

```
1  /**
2   * File Name: R12.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.R12 = {
25    preload: function(){
26      console.log("loading Room 12 state");
27      this.game.load.image('R12', 'assets/images/staticRooms/R12.jpg');
28      GAMEAPP.CrntRoom = 12;
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 12 state");
52      this.game.add.image(0, 0, 'R12');
53      var rect = new Phaser.Rectangle(85, 170, 195, 80);
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 - aligned to smaller rectangle
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(0,0,box({length:170,width:16
56     ,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(0,0,box({length:170,width:16
64     ,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(0,0,box({length:16,width:67,
71     color:'#999'}));
72     this.WestWall.body.immovable = true;
73     this.WestWall.visible = false;
74     this.Room.add(this.WestWall);
75     this.WestWall.alignTo(rect, Phaser.LEFT_CENTER);
76
77     this.EastWall = this.game.add.sprite(0,0,box({length:16,width:67,
78     color:'#999'}));
79     this.EastWall.body.immovable = true;
80     this.EastWall.visible = false;
81     this.Room.add(this.EastWall);
82     this.EastWall.alignTo(rect, Phaser.RIGHT_CENTER);
83
84     //Room Doors
85
86     this.doorW = this.game.add.sprite(77,176,'doorButton');
87     //perfectly aligned over placeholder
88     GAMEAPP.doorW = this.doorW;
89     this.doorW.name = "West";
90     this.doorW.frameName = 'vertWood.bmp';
91     this.doorW.enableBody = true;
92     this.doorW.body.immovable = true;
93     this.doorW.inputEnabled = true;
94     this.game.physics.enable(this.doorW, Phaser.Physics.ARCADE);
95     this.doorW.events.onInputDown.add(this.newRoom, this);
96
97     //create treasure if needed; check for treasure in this room
98     if(MT[GAMEAPP.CrntRoom].Tres > 0){
99         GAMEAPP.InfoText = "Score: "+CCP.Score+"\n Click on the Door
100         to enter.\nThere treasure here! See it?!!";
101         this.Treasure = this.game.add.sprite(200,200,'tres');
102         this.Treasure.frameName = 'treasure.bmp';
103         this.Treasure.body.collideWorldBounds = true;
104         this.Treasure.enableBody = true;
```

```

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

```

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

```
201         //Check for treasure in this room
202         if (MT[GAMEAPP.CrntRoom].Tres > 0) {
203
204             this.game.physics.arcade.collide(this.player, this.Treasure,
205                 this.CTreasure, null, this);
206
207         }
208         this._toolTip.setText(GAMEAPP.InfoText);
209     },
210     //
211     // =====
212     //
213     // =====
214     //Main ARRA Door click handler
215     newRoom: function(door){
216         GAMEAPP.LastRoom = GAMEAPP.CrntRoom;
217         GAMEAPP.LastDoor = door.name;
218         console.log('Last Door Used: '+GAMEAPP.LastDoor);
219     /**
220         if (door.name == "North"){
221
222             //debug
223             console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked:
224                 "+door.name);
225             this.camera.fade(0x000000, 200, false);
226             this.time.events.add(200, function() {
227
228                 this.game.state.start('R'+MT[GAMEAPP.CrntRoom].North);
229                 //ARRA Main Entrance (hard coded) rv_3 through rv_8
230             }, this);
231         }
232     */
233     /**
234         if (door.name == "East"){
235
236             //debug
237             console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked:
238                 "+door.name);
239             this.camera.fade(0x000000, 200, false);
240             this.time.events.add(200, function() {
241
242                 this.game.state.start('R'+MT[GAMEAPP.CrntRoom].East);
243                 //ARRA Main Entrance (hard coded) rv_3 through rv_8
244             }, this);
245         }
246     */
247     /**
248         if (door.name == "South" ){
249
250             //debug
251             console.log("New Room #: "+GAMEAPP.CrntRoom+"; Door Clicked:
252                 "+door.name);
253             this.camera.fade(0x000000, 200, false);
254             this.time.events.add(200, function() {
255
256                 this.game.state.start('R'+MT[GAMEAPP.CrntRoom].South);
257                 //ARRA Main Entrance (hard coded) rv_3 through rv_8
258             }, this);
259         }
260     */
261     /**
```

```
250     }
251     */
252     if (door.name == "West"){
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].West);
260             //ARRA Main Entrance (hard coded) rv_3 through rv_8
261         }, this);
262     }
263     //
264     //=====
265     _talkToMonster: function(){
266         GAMEAPP.InfoText = "... mmm, not in the mood to negotiate?!";
267     },
268     //
269     //=====
270     CTreasure: function(){
271         CCP.HGold += MT[GAMEAPP.CrntRoom].Tres;
272         this.Treasure.kill();
273         GAMEAPP.InfoText = "You found $" + MT[GAMEAPP.CrntRoom].Tres + " in
274             the chest.";
275         MT[GAMEAPP.CrntRoom].Tres = 0;
276         this._toolTip.setText("You found $" + MT[GAMEAPP.CrntRoom].Tres + "
277             in the chest.");
278     }
279 };
280 //
281 // =====
282 //create a box Image (pseudo graphics) for the HTML5 canvas.
283 var box = function(options) {
284     var bxImg = GAMEAPP.game.add.bitmapData(options.length, options.
285         width);
286     bxImg.ctx.beginPath();
287     bxImg.ctx.rect(0,0,options.length,options.width);
288     bxImg.ctx.fillStyle = options.color;
289     bxImg.ctx.fill();
290     return bxImg;
291 };
292 //
293 // =====
294 /* End of file */
295 /* Location: ./js/state/R12.js */
```