

```
1  /**
2   * File Name: R14.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.R14 = {
25    preload: function(){
26      console.log("loading Room 14 state");
27      this.game.load.image('R14', 'assets/images/staticRooms/R14.jpg');
28      GAMEAPP.CrntRoom = 14;
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 14 state");
52      this.game.add.image(0, 0, 'R14');
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     //Room Doors
83     this.doorS = this.game.add.sprite(139,275,'doorButton');
84     //perfectly aligned over placeholder
85     GAMEAPP.doorS = this.doorS;
86     this.doorS.name = "South";
87     this.doorS.frameName = 'horizWood.bmp';
88     this.doorS.enableBody = true;
89     this.doorS.body.immovable = true;
90     this.doorS.inputEnabled = true;
91     this.game.physics.enable(this.doorS, Phaser.Physics.ARCADE);
92     this.doorS.events.onInputDown.add(this.newRoom, this);
93
94     this.doorE = this.game.add.sprite(254,180,'doorButton');
95     //perfectly aligned over placeholder
96     GAMEAPP.doorE = this.doorE;
97     this.doorE.name = "East";
98     this.doorE.frameName = 'vertWood.bmp';
99     this.doorE.enableBody = true;
100    this.doorE.body.immovable = true;
101    this.doorE.inputEnabled = true;
102    this.game.physics.enable(this.doorE, Phaser.Physics.ARCADE);
103    this.doorE.events.onInputDown.add(this.newRoom, this);
```

```

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

```

```

        styleBTN);
147     this.fightButton = this.game.add.button(152, 338, 'button',
        GAMEAPP._CombatCheck, this, 2, 1, 0,1);
148     this.fightButton.anchor.set(0.5,0.5);
149     this.fightButton.scale.setTo(0.5,0.5);
150     this.fightButton.addChild(this.fighttxt).anchor.set(0.5,0.5);
151
152     // Talk button
153     var talktxt = this.game.add.text(0, 0, "Talk" , GAMEAPP.
        styleBTN);
154     var talkButton = this.game.add.button(152, 361, 'button', this
        ._talkToMonster, this, 2, 1, 0,1);
155     talkButton.anchor.set(0.5,0.5);
156     talkButton.scale.setTo(0.5,0.5);
157     talkButton.addChild(talktxt).anchor.set(0.5,0.5);
158 }
159
160 GAMEAPP._RmAdmin(GAMEAPP.CrntRoom);
161
162 //character placement -
163 var entryBy = new Phaser.Point();
164 if(GAMEAPP.LastDoor=="North" || GAMEAPP.LastDoor=="North2"){
165     entryBy.copyFrom(this.doorS);
166     GAMEAPP.xPos = entryBy.x+15;
167     GAMEAPP.yPos = entryBy.y-35;
168 }
169
170 if(GAMEAPP.LastDoor=="East"){
171     entryBy.copyFrom(this.doorW);
172     GAMEAPP.xPos = entryBy.x+35;
173     GAMEAPP.yPos = entryBy.y+15;
174 }
175
176 if(GAMEAPP.LastDoor=="South"){
177     entryBy.copyFrom(this.doorN);
178     GAMEAPP.xPos = entryBy.x+15;
179     GAMEAPP.yPos = entryBy.y+35;
180 }
181
182 if(GAMEAPP.LastDoor=="West"){
183     entryBy.copyFrom(this.doorE);
184     GAMEAPP.xPos = entryBy.x-35;
185     GAMEAPP.yPos = entryBy.y+15;
186 }
187
188 this.player = this.game.add.sprite(GAMEAPP.xPos,GAMEAPP.yPos,
    'avatar');
189 this.player.frameName = 'avatar.bmp';
190 this.player.body.collideWorldBounds = true;
191 this.cursor = this.game.input.keyboard.createCursorKeys();
192
193 },
194
195 update: function(){
196     GAMEAPP._MoveChar(this.cursor,this.player);
197     this.game.physics.arcade.collide(this.player, this.Room);
198     if(this.game.physics.arcade.intersects(this.player, this.doorS)){
199         this.newRoom(this.doorS);
200     }

```

```

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

```

```

253
254         //debug
255         console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked: "
256         +door.name);
257         this.camera.fade(0x000000, 200, false);
258         this.time.events.add(200, function() {
259             this.game.state.start('R'+MT[GAMEAPP.CrntRoom].South);
260             //ARRA Main Entrance (hard coded) rv_3 through rv_8
261         }, this);
262     }
263 /**
264     if (door.name == "West"){
265         //debug
266         console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked:
267         "+door.name);
268         this.camera.fade(0x000000, 200, false);
269         this.time.events.add(200, function() {
270             this.game.state.start('R'+MT[GAMEAPP.CrntRoom].West);
271             //ARRA Main Entrance (hard coded) rv_3 through rv_8
272         }, this);
273     }
274 */
275 },
276 //
277 //=====
278 _talkToMonster: function(){
279     GAMEAPP.InfoText = "... mmm, not in the mood to negotiate?!";
280 },
281 //
282 //=====
283 CTreasure: function(){
284     CCP.HGold += MT[GAMEAPP.CrntRoom].Tres;
285     this.Treasure.kill();
286     GAMEAPP.InfoText = "You found $" + MT[GAMEAPP.CrntRoom].Tres + " in
287     the chest.";
288     MT[GAMEAPP.CrntRoom].Tres = 0;
289     this._toolTip.setText("You found $" + MT[GAMEAPP.CrntRoom].Tres + "
290     in the chest.");
291 }
292 };
293 //
294 // =====
295 //create a box Image (pseudo graphics) for the HTML5 canvas.
296 var box = function(options) {
297     var bxImg = GAMEAPP.game.add.bitmapData(options.length, options.
298     width);
299     bxImg.ctx.beginPath();
300     bxImg.ctx.rect(0,0,options.length,options.width);
301     bxImg.ctx.fillStyle = options.color;
302     bxImg.ctx.fill();
303     return bxImg;
304 };
305 //
306 // =====
307
308 /* End of file */
309 /* Location: ./js/state/R14.js */

```

304

305