

```
1  /**
2   * File Name: R9.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.R9 = {
25    preload: function(){
26      console.log("loading Room 9 state");
27      this.game.load.image('R9', 'assets/images/staticRooms/R9.jpg');
28      GAMEAPP.CrntRoom = 9;
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.spritesheet('floors',
38        'assets/images/floor/downStairs.bmp', 40, 50);
39      this.load.atlas('inventoryButton',
40        'assets/spriteSheets/mmog-sprites-silver.png',
41        'assets/spriteSheets/mmog-sprites.json');
42      this.load.atlas('exitButton',
43        'assets/spriteSheets/mmog-sprites-silver.png',
44        'assets/spriteSheets/mmog-sprites.json');
45      this.load.atlas('fightButton',
46        'assets/spriteSheets/mmog-sprites-silver.png',
47        'assets/spriteSheets/mmog-sprites.json');
48      this.load.atlas('talkButton',
49        'assets/spriteSheets/mmog-sprites-silver.png',
50        'assets/spriteSheets/mmog-sprites.json');
51    },
52    create: function(){
53      console.log("starting Room 9 state");
54      this.game.add.image(0, 0, 'R9');
55      var rect = new Phaser.Rectangle(130, 105, 100, 130);
56      //Set a neutral background color
```

```
45     this.game.stage.backgroundColor = "#000";
46     //Set game to ARCADE physics systemLanguage
47     this.game.physics.startSystem(Phaser.Physics.ARCADE);
48     this.game.renderer.renderSession.roundPixels = true;
49     this.game.world.enableBody = true;
50
51     //Create Room - aligned to smaller rectangle
52     this.Room = this.game.add.group();
53     this.Room.enableBody = true;
54     this.game.physics.enable(this.Room, Phaser.Physics.ARCADE);
55
56     this.NorthWall = this.game.add.sprite(0,0,box({length:100,width:16
57     ,color:'#999'}));
58     this.NorthWall.enableBody = true;
59     this.NorthWall.body.immovable = true;
60     this.NorthWall.visible = false;
61     this.Room.add(this.NorthWall);
62     this.NorthWall.alignTo(rect, Phaser.TOP_CENTER);
63
64     this.SouthWall = this.game.add.sprite(0,0,box({length:100,width:16
65     ,color:'#999'}));
66     this.SouthWall.body.immovable = true;
67     this.SouthWall.visible = false;
68     this.Room.add(this.SouthWall);
69     this.SouthWall.alignTo(rect, Phaser.BOTTOM_CENTER);
70
71     this.WestWall = this.game.add.sprite(0,0,box({length:16,width:100,
72     color:'#999'}));
73     this.WestWall.body.immovable = true;
74     this.WestWall.visible = false;
75     this.Room.add(this.WestWall);
76     this.WestWall.alignTo(rect, Phaser.LEFT_CENTER);
77
78     this.EastWall = this.game.add.sprite(0,0,box({length:16,width:100,
79     color:'#999'}));
80     this.EastWall.body.immovable = true;
81     this.EastWall.visible = false;
82     this.Room.add(this.EastWall);
83     this.EastWall.alignTo(rect, Phaser.RIGHT_CENTER);
84
85     //Room Doors
86     this.doorS = this.game.add.sprite(150,217,'doorButton');
87     //perfectly aligned over placeholder
88     GAMEAPP.doorS = this.doorS;
89     this.doorS.name = "South";
90     this.doorS.frameName = 'horizWood.bmp';
91     this.doorS.enableBody = true;
92     this.doorS.body.immovable = true;
93     this.doorS.inputEnabled = true;
94     this.game.physics.enable(this.doorS, Phaser.Physics.ARCADE);
95     this.doorS.events.onInputDown.add(this.newRoom, this);
96
97     this.downStairs = this.game.add.sprite(0,0,'floors'); //perfectly
98     aligned over placeholder
99     this.downStairs.name = "DownStairs";
100    this.downStairs.enableBody = true;
101    this.downStairs.body.immovable = true;
102    this.downStairs.inputEnabled = true;
103    this.game.physics.enable(this.downStairs, Phaser.Physics.ARCADE);
```

```

98     this.downStairs.alignIn(rect, Phaser.TOP_CENTER);
99     this.downStairs.events.onInputDown.add(this.newRoom, this);
100
101     //create treasure if needed; check for treasure in this room
102     if(MT[GAMEAPP.CrntRoom].Tres > 0){
103         GAMEAPP.InfoText = "Score: "+CCP.Score+"\n Click on the Door
104         to enter.\nThere treasure here! See it?!!";
105         this.Treasure = this.game.add.sprite(200,200,'tres');
106         this.Treasure.frameName = 'treasure.bmp';
107         this.Treasure.body.collideWorldBounds = true;
108         this.Treasure.enableBody = true;
109         this.Treasure.body.immovable = true;
110         this.Treasure.inputEnabled = true;
111         this.game.physics.enable(this.Treasure, Phaser.Physics.ARCADE);
112         //clicking treasure for mobile interaction
113         this.Treasure.events.onInputDown.add(this.CTreasure, this);
114         this.Treasure.alignIn(rect, Phaser.CENTER);
115     }
116
117     this.inventorytxt = this.game.add.text(0, 0, "Inventory" , GAMEAPP
118     .styleBTN); // "Attack" button text;
119     this.exittxt = this.game.add.text(0, 0, "Exit" , GAMEAPP.styleBTN
120     ); // "Attack" button text;
121     this._toolTip = this.game.add.text(this.game.world.width-100, this
122     .game.world.height-90, GAMEAPP.InfoText, GAMEAPP.styleTT);
123     this._toolTip.anchor.set(0.5);
124     GAMEAPP.RoomAlert = "";
125
126     // Inventory button
127     this.inventoryButton = this.game.add.button(65, 338, 'button',
128     GAMEAPP._InventoryRoom, this, 2, 1, 0,1);
129     this.inventoryButton.anchor.set(0.5,0.5);
130     this.inventoryButton.scale.setTo(0.5,0.5);
131     this.inventoryButton.addChild(this.inventorytxt).anchor.set(0.5,
132     0.5);
133
134     // Exit button
135     this.exitButton = this.game.add.button(65, 361, 'button', GAMEAPP.
136     _ExitGame, this, 2, 1, 0,1);
137     this.exitButton.anchor.set(0.5,0.5);
138     this.exitButton.scale.setTo(0.5,0.5);
139     this.exitButton.addChild(this.exittxt).anchor.set(0.5,0.5);
140
141     //create an opponent; Check for monster in this room
142     if(MT[GAMEAPP.CrntRoom].Mnstr > 0){
143         GAMEAPP.RoomAlert = "Beware, a " + M[MT[GAMEAPP.CrntRoom].
144         Mnstr].Race + " is here!"
145         GAMEAPP.raTxt = this.game.add.text(this.game.world.width-268,
146         65, GAMEAPP.RoomAlert, GAMEAPP.styleRA);
147         GAMEAPP.raTxt.anchor.set(0);
148         this.enemy = this.game.add.sprite(MT[GAMEAPP.CrntRoom].mxPos,
149         MT[GAMEAPP.CrntRoom].myPos, 'avatar');
150         this.enemy.frameName = String(M[MT[GAMEAPP.CrntRoom].Mnstr].
151         icon+'.bmp');
152         this.enemy.body.collideWorldBounds = true;
153         this.game.physics.enable(this.enemy, Phaser.Physics.ARCADE);
154
155         GAMEAPP._PUpdate();

```

```
146         //clicking fight for mobile interaction
147         // Fight button
148         this.fighttxt = this.game.add.text(0, 0, "Fight" , GAMEAPP.
            styleBTN);
149         this.fightButton = this.game.add.button(152, 338, 'button',
            GAMEAPP._CombatCheck, this, 2, 1, 0,1);
150         this.fightButton.anchor.set(0.5,0.5);
151         this.fightButton.scale.setTo(0.5,0.5);
152         this.fightButton.addChild(this.fighttxt).anchor.set(0.5,0.5);
153
154         // Talk button
155         var talktxt = this.game.add.text(0, 0, "Talk" , GAMEAPP.
            styleBTN);
156         var talkButton = this.game.add.button(152, 361, 'button', this
            ._talkToMonster, this, 2, 1, 0,1);
157         talkButton.anchor.set(0.5,0.5);
158         talkButton.scale.setTo(0.5,0.5);
159         talkButton.addChild(talktxt).anchor.set(0.5,0.5);
160     }else{
161         GAMEAPP.InfoText = " Looking for something to kill?\n Try the
            next room.";
162     }
163
164     GAMEAPP._RmAdmin(GAMEAPP.CrntRoom);
165
166         //character placement -
167     var entryBy = new Phaser.Point();
168     if(GAMEAPP.LastDoor=="North" || GAMEAPP.LastDoor=="North2"){
169         entryBy.copyFrom(this.doorS);
170         GAMEAPP.xPos = entryBy.x+15;
171         GAMEAPP.yPos = entryBy.y-35;
172     }
173
174     if(GAMEAPP.LastDoor=="East"){
175         entryBy.copyFrom(this.doorW);
176         GAMEAPP.xPos = entryBy.x+35;
177         GAMEAPP.yPos = entryBy.y+15;
178     }
179
180     if(GAMEAPP.LastDoor=="South"){
181         entryBy.copyFrom(this.doorN);
182         GAMEAPP.xPos = entryBy.x+15;
183         GAMEAPP.yPos = entryBy.y+35;
184     }
185
186     if(GAMEAPP.LastDoor=="West"){
187         entryBy.copyFrom(this.doorE);
188         GAMEAPP.xPos = entryBy.x-35;
189         GAMEAPP.yPos = entryBy.y+15;
190     }
191
192     if(GAMEAPP.LastDoor=="DownStairs"){
193         entryBy.copyFrom(this.upStairs);
194         GAMEAPP.xPos = entryBy.x-35;
195         GAMEAPP.yPos = entryBy.y+15;
196     }
197
198     if(GAMEAPP.LastDoor=="UpStairs"){
199         entryBy.copyFrom(this.downStairs);
```

```

200         GAMEAPP.xPos = entryBy.x; //unique for this room
201         GAMEAPP.yPos = entryBy.y+65;//unique for this room
202     }
203     this.player = this.game.add.sprite(GAMEAPP.xPos,GAMEAPP.yPos,
    'avatar');
204     this.player.frameName = 'avatar.bmp';
205     this.player.body.collideWorldBounds = true;
206     this.cursor = this.game.input.keyboard.createCursorKeys();
207
208 },
209
210 update: function(){
211     GAMEAPP._MoveChar(this.cursor,this.player);
212     this.game.physics.arcade.collide(this.player, this.Room);
213     if(this.game.physics.arcade.intersects(this.player, this.doorS)){
214         this.newRoom(this.doorS);
215     }
216     if(this.game.physics.arcade.intersects(this.player, this.
    downStairs)){
217         this.newRoom(this.downStairs);
218     }
219     //Check for monster in this room
220     if (MT[GAMEAPP.CrntRoom].Mnstr > 0) {
221         GAMEAPP._MoveMonster(this.cursor,this.enemy);
222         this.game.physics.arcade.collide(this.enemy, this.Room);
223         this.game.physics.arcade.overlap(this.player,this.enemy,
    GAMEAPP._CombatCheck,null,this);
224
225     }
226     //Check for treasure in this room
227     if (MT[GAMEAPP.CrntRoom].Tres > 0) {
228
229         this.game.physics.arcade.collide(this.player,this.Treasure,
    this.CTreasure,null,this);
230
231     }
232     this._toolTip.setText(GAMEAPP.InfoText);
233 },
234 //
235 // =====
236
237 //
238 // =====
239 //Main ARRA Door click handler
240 newRoom: function(door){
241     GAMEAPP.LastRoom = GAMEAPP.CrntRoom;
242     GAMEAPP.LastDoor = door.name;
243     console.log('Last Door Used: '+GAMEAPP.LastDoor);
244
245 /**
246     if (door.name == "North"){
247
248         //debug
249         console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked:
    "+door.name);
250         this.camera.fade(0x000000, 200, false);
251         this.time.events.add(200, function() {
252
                this.game.state.start('R'+MT[GAMEAPP.CrntRoom].North);

```

```

253         //ARRA Main Entrance (hard coded) rv_3 through rv_8
254     }, this);
255 }
256 */
257 /**
258     if (door.name == "East"){
259         //debug
260         console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked:
261             "+door.name);
262         this.camera.fade(0x000000, 200, false);
263         this.time.events.add(200, function() {
264             this.game.state.start('R'+MT[GAMEAPP.CrntRoom].East);
265             //ARRA Main Entrance (hard coded) rv_3 through rv_8
266         }, this);
267     }
268 */
269 if (door.name == "South" ){
270     //debug
271     console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked: "
272         +door.name);
273     this.camera.fade(0x000000, 200, false);
274     this.time.events.add(200, function() {
275         this.game.state.start('R'+MT[GAMEAPP.CrntRoom].South);
276         //ARRA Main Entrance (hard coded) rv_3 through rv_8
277     }, this);
278 }
279 /**
280     if (door.name == "West"){
281         //debug
282         console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked:
283             "+door.name);
284         this.camera.fade(0x000000, 200, false);
285         this.time.events.add(200, function() {
286             this.game.state.start('R'+MT[GAMEAPP.CrntRoom].West);
287             //ARRA Main Entrance (hard coded) rv_3 through rv_8
288         }, this);
289     }
290 */
291 if (door.name == "DownStairs"){
292     //debug
293     console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked: "
294         +door.name);
295     this.camera.fade(0x000000, 200, false);
296     this.time.events.add(200, function() {
297         this.game.state.start('R'+MT[GAMEAPP.CrntRoom].Dn);
298         //ARRA Main Entrance (hard coded) rv_3 through rv_8
299     }, this);
300 }
301 },
302 //
303 //=====
304 _talkToMonster: function(){
305     GAMEAPP.InfoText = "... mmm, not in the mood to negotiate?!";

```

```
301     },
302     //
303     //=====
304     CTreasure: function() {
305         CCP.HGold += MT[GAMEAPP.CrntRoom].Tres;
306         this.Treasure.kill();
307         GAMEAPP.InfoText = "You found $" + MT[GAMEAPP.CrntRoom].Tres + " in
            the chest.";
308         MT[GAMEAPP.CrntRoom].Tres = 0;
309         this._toolTip.setText("You found $" + MT[GAMEAPP.CrntRoom].Tres +
            in the chest.");
310     }
311 };
312 //
313 //=====
314 //create a box Image (pseudo graphics) for the HTML5 canvas.
315 var box = function(options) {
316     var bxImg = GAMEAPP.game.add.bitmapData(options.length, options.
        width);
317     bxImg.ctx.beginPath();
318     bxImg.ctx.rect(0, 0, options.length, options.width);
319     bxImg.ctx.fillStyle = options.color;
320     bxImg.ctx.fill();
321     return bxImg;
322 };
323 //
324 //=====
325
326 /* End of file */
327 /* Location: ./js/state/R9.js */
328
329
```