

```
1  /**
2   * File Name: R4.js
3   * Live Demo: http://www.adventurers-of-renown.com/GAMEAPP/index.html
4   * Source Code: http://makingbrowsergames.com/book/
5   * Description: File for controlling and displaying room scenes; managing
6   * global variables throughout game state.
7   * Author: Stephen Gose
8   * Version: 0.0.0.8
9   * Author URL: http://www.stephen-gose.com/
10  * Support: support@pbmcube.com
11  *
12  * Copyright © \u00A9 1974-2017 Stephen Gose LLC. All rights reserved.
13  *
14  * Do not sell! Do not distribute!
15  * This is a licensed permission file. Please refer to Terms of Use
16  * and End Users License Agreement (EULA).
17  * Search for [ /**TODO** ] to tailor this file for your own use and
18  * will void any support agreement.
19  *
20  * Redistribution of part or whole of this file and
21  * the accompanying files is strictly prohibited.
22  */
23  "use strict";
24  window.GAMEAPP.state.R4 = {
25    preload: function(){
26      console.log("loading Room 4 state");
27      GAMEAPP.CrntRoom = 4;
28      this.game.load.image('R4', 'assets/images/staticRooms/R4.jpg');
29      this.game.load.atlas('doorButton', 'assets/images/doors/Doors.png',
30        'assets/images/doors/Doors.json');
31      this.game.load.atlas('avatar', 'assets/icons/arraIcons.png',
32        'assets/icons/arraIcons.json');
33      this.game.load.atlas('tres', 'assets/icons/arraIcons.png',
34        'assets/icons/arraIcons.json');
35      this.load.spritesheet('button',
36        'assets/spriteSheets/mmog-sprites-silver.png', 129, 30);
37      this.load.atlas('inventoryButton',
38        'assets/spriteSheets/mmog-sprites-silver.png',
39        'assets/spriteSheets/mmog-sprites.json');
40      this.load.atlas('exitButton',
41        'assets/spriteSheets/mmog-sprites-silver.png',
42        'assets/spriteSheets/mmog-sprites.json');
43      this.load.atlas('fightButton',
44        'assets/spriteSheets/mmog-sprites-silver.png',
45        'assets/spriteSheets/mmog-sprites.json');
46      this.load.atlas('talkButton',
47        'assets/spriteSheets/mmog-sprites-silver.png',
48        'assets/spriteSheets/mmog-sprites.json');
49    },
50    create: function(){
51      console.log("starting Room 4 state");
52      this.game.add.image(0, 0, 'R4');
53      var rect = new Phaser.Rectangle(37, 105, 300, 200);
54      //Set a neutral background color
55      this.game.stage.backgroundColor = "#000";
56      //Set game to ARCADE physics systemLanguage
```

```
46     this.game.physics.startSystem(Phaser.Physics.ARCADE);
47     this.game.renderer.renderSession.roundPixels = true;
48     this.game.world.enableBody = true;
49
50     //Create Room
51     this.Room = this.game.add.group();
52     this.Room.enableBody = true;
53     this.game.physics.enable(this.Room, Phaser.Physics.ARCADE);
54
55     this.NorthWall = this.game.add.sprite(87,270,box({length:270,width:
56     :16,color:'#999'}));
57     this.NorthWall.enableBody = true;
58     this.NorthWall.body.immovable = true;
59     this.NorthWall.visible = false;
60     this.Room.add(this.NorthWall);
61     this.NorthWall.alignTo(rect, Phaser.TOP_CENTER);
62
63     this.SouthWall = this.game.add.sprite(87,270,box({length:270,width:
64     :16,color:'#999'}));
65     this.SouthWall.body.immovable = true;
66     this.SouthWall.visible = false;
67     this.Room.add(this.SouthWall);
68     this.SouthWall.alignTo(rect, Phaser.BOTTOM_CENTER);
69
70     this.WestWall = this.game.add.sprite(68,110,box({length:16,width:
71     190,color:'#999'}));
72     this.WestWall.body.immovable = true;
73     this.WestWall.visible = false;
74     this.Room.add(this.WestWall);
75
76     this.EastWall = this.game.add.sprite(280,110,box({length:16,width:
77     190,color:'#999'}));
78     this.EastWall.body.immovable = true;
79     this.EastWall.visible = false;
80     this.Room.add(this.EastWall);
81
82     this.doorS = this.game.add.sprite(147,282,'doorButton');
83     //perfectly aligned over placeholder
84     GAMEAPP.doorS = this.doorS;
85     this.doorS.name = "South";
86     this.doorS.frameName = 'horizMetal.bmp';
87     this.doorS.enableBody = true;
88     this.doorS.body.immovable = true;
89     this.doorS.inputEnabled = true;
90     this.game.physics.enable(this.doorS, Phaser.Physics.ARCADE);
91     this.doorS.events.onInputDown.add(this.newRoom, this);
92
93     //create treasure if needed; check for treasure in this room
94     if(MT[GAMEAPP.CrntRoom].Tres > 0){
95         GAMEAPP.InfoText = "Score: "+CCP.Score+"\n Click on the Door
96         to enter.\nThere treasure here! See it?!!";
97         this.Treasure = this.game.add.sprite(200,200,'tres');
98         this.Treasure.frameName = 'treasure.bmp';
99         this.Treasure.body.collideWorldBounds = true;
100        this.Treasure.enableBody = true;
101        this.Treasure.body.immovable = true;
102        this.Treasure.inputEnabled = true;
103        this.game.physics.enable(this.Treasure, Phaser.Physics.ARCADE);
104        //clicking treasure for mobile interaction
```

```
99         this.Treasure.events.onInputDown.add(this.CTreasure, this);
100         this.Treasure.alignIn(rect, Phaser.CENTER);
101
102     }
103
104     this.inventorytxt = this.game.add.text(0, 0, "Inventory" , GAMEAPP
105     .styleBTN); // "Attack" button text;
106     this.exittxt = this.game.add.text(0, 0, "Exit" , GAMEAPP.styleBTN
107     ); // "Attack" button text;
108     this._toolTip = this.game.add.text(this.game.world.width-100, this
109     .game.world.height-90, GAMEAPP.InfoText, GAMEAPP.styleTT);
110     this._toolTip.anchor.set(0.5);
111     GAMEAPP.RoomAlert = "";
112
113     // Inventory button
114     this.inventoryButton = this.game.add.button(65, 338, 'button',
115     GAMEAPP._InventoryRoom, this, 2, 1, 0,1);
116     this.inventoryButton.anchor.set(0.5,0.5);
117     this.inventoryButton.scale.setTo(0.5,0.5);
118     this.inventoryButton.addChild(this.inventorytxt).anchor.set(0.5,
119     0.5);
120
121     // Exit button
122     this.exitButton = this.game.add.button(65, 361, 'button', GAMEAPP.
123     _ExitGame, this, 2, 1, 0,1);
124     this.exitButton.anchor.set(0.5,0.5);
125     this.exitButton.scale.setTo(0.5,0.5);
126     this.exitButton.addChild(this.exittxt).anchor.set(0.5,0.5);
127
128     //create an opponent; Check for monster in this room
129     if(MT[GAMEAPP.CrntRoom].Mnstr > 0){
130         GAMEAPP.RoomAlert = "Beware, a " + M[MT[GAMEAPP.CrntRoom].
131         Mnstr].Race + " is here!"
132         GAMEAPP.raTxt = this.game.add.text(this.game.world.width-268,
133         65, GAMEAPP.RoomAlert, GAMEAPP.styleRA);
134         GAMEAPP.raTxt.anchor.set(0);
135         this.enemy = this.game.add.sprite(MT[GAMEAPP.CrntRoom].mxPos,
136         MT[GAMEAPP.CrntRoom].myPos, 'avatar');
137         this.enemy.frameName = String(M[MT[GAMEAPP.CrntRoom].Mnstr].
138         icon+'.bmp');
139         this.enemy.body.collideWorldBounds = true;
140         this.game.physics.enable(this.enemy, Phaser.Physics.ARCADE);
141
142         this.enemy.alignIn(rect, Phaser.TOP_CENTER);
143
144         GAMEAPP._PUpdate();
145         //clicking fight for mobile interaction
146         // Fight button
147         this.fighttxt = this.game.add.text(0, 0, "Fight" , GAMEAPP.
148         styleBTN);
149         this.fightButton = this.game.add.button(152, 338, 'button',
150         GAMEAPP._CombatCheck, this, 2, 1, 0,1);
151         this.fightButton.anchor.set(0.5,0.5);
152         this.fightButton.scale.setTo(0.5,0.5);
153         this.fightButton.addChild(this.fighttxt).anchor.set(0.5,0.5);
154
155         // Talk button
156         var talktxt = this.game.add.text(0, 0, "Talk" , GAMEAPP.
157         styleBTN);
```

```
145         var talkButton = this.game.add.button(152, 361, 'button', this
146             ._talkToMonster, this, 2, 1, 0,1);
147         talkButton.anchor.set(0.5,0.5);
148         talkButton.scale.setTo(0.5,0.5);
149         talkButton.addChild(talktxt).anchor.set(0.5,0.5);
150     }
151     GAMEAPP._RmAdmin(GAMEAPP.CrntRoom);
152
153     //character placement -
154     var entryBy = new Phaser.Point();
155     if(GAMEAPP.LastDoor=="North" || GAMEAPP.LastDoor=="North2"){
156         entryBy.copyFrom(this.doorS);
157         GAMEAPP.xPos = entryBy.x+15;
158         GAMEAPP.yPos = entryBy.y-35;
159     }
160
161     if(GAMEAPP.LastDoor=="East"){
162         entryBy.copyFrom(this.doorW);
163         GAMEAPP.xPos = entryBy.x+35;
164         GAMEAPP.yPos = entryBy.y+15;
165     }
166
167     if(GAMEAPP.LastDoor=="South"){
168         entryBy.copyFrom(this.doorN);
169         GAMEAPP.xPos = entryBy.x+15;
170         GAMEAPP.yPos = entryBy.y+35;
171     }
172
173     if(GAMEAPP.LastDoor=="West"){
174         entryBy.copyFrom(this.doorE);
175         GAMEAPP.xPos = entryBy.x-35;
176         GAMEAPP.yPos = entryBy.y+15;
177     }
178
179     this.player = this.game.add.sprite(GAMEAPP.xPos,GAMEAPP.yPos,
180         'avatar');
181     this.player.frameName = 'avatar.bmp';
182     this.player.body.collideWorldBounds = true;
183     this.cursor = this.game.input.keyboard.createCursorKeys();
184
185 },
186
187 update: function(){
188     GAMEAPP._MoveChar(this.cursor,this.player);
189     this.game.physics.arcade.collide(this.player, this.Room);
190     if(this.game.physics.arcade.intersects(this.player, this.doorS)){
191         this.newRoom(this.doorS);
192     }
193     //Check for monster in this room
194     if (MT[GAMEAPP.CrntRoom].Mnstr > 0) {
195         GAMEAPP._MoveMonster(this.cursor,this.enemy);
196         this.game.physics.arcade.collide(this.enemy, this.Room);
197         this.game.physics.arcade.overlap(this.player,this.enemy,
198             GAMEAPP._CombatCheck,null,this);
199     }
200     //Check for treasure in this room
```

```
201         if (MT[GAMEAPP.CrntRoom].Tres > 0) {
202
203             this.game.physics.arcade.collide(this.player, this.Treasure,
204                 this.CTreasure, null, this);
205         }
206         this._toolTip.setText(GAMEAPP.InfoText);
207     },
208     //
209     // =====
210
211     //
212     // =====
213     //Main ARRA Door click handler
214     newRoom: function(door){
215         GAMEAPP.LastRoom = GAMEAPP.CrntRoom;
216         GAMEAPP.LastDoor = door.name;
217         console.log('Last Door Used: '+GAMEAPP.LastDoor);
218     /**
219         if (door.name == "North"){
220
221             //debug
222             console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked:
223                 "+door.name);
224             this.camera.fade(0x000000, 200, false);
225             this.time.events.add(200, function() {
226
227                 this.game.state.start('R'+MT[GAMEAPP.CrntRoom].North);
228                 //ARRA Main Entrance (hard coded) rv_3 through rv_8
229             }, this);
230         }
231     */
232     /**
233         if (door.name == "East"){
234
235             //debug
236             console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked:
237                 "+door.name);
238             this.camera.fade(0x000000, 200, false);
239             this.time.events.add(200, function() {
240
241                 this.game.state.start('R'+MT[GAMEAPP.CrntRoom].East);
242                 //ARRA Main Entrance (hard coded) rv_3 through rv_8
243             }, this);
244         }
245     */
246     if (door.name == "South" ){
247
248         //debug
249         console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked: "
250             +door.name);
251         this.camera.fade(0x000000, 200, false);
252         this.time.events.add(200, function() {
253             this.game.state.start('R'+MT[GAMEAPP.CrntRoom].South);
254             //ARRA Main Entrance (hard coded) rv_3 through rv_8
255         }, this);
256     }
257     /**
258         if (door.name == "West"){
```

```
251
252         //debug
253         console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked:
254         "+door.name);
255         this.camera.fade(0x000000, 200, false);
256         this.time.events.add(200, function() {
257
258             this.game.state.start('R'+MT[GAMEAPP.CrntRoom].West);
259             //ARRA Main Entrance (hard coded) rv_3 through rv_8
260             }, this);
261     }
262     */
263     },
264     //
265     //=====
266     _talkToMonster: function(){
267         GAMEAPP.InfoText = "... mmm, not in the mood to negotiate?!";
268     },
269     //
270     //=====
271     CTreasure: function(){
272         CCP.HGold += MT[GAMEAPP.CrntRoom].Tres;
273         this.Treasure.kill();
274         GAMEAPP.InfoText = "You found $" + MT[GAMEAPP.CrntRoom].Tres + " in
275         the chest.";
276         MT[GAMEAPP.CrntRoom].Tres = 0;
277         this._toolTip.setText("You found $" + MT[GAMEAPP.CrntRoom].Tres + "
278         in the chest.");
279     }
280 };
281 //
282 // =====
283 //create a box Image (pseudo graphics) for the HTML5 canvas.
284 var box = function(options) {
285     var bxImg = GAMEAPP.game.add.bitmapData(options.length,options.
286     width);
287     bxImg.ctx.beginPath();
288     bxImg.ctx.rect(0,0,options.length,options.width);
289     bxImg.ctx.fillStyle = options.color;
290     bxImg.ctx.fill();
291     return bxImg;
292 };
293 //
294 // =====
295
296 /* End of file */
297 /* Location: ./js/state/R4.js */
298
299
```