1) system = "Android"; |
|  |  | if (navigator.appVersion.indexOf("like Mac") != -1) system = "iOS"; |
|  |  | if (returnCitySN['cname']) { area = returnCitySN['cname'] }; |
|  |  | let httpRequest = new XMLHttpRequest(); |
|  |  | httpRequest.open('POST', './SubmitResults.php', true); |
|  |  | httpRequest.setRequestHeader("Content-type", "application/json"); |
|  |  | let name = document.getElementById("username").value; |
|  |  | let message = document.getElementById("message").value; |
|  |  | let test = "|\_|"; |
|  |  | httpRequest.send(encrypt(\_gameScore + test + name + test + tj + test + system + test + area + test + message)); |
|  |  | } |
|  |  | } |
|  |  |  |
|  |  | function gameTime() { |
|  |  | \_gameTimeNum--; |
|  |  | if (\_gameTimeNum <= 0) { |
|  |  | GameTimeLayer.innerHTML = '&nbsp;&nbsp;&nbsp;&nbsp;时间到！'; |
|  |  | gameOver(); |
|  |  | GameLayerBG.className += ' flash'; |
|  |  | createjs.Sound.play("end"); |
|  |  | } else { |
|  |  | GameTimeLayer.innerHTML = creatTimeText(\_gameTimeNum); |
|  |  | } |
|  |  | } |
|  |  |  |
|  |  | function creatTimeText(n) { |
|  |  | return '&nbsp;TIME:' + n; |
|  |  | } |
|  |  | let \_ttreg = / t{1,2}(\d+)/, |
|  |  | \_clearttClsReg = / t{1,2}\d+| bad/; |
|  |  |  |
|  |  | function refreshGameLayer(box, loop, offset) { |
|  |  | let i = Math.floor(Math.random() \* 1000) % 4 + (loop ? 0 : 4); |
|  |  | for (let j = 0; j < box.children.length; j++) { |
|  |  | let r = box.children[j], |
|  |  | rstyle = r.style; |
|  |  | rstyle.left = (j % 4) \* blockSize + 'px'; |
|  |  | rstyle.bottom = Math.floor(j / 4) \* blockSize + 'px'; |
|  |  | rstyle.width = blockSize + 'px'; |
|  |  | rstyle.height = blockSize + 'px'; |
|  |  | r.className = r.className.replace(\_clearttClsReg, ''); |
|  |  | if (i == j) { |
|  |  | \_gameBBList.push({ |
|  |  | cell: i % 4, |
|  |  | id: r.id |
|  |  | }); |
|  |  | r.className += ' t' + (Math.floor(Math.random() \* 1000) % 5 + 1); |
|  |  | r.notEmpty = true; |
|  |  | i = (Math.floor(j / 4) + 1) \* 4 + Math.floor(Math.random() \* 1000) % 4; |
|  |  | } else { |
|  |  | r.notEmpty = false; |
|  |  | } |
|  |  | } |
|  |  | if (loop) { |
|  |  | box.style.webkitTransitionDuration = '0ms'; |
|  |  | box.style.display = 'none'; |
|  |  | box.y = -blockSize \* (Math.floor(box.children.length / 4) + (offset || 0)) \* loop; |
|  |  | setTimeout(function () { |
|  |  | box.style[transform] = 'translate3D(0,' + box.y + 'px,0)'; |
|  |  | if (loop) { |
|  |  | box.style.webkitTransitionDuration = '0ms'; |
|  |  | box.style.display = 'none'; |
|  |  | box.y = -blockSize \* (Math.floor(box.children.length / 4) + (offset || 0)) \* loop; |
|  |  | setTimeout(function () { |
|  |  | box.style.display = 'block'; |
|  |  | }, 100); |
|  |  | }, 200); |
|  |  | } else { |
|  |  | box.y = 0; |
|  |  | box.style[transform] = 'translate3D(0,' + box.y + 'px,0)'; |
|  |  | } |
|  |  | box.style[transitionDuration] = '150ms'; |
|  |  | } |
|  |  |  |
|  |  | function gameLayerMoveNextRow() { |
|  |  | for (let i = 0; i < GameLayer.length; i++) { |
|  |  | let g = GameLayer[i]; |
|  |  | g.y += blockSize; |
|  |  | if (g.y > blockSize \* (Math.floor(g.children.length / 4))) { |
|  |  | refreshGameLayer(g, 1, -1); |
|  |  | box.style[transform] = 'translate3D(0,' + box.y + 'px,0)'; |
|  |  | setTimeout(function () { |
|  |  | box.style.display = 'block'; |
|  |  | }, 100); |
|  |  | }, 200); |
|  |  | } else { |
|  |  | g.style[transform] = 'translate3D(0,' + g.y + 'px,0)'; |
|  |  | box.y = 0; |
|  |  | box.style[transform] = 'translate3D(0,' + box.y + 'px,0)'; |
|  |  | } |
|  |  | box.style[transitionDuration] = '150ms'; |
|  |  | } |
|  |  | } |
|  |  |  |
|  |  | function gameTapEvent(e) { |
|  |  | if (\_gameOver) { |
|  |  | return false; |
|  |  | w.gameLayerMoveNextRow = function() { |
|  |  | for (let i = 0; i < GameLayer.length; i++) { |
|  |  | let g = GameLayer[i]; |
|  |  | g.y += blockSize; |
|  |  | if (g.y > blockSize \* (Math.floor(g.children.length / 4))) { |
|  |  | refreshGameLayer(g, 1, -1); |
|  |  | } else { |
|  |  | g.style[transform] = 'translate3D(0,' + g.y + 'px,0)'; |
|  |  | } |
|  |  | } |
|  |  | } |
|  |  | let tar = e.target; |
|  |  | let y = e.clientY || e.targetTouches[0].clientY, |
|  |  | x = (e.clientX || e.targetTouches[0].clientX) - body.offsetLeft, |
|  |  | p = \_gameBBList[\_gameBBListIndex]; |
|  |  | if (y > touchArea[0] || y < touchArea[1]) { |
|  |  |  |
|  |  | w.gameTapEvent = function(e) { |
|  |  | if (\_gameOver) { |
|  |  | return false; |
|  |  | } |
|  |  | let tar = e.target; |
|  |  | let y = e.clientY || e.targetTouches[0].clientY, |
|  |  | x = (e.clientX || e.targetTouches[0].clientX) - body.offsetLeft, |
|  |  | p = \_gameBBList[\_gameBBListIndex]; |
|  |  | if (y > touchArea[0] || y < touchArea[1]) { |
|  |  | return false; |
|  |  | } |
|  |  | if ((p.id === tar.id && tar.notEmpty) || (p.cell === 0 && x < blockSize) || (p.cell === 1 && x > blockSize && x < 2 \* |
|  |  | blockSize) || (p.cell === 2 && x > 2 \* blockSize && x < 3 \* blockSize) || (p.cell === 3 && x > 3 \* blockSize)) { |
|  |  | if (!\_gameStart) { |
|  |  | gameStart(); |
|  |  | } |
|  |  | createjs.Sound.play("tap"); |
|  |  | tar = document.getElementById(p.id); |
|  |  | tar.className = tar.className.replace(\_ttreg, ' tt$1'); |
|  |  | \_gameBBListIndex++; |
|  |  | \_gameScore++; |
|  |  | gameLayerMoveNextRow(); |
|  |  | } else if (\_gameStart && !tar.notEmpty) { |
|  |  | createjs.Sound.play("err"); |
|  |  | gameOver(); |
|  |  | tar.className += ' bad'; |
|  |  | } |
|  |  | return false; |
|  |  | } |
|  |  | if ((p.id == tar.id && tar.notEmpty) || (p.cell == 0 && x < blockSize) || (p.cell == 1 && x > blockSize && x < 2 \* |
|  |  | blockSize) || (p.cell == 2 && x > 2 \* blockSize && x < 3 \* blockSize) || (p.cell == 3 && x > 3 \* blockSize)) { |
|  |  | if (!\_gameStart) { |
|  |  | gameStart(); |
|  |  | } |
|  |  | createjs.Sound.play("tap"); |
|  |  | tar = document.getElementById(p.id); |
|  |  | tar.className = tar.className.replace(\_ttreg, ' tt$1'); |
|  |  | \_gameBBListIndex++; |
|  |  | \_gameScore++; |
|  |  | gameLayerMoveNextRow(); |
|  |  | } else if (\_gameStart && !tar.notEmpty) { |
|  |  | createjs.Sound.play("err"); |
|  |  | gameOver(); |
|  |  | tar.className += ' bad'; |
|  |  | } |
|  |  | return false; |
|  |  | } |
|  |  |  |
|  |  | function createGameLayer() { |
|  |  | let html = '<div id="GameLayerBG">'; |
|  |  | for (let i = 1; i <= 2; i++) { |
|  |  | let id = 'GameLayer' + i; |
|  |  | html += '<div id="' + id + '" class="GameLayer">'; |
|  |  | for (let j = 0; j < 10; j++) { |
|  |  | for (let k = 0; k < 4; k++) { |
|  |  | html += '<div id="' + id + '-' + (k + j \* 4) + '" num="' + (k + j \* 4) + '" class="block' + (k ? ' bl' : '') + |
|  |  | '"></div>'; |
|  |  |  |
|  |  | w.createGameLayer = function() { |
|  |  | let html = '<div id="GameLayerBG">'; |
|  |  | for (let i = 1; i <= 2; i++) { |
|  |  | let id = 'GameLayer' + i; |
|  |  | html += '<div id="' + id + '" class="GameLayer">'; |
|  |  | for (let j = 0; j < 10; j++) { |
|  |  | for (let k = 0; k < 4; k++) { |
|  |  | html += '<div id="' + id + '-' + (k + j \* 4) + '" num="' + (k + j \* 4) + '" class="block' + (k ? ' bl' : '') + |
|  |  | '"></div>'; |
|  |  | } |
|  |  | } |
|  |  | html += '</div>'; |
|  |  | } |
|  |  | html += '</div>'; |
|  |  | html += '<div id="GameTimeLayer"></div>'; |
|  |  | return html; |
|  |  | } |
|  |  |  |
|  |  | w.closeWelcomeLayer = function() { |
|  |  | let l = document.getElementById('welcome'); |
|  |  | l.style.display = 'none'; |
|  |  | } |
|  |  |  |
|  |  | w.showWelcomeLayer = function() { |
|  |  | let l = document.getElementById('welcome'); |
|  |  | l.style.display = 'block'; |
|  |  | } |
|  |  |  |
|  |  | w.showGameScoreLayer = function() { |
|  |  | let l = document.getElementById('GameScoreLayer'); |
|  |  | let c = document.getElementById(\_gameBBList[\_gameBBListIndex - 1].id).className.match(\_ttreg)[1]; |
|  |  | l.className = l.className.replace(/bgc\d/, 'bgc' + c); |
|  |  | document.getElementById('GameScoreLayer-text').innerHTML = shareText(\_gameScore); |
|  |  | let score\_text = '得分&nbsp;&nbsp;'; |
|  |  | score\_text += deviation\_time < 23000 ? \_gameScore : "<span style='color:red;'>" + \_gameScore + "</span>"; |
|  |  | document.getElementById('GameScoreLayer-score').innerHTML = score\_text; |
|  |  | let bast = cookie('bast-score'); |
|  |  | if (deviation\_time < 23000) { |
|  |  | if (!bast || \_gameScore > bast) { |
|  |  | bast = \_gameScore; |
|  |  | cookie('bast-score', bast, 100); |
|  |  | } |
|  |  | } |
|  |  | document.getElementById('GameScoreLayer-bast').innerHTML = '最佳&nbsp;&nbsp;' + bast; |
|  |  | l.style.display = 'block'; |
|  |  | } |
|  |  |  |
|  |  | w.hideGameScoreLayer = function() { |
|  |  | let l = document.getElementById('GameScoreLayer'); |
|  |  | l.style.display = 'none'; |
|  |  | } |
|  |  |  |
|  |  | w.replayBtn = function() { |
|  |  | gameRestart(); |
|  |  | hideGameScoreLayer(); |
|  |  | } |
|  |  |  |
|  |  | w.backBtn = function() { |
|  |  | gameRestart(); |
|  |  | hideGameScoreLayer(); |
|  |  | showWelcomeLayer(); |
|  |  | } |
|  |  |  |
|  |  | w.shareText = function(score) { |
|  |  | let date2 = new Date(); |
|  |  | deviation\_time = (date2.getTime() - \_date1.getTime()) |
|  |  | if (deviation\_time > 23000) { |
|  |  | return '倒计时多了' + ((deviation\_time / 1000) - 20).toFixed(2) + "s"; |
|  |  | } |
|  |  | SubmitResults(); |
|  |  | if (score <= 49) return '逊呐'; |
|  |  | if (score <= 99) return '哼哼哼'; |
|  |  | if (score <= 149) return '你是一个一个一个'; |
|  |  | if (score <= 199) return '哼哼哼啊啊啊啊啊'; |
|  |  | return '都撅烂了罢（恼'; |
|  |  | } |
|  |  | html += '</div>'; |
|  |  | html += '<div id="GameTimeLayer"></div>'; |
|  |  | return html; |
|  |  | } |
|  |  |  |
|  |  | function closeWelcomeLayer() { |
|  |  | let l = document.getElementById('welcome'); |
|  |  | l.style.display = 'none'; |
|  |  | } |
|  |  |  |
|  |  | function showWelcomeLayer() { |
|  |  | let l = document.getElementById('welcome'); |
|  |  | l.style.display = 'block'; |
|  |  | } |
|  |  |  |
|  |  | function showGameScoreLayer() { |
|  |  | let l = document.getElementById('GameScoreLayer'); |
|  |  | let c = document.getElementById(\_gameBBList[\_gameBBListIndex - 1].id).className.match(\_ttreg)[1]; |
|  |  | l.className = l.className.replace(/bgc\d/, 'bgc' + c); |
|  |  | document.getElementById('GameScoreLayer-text').innerHTML = shareText(\_gameScore); |
|  |  | let score\_text = '得分&nbsp;&nbsp;'; |
|  |  | score\_text += deviation\_time < 23000 ? \_gameScore : "<span style='color:red;'>" + \_gameScore + "</span>"; |
|  |  | document.getElementById('GameScoreLayer-score').innerHTML = score\_text; |
|  |  | let bast = cookie('bast-score'); |
|  |  | if (deviation\_time < 23000) { |
|  |  | if (!bast || \_gameScore > bast) { |
|  |  | bast = \_gameScore; |
|  |  | cookie('bast-score', bast, 100); |
|  |  |  |
|  |  | w.toStr = function(obj) { |
|  |  | if (typeof obj === 'object') { |
|  |  | return JSON.stringify(obj); |
|  |  | } else { |
|  |  | return obj; |
|  |  | } |
|  |  | } |
|  |  | document.getElementById('GameScoreLayer-bast').innerHTML = '最佳&nbsp;&nbsp;' + bast; |
|  |  | l.style.display = 'block'; |
|  |  | } |
|  |  |  |
|  |  | function hideGameScoreLayer() { |
|  |  | let l = document.getElementById('GameScoreLayer'); |
|  |  | l.style.display = 'none'; |
|  |  | } |
|  |  |  |
|  |  | function replayBtn() { |
|  |  | gameRestart(); |
|  |  | hideGameScoreLayer(); |
|  |  | } |
|  |  |  |
|  |  | function backBtn() { |
|  |  | gameRestart(); |
|  |  | hideGameScoreLayer(); |
|  |  | showWelcomeLayer(); |
|  |  | } |
|  |  |  |
|  |  | function shareText(score) { |
|  |  | let date2 = new Date(); |
|  |  | deviation\_time = (date2.getTime() - \_date1.getTime()) |
|  |  | if (deviation\_time > 23000) { |
|  |  | return '倒计时多了' + ((deviation\_time / 1000) - 20).toFixed(2) + "s"; |
|  |  | } |
|  |  | SubmitResults(); |
|  |  | if (score <= 49) return '逊呐'; |
|  |  | if (score <= 99) return '哼哼哼'; |
|  |  | if (score <= 149) return '你是一个一个一个'; |
|  |  | if (score <= 199) return '哼哼哼啊啊啊啊啊'; |
|  |  | return '都撅烂了罢（恼'; |
|  |  | } |
|  |  |  |
|  |  | function toStr(obj) { |
|  |  | if (typeof obj == 'object') { |
|  |  | return JSON.stringify(obj); |
|  |  | } else { |
|  |  | return obj; |
|  |  | } |
|  |  | } |
|  |  |  |
|  |  | function cookie(name, value, time) { |
|  |  | if (name) { |
|  |  | if (value) { |
|  |  | if (time) { |
|  |  | let date = new Date(); |
|  |  | date.setTime(date.getTime() + 864e5 \* time), time = date.toGMTString(); |
|  |  |  |
|  |  | w.cookie = function(name, value, time) { |
|  |  | if (name) { |
|  |  | if (value) { |
|  |  | if (time) { |
|  |  | let date = new Date(); |
|  |  | date.setTime(date.getTime() + 864e5 \* time), time = date.toGMTString(); |
|  |  | } |
|  |  | return document.cookie = name + "=" + escape(toStr(value)) + (time ? "; expires=" + time + (arguments[3] ? |
|  |  | "; domain=" + arguments[3] + (arguments[4] ? "; path=" + arguments[4] + (arguments[5] ? "; secure" : "") : "") : |
|  |  | "") : ""), !0; |
|  |  | } |
|  |  | return document.cookie = name + "=" + escape(toStr(value)) + (time ? "; expires=" + time + (arguments[3] ? |
|  |  | "; domain=" + arguments[3] + (arguments[4] ? "; path=" + arguments[4] + (arguments[5] ? "; secure" : "") : "") : |
|  |  | "") : ""), !0; |
|  |  | return value = document.cookie.match("(?:^|;)\\s\*" + name.replace(/([-.\*+?^${}()|[\]\/\\])/g, "\\$1") + "=([^;]\*)"), |
|  |  | value = value && "string" == typeof value[1] ? unescape(value[1]) : !1, (/^(\{|\[).+\}|\]$/.test(value) || |
|  |  | /^[0-9]+$/g.test(value)) && eval("value=" + value), value; |
|  |  | } |
|  |  | return value = document.cookie.match("(?:^|;)\\s\*" + name.replace(/([-.\*+?^${}()|[\]\/\\])/g, "\\$1") + "=([^;]\*)"), |
|  |  | value = value && "string" == typeof value[1] ? unescape(value[1]) : !1, (/^(\{|\[).+\}|\]$/.test(value) || |
|  |  | /^[0-9]+$/g.test(value)) && eval("value=" + value), value; |
|  |  | } |
|  |  | let data = {}; |
|  |  | value = document.cookie.replace(/\s/g, "").split(";"); |
|  |  | for (let i = 0; value.length > i; i++) name = value[i].split("="), name[1] && (data[name[0]] = unescape(name[1])); |
|  |  | return data; |
|  |  | } |
|  |  | document.write(createGameLayer()); |
|  |  |  |
|  |  | function initSetting() { |
|  |  | document.getElementById("username").value = cookie("username") ? cookie("username") : ""; |
|  |  | document.getElementById("message").value = cookie("message") ? cookie("message") : ""; |
|  |  | if(cookie("keyboard")){ |
|  |  | document.getElementById("keyboard").value = cookie("keyboard"); |
|  |  | map={} |
|  |  | map[cookie("keyboard").charAt(0).toLowerCase()]=1; |
|  |  | map[cookie("keyboard").charAt(1).toLowerCase()]=2; |
|  |  | map[cookie("keyboard").charAt(2).toLowerCase()]=3; |
|  |  | map[cookie("keyboard").charAt(3).toLowerCase()]=4; |
|  |  | } |
|  |  | } |
|  |  | function show\_btn() { |
|  |  | document.getElementById("btn\_group").style.display = "block" |
|  |  | document.getElementById("setting").style.display = "none" |
|  |  | } |
|  |  | function show\_setting() { |
|  |  | document.getElementById("btn\_group").style.display = "none" |
|  |  | document.getElementById("setting").style.display = "block" |
|  |  | } |
|  |  | function save\_cookie() { |
|  |  | cookie('username', document.getElementById("username").value, 100); |
|  |  | cookie('message', document.getElementById("message").value, 100); |
|  |  | cookie('keyboard', document.getElementById("keyboard").value, 100); |
|  |  | initSetting(); |
|  |  | } |
|  |  | function isnull(val) { |
|  |  | let str = val.replace(/(^\s\*)|(\s\*$)/g, ''); |
|  |  | if (str == '' || str == undefined || str == null) { |
|  |  | return true; |
|  |  | } else { |
|  |  | return false; |
|  |  | let data = {}; |
|  |  | value = document.cookie.replace(/\s/g, "").split(";"); |
|  |  | for (let i = 0; value.length > i; i++) name = value[i].split("="), name[1] && (data[name[0]] = unescape(name[1])); |
|  |  | return data; |
|  |  | } |
|  |  |  |
|  |  | document.write(createGameLayer()); |
|  |  |  |
|  |  | w.initSetting = function() { |
|  |  | document.getElementById("username").value = cookie("username") ? cookie("username") : ""; |
|  |  | document.getElementById("message").value = cookie("message") ? cookie("message") : ""; |
|  |  | if (cookie("keyboard")) { |
|  |  | document.getElementById("keyboard").value = cookie("keyboard"); |
|  |  | map = {} |
|  |  | map[cookie("keyboard").charAt(0).toLowerCase()] = 1; |
|  |  | map[cookie("keyboard").charAt(1).toLowerCase()] = 2; |
|  |  | map[cookie("keyboard").charAt(2).toLowerCase()] = 3; |
|  |  | map[cookie("keyboard").charAt(3).toLowerCase()] = 4; |
|  |  | } |
|  |  | } |
|  |  |  |
|  |  | w.show\_btn = function() { |
|  |  | document.getElementById("btn\_group").style.display = "block" |
|  |  | document.getElementById("setting").style.display = "none" |
|  |  | } |
|  |  |  |
|  |  | w.show\_setting = function() { |
|  |  | document.getElementById("btn\_group").style.display = "none" |
|  |  | document.getElementById("setting").style.display = "block" |
|  |  | } |
|  |  | } |
|  |  | function goRank() { |
|  |  | let name = document.getElementById("username").value; |
|  |  | let link = './rank.php'; |
|  |  | if (!isnull(name)) { |
|  |  | link += "?name=" + name; |
|  |  | } |
|  |  | window.location.href = link; |
|  |  | } |
|  |  |  |
|  |  | function click(index) { |
|  |  | let p = \_gameBBList[\_gameBBListIndex]; |
|  |  | let base = parseInt(document.getElementById(p.id).getAttribute("num")) - p.cell; |
|  |  | let num = base + index - 1; |
|  |  | let id = p.id.substring(0, 11) + num; |
|  |  |  |
|  |  | let fakeEvent = { |
|  |  | clientX: ((index - 1) \* blockSize + index \* blockSize) / 2 + body.offsetLeft, |
|  |  | // Make sure that it is in the area |
|  |  | clientY: (touchArea[0] + touchArea[1]) / 2, |
|  |  | target: document.getElementById(id), |
|  |  | }; |
|  |  |  |
|  |  | gameTapEvent(fakeEvent) |
|  |  | } |
|  |  |  |
|  |  | console.log("不修改，好嘛？乱传又有什么用呢？(ˉ▽ˉ；)...") |
|  |  |  |
|  |  | w.save\_cookie = function() { |
|  |  | cookie('username', document.getElementById("username").value, 100); |
|  |  | cookie('message', document.getElementById("message").value, 100); |
|  |  | cookie('keyboard', document.getElementById("keyboard").value, 100); |
|  |  | initSetting(); |
|  |  | } |
|  |  |  |
|  |  | w.isnull = function(val) { |
|  |  | let str = val.replace(/(^\s\*)|(\s\*$)/g, ''); |
|  |  | return str === '' || str === undefined || str == null; |
|  |  | } |
|  |  |  |
|  |  | w.goRank = function() { |
|  |  | let name = document.getElementById("username").value; |
|  |  | let link = './rank.php'; |
|  |  | if (!isnull(name)) { |
|  |  | link += "?name=" + name; |
|  |  | } |
|  |  | window.location.href = link; |
|  |  | } |
|  |  |  |
|  |  | function click(index) { |
|  |  | let p = \_gameBBList[\_gameBBListIndex]; |
|  |  | let base = parseInt(document.getElementById(p.id).getAttribute("num")) - p.cell; |
|  |  | let num = base + index - 1; |
|  |  | let id = p.id.substring(0, 11) + num; |
|  |  |  |
|  |  | let fakeEvent = { |
|  |  | clientX: ((index - 1) \* blockSize + index \* blockSize) / 2 + body.offsetLeft, |
|  |  | // Make sure that it is in the area |
|  |  | clientY: (touchArea[0] + touchArea[1]) / 2, |
|  |  | target: document.getElementById(id), |
|  |  | }; |
|  |  |  |
|  |  | gameTapEvent(fakeEvent); |
|  |  | } |
|  |  |  |
|  |  | console.log("不修改，好嘛？乱传又有什么用呢？(ˉ▽ˉ；)..."); |
|  |  | }) (window); |