

```
1  /**
2   * File Name: exit.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 game exit and gargabe collection.
6   * Author: Stephen Gose
7   * Version: 0.0.0.8
8   * Author URL: http://www.stephen-gose.com/
9   * Support: support@pbmcube.com
10  *
11  * Copyright © \u00A9 1974-2017 Stephen Gose LLC. All rights reserved.
12  *
13  * Do not sell! Do not distribute!
14  * This is a licensed permission file. Please refer to Terms of Use
15  * and End Users License Agreement (EULA).
16  * Search for [ /***TODO** ] to tailor this file for your own use and
17  * will void any support agreement.
18  *
19  * Redistribution of part or whole of this file and
20  * the accompanying files is strictly prohibited.
21  */
22  "use strict";
23  window.GAMEAPP.state.exitGame = {
24    preload: function() {
25      console.log("loading Exit Game state");
26      GAMEAPP.CrntRoom = GAMEAPP.Logout;
27      this.game.load.image('exitGame',
28        'assets/images/staticRooms/exitGame.jpg');
29      this.load.spritesheet('button',
30        'assets/spriteSheets/mnog-sprites-silver.png', 129, 30);
31      this.load.spritesheet('Death',
32        'assets/spriteSheets/ARRADeath/ARRADeath.png', 160, 100);
33      this.load.atlas('inventoryButton',
34        'assets/spriteSheets/mnog-sprites-silver.png',
35        'assets/spriteSheets/mnog-sprites.json');
36      this.load.atlas('exitButton',
37        'assets/spriteSheets/mnog-sprites-silver.png',
38        'assets/spriteSheets/mnog-sprites.json');
39      this.load.atlas('saveButton',
40        'assets/spriteSheets/mnog-sprites-silver.png',
41        'assets/spriteSheets/mnog-sprites.json');
42      this.load.atlas('shareButton',
43        'assets/spriteSheets/mnog-sprites-silver.png',
44        'assets/spriteSheets/mnog-sprites.json');
45      this.load.atlas('submitButton',
46        'assets/spriteSheets/mnog-sprites-silver.png',
47        'assets/spriteSheets/mnog-sprites.json');
48    },
49    create: function() {
50      console.log("starting Exit Game state");
51      this.game.add.image(0, 0, 'exitGame');
52      //Set a neutral background color
53      this.game.stage.backgroundColor = "#000";
54
55      //ARRA Character Death animation
56      this.ARRADeath = this.game.add.sprite(372, 212, 'Death');
```

```
46 // Here we add a new animation called 'show Death'
47 // Because we didn't give any other parameters it's going to
// make an animation from all available frames in the ARRA Death
// sprite sheet
48 var showDeath = this.ARRADeath.animations.add('showDeath');
49
50 // And this starts the animation playing by using its key
// ("showDeath")
51 // 30 is the frame rate (30fps)
52 // true means it will loop when it finishes
53 this.ARRADeath.animations.play('showDeath', 8, true);
54
55 this.inventorytxt = this.game.add.text(0, 0, "Inventory" ,
GAMEAPP.styleBTN);
56 this.exittxt = this.game.add.text(0, 0, "Exit" , GAMEAPP.
styleBTN);
57 this.submitScore = this.game.add.text(0, 0, "Submit" , GAMEAPP
.styleBTN);
58 this.save = this.game.add.text(0, 0, "Save" , GAMEAPP.styleBTN
);
59 this.newGame = this.game.add.text(0, 0, "New Game?" , GAMEAPP.
styleBTN);
60 this.newGame.events.onInputDown.add(this._NewGame, this);
61 this.share = this.game.add.text(0, 0, "Share" , GAMEAPP.
styleBTN);
62
63 // Inventory button
64 this.inventoryButton = this.game.add.button(65, 338, 'button',
this._Inventory, this, 2, 1, 0,1);
65 this.inventoryButton.anchor.set(0.5,0.5);
66 this.inventoryButton.scale.setTo(0.5,0.5);
67 this.inventoryButton.addChild(this.inventorytxt).anchor.set(
0.5,0.5);
68
69 // Exit button
70 this.exitButton = this.game.add.button(65, 361, 'button',
GAMEAPP._ExitGame, this, 2, 1, 0,1);
71 this.exitButton.anchor.set(0.5,0.5);
72 this.exitButton.scale.setTo(0.5,0.5);
73 this.exitButton.addChild(this.exittxt).anchor.set(0.5,0.5);
74
75 // New Game button
76 this.newButton = this.game.add.button(240, 338, 'button',
GAMEAPP._newGame, this, 2, 1, 0,1);
77 this.newButton.anchor.set(0.5,0.5);
78 this.newButton.scale.setTo(0.5,0.5);
79 this.newButton.addChild(this.newGame).anchor.set(0.5,0.5);
80
81 // Submit button
82 this.submitButton = this.game.add.button(152, 338, 'button',
this._SubmitRoom, this, 2, 1, 0,1);
83 this.submitButton.anchor.set(0.5,0.5);
84 this.submitButton.scale.setTo(0.5,0.5);
85 this.submitButton.addChild(this.submitScore).anchor.set(0.5,
0.5);
86
87 // Save button
88 this.saveButton = this.game.add.button(152, 361, 'button',
this._SaveRoom, this, 2, 1, 0,1);
```

```

89     this.saveButton.anchor.set(0.5,0.5);
90     this.saveButton.scale.setTo(0.5,0.5);
91     this.saveButton.addChild(this.save).anchor.set(0.5,0.5);
92
93     // Share button
94     this.shareButton = this.game.add.button(240, 361, 'button',
this._ShareRoom, this, 2, 1, 0,1);
95     this.shareButton.anchor.set(0.5,0.5);
96     this.shareButton.scale.setTo(0.5,0.5);
97     this.shareButton.addChild(this.share).anchor.set(0.5,0.5);
98
99     this._toolTip = this.game.add.text(this.game.world.width-100,
this.game.world.height-90, GAMEAPP.InfoText, GAMEAPP.styleTT);
100    this._toolTip.anchor.set(0.5);
101    GAMEAPP.InfoText = "Experience & Score: "+String(CCP.Score)+
"\n Renown Earned: "+String(CCP.Renown);
102    GAMEAPP.RoomAlert = "Quest failed!";
103    //rv_5
104    //Send update to server as a "heart beat / keep alive"
105    //TXBuffer();
106    //Heart.send("update.asp",POST);
107
108    //rv_8
109    //IBPrpArcade game scores enabled
110    var gname="ARRA-PBMCube";
111    var pname = CCP.Name;
112    var gscore = CCP.Renown;
113    //getURL("index.php?act=Arcade&do=newscore", "_blank", "POST");
114
115    },
116
117    update: function(){
118        this._toolTip.setText(GAMEAPP.InfoText);
119
120        //this.ARRADeath.x = this.game.world.width-178;
121        //this.ARRADeath.y = 212;
122        this.ARRADeath.animations.play('showDeath', 8, true);
123    },
124
125    //
126    // =====
127    _InventoryRoom: function(){
128
129        //debug
130        console.log("Room #: Exit defeated; Inventory Clicked: ");
131
132        this.camera.fade(0x000000, 200, false);
133        this.time.events.add(200, function() {
134            this.game.state.start('Inventory'); //Inventory Scene
            rv_3 through rv_8
135        }, this);
136
137    },
138    //
139    // =====
140    _NewGame: function(){
141        //Refer to book Bonus Content server-side middleware
142        //debug
143        console.log("Room #: Exit defeated; New Game Clicked: ");

```

```
144         console.log(' - Equip ALL Monsters in Database once per game.'
145         );
146         GAMEAPP._MonsterPrep();
147         GAMEAPP._AllotMonster();
148
149         console.log(' - Creating internal static Monster Database:');
150         console.table(M);
151
152         console.log(' - Creating internal static Movement Table:\n See
153         book for dynamically generated Movement Tables and Mazes. ');
154         console.log('Distributing 6 Treasures and 7 Monsters once per
155         game. ');
156         GAMEAPP._AllotTreasure();
157         console.table(MT);
158
159         this.camera.fade(0x000000, 200, false);
160         this.time.events.add(200, function() {
161             this.game.state.start('boot'); //Boot/Preload Scene
162             rv_3 through rv_8
163         }, this);
164     },
165     //
166     // =====
167     _SaveRoom: function() {
168         //Refer to book Bonus Content server-side middleware
169         //debug
170         console.log("Room #: Exit defeated; Save Clicked: ");
171     },
172     //
173     // =====
174     _ShareRoom: function() {
175         //Refer to book Bonus Content server-side middleware
176         //debug
177         console.log("Room #: Exit defeated; Share Clicked: ");
178     },
179     //
180     // =====
181     _SubmitRoom: function() {
182         //Refer to book Bonus Content server-side middleware
183         //debug
184         console.log("Room #: Exit defeated; Submit Clicked: ");
185     }
186 };
187
188 /* End of file */
189 /* Location: ./js/state/exitGame.js */
```