

```
1  /**
2   * File Name: R19.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.R19 = {
25    preload: function(){
26      console.log("loading Room 19 state");
27      this.game.load.image('R19', 'assets/images/staticRooms/R19.jpg');
28      GAMEAPP.CrntRoom = 19;
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 19 state");
52      this.game.add.image(0, 0, 'R19');
53      var rect = new Phaser.Rectangle(75, 150, 200, 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:270,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:270,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:190,
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:190,
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     this.doorN = this.game.add.sprite(137,136,'doorButton');
86     //perfectly aligned over placeholder
87     GAMEAPP.doorN = this.doorN;
88     this.doorN.name = "North";
89     this.doorN.frameName = 'horizWood.bmp';
90     this.doorN.enableBody = true;
91     this.doorN.body.immovable = true;
92     this.doorN.inputEnabled = true;
93     this.game.physics.enable(this.doorN, Phaser.Physics.ARCADE);
94     this.doorN.events.onInputDown.add(this.newRoom, this);
95
96     this.doorE = this.game.add.sprite(254,155,'doorButton');
97     //perfectly aligned over placeholder
98     GAMEAPP.doorE = this.doorE;
99     this.doorE.name = "East";
100    this.doorE.frameName = 'vertWood.bmp';
101    this.doorE.enableBody = true;
102    this.doorE.body.immovable = true;
103    this.doorE.inputEnabled = true;
104    this.game.physics.enable(this.doorE, Phaser.Physics.ARCADE);
```

```

99         this.doorE.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();
156             //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     this.player = this.game.add.sprite(GAMEAPP.xPos,GAMEAPP.yPos,
        'avatar');
193     this.player.frameName = 'avatar.bmp';
194     this.player.body.collideWorldBounds = true;
195     this.cursor = this.game.input.keyboard.createCursorKeys();
196
197 },
198
199     update: function() {
```

```

200     GAMEAPP._MoveChar(this.cursor,this.player);
201     this.game.physics.arcade.collide(this.player, this.Room);
202     if(this.game.physics.arcade.intersects(this.player, this.doorN)){
203         this.newRoom(this.doorN);
204     }
205     if(this.game.physics.arcade.intersects(this.player, this.doorE)){
206         this.newRoom(this.doorE);
207     }
208     //Check for monster in this room
209     if (MT[GAMEAPP.CrntRoom].Mnstr > 0) {
210         GAMEAPP._MoveMonster(this.cursor,this.enemy);
211         this.game.physics.arcade.collide(this.enemy, this.Room);
212         this.game.physics.arcade.overlap(this.player,this.enemy,
213             GAMEAPP._CombatCheck,null,this);
214     }
215     //Check for treasure in this room
216     if (MT[GAMEAPP.CrntRoom].Tres > 0) {
217
218         this.game.physics.arcade.collide(this.player,this.Treasure,
219             this.CTreasure,null,this);
220     }
221     this._toolTip.setText(GAMEAPP.InfoText);
222 },
223 //
224 // =====
225
226 //
227 // =====
228 //Main ARRA Door click handler
229 newRoom: function(door){
230     GAMEAPP.LastRoom = GAMEAPP.CrntRoom;
231     GAMEAPP.LastDoor = door.name;
232     console.log('Last Door Used: '+GAMEAPP.LastDoor);
233
234     if (door.name == "North"){
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             this.game.state.start('R'+MT[GAMEAPP.CrntRoom].North);
242             //ARRA Main Entrance (hard coded) rv_3 through rv_8
243         }, this);
244     }
245
246     if (door.name == "East"){
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].East);
254             //ARRA Main Entrance (hard coded) rv_3 through rv_8
255         }, this);

```

```

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

```

```
303      //
304      // =====
305
306      /* End of file */
307      /* Location: ./js/state/R19.js */
308
309
```