

```
1  /**
2   * File Name: Main.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 game 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; doing
18  * so 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 /** game set-up */
25 //Beginning of Static databases
26 /**TODO**
27 // NOTE: the following data structures are for development only.
28 // The optimum usage is a local or remote database using PouchDB or
29 SQLite.
30 // Stay away from IndexedDB since it is deprecated.
31 console.log("%c      Starting my awesome MMoG game Prototype!      \n
32 Adventurers of Renown: Ruins of Able-Wyvern rv_8      \n      Copyright
33 \u00A9 1974-2017, Stephen Gose.      \n | http://pbmcube.net/shop
34 |      \n | \u2665\u2665\u2665\u2665\u2665 -> $120
35 License included in book!      \n | Book available at:
36 http://leanpub.com/LoRD |      ",
37 "color:white; background:blue");
38 console.log('Initial Avatar Character for FREE Game play;\n Members can
39 customize characters.');
```

```
40 //
41 // =====
42 function ArmorClass(
43     a1, a2, a3, a4, a5, a6, a7, a8, a9, a10, a11, a12, a13, a14)
44 {
45     this.AID = a1;
46     this.Name = a2;
47     this.Protect = a3;
48     this.Stack = a4;
49     this.Cost = a5;
50     this.Worth = a6;
51     // New property. Current armor value to sell
52     this.Weight = a7;
53     this.CoorMod = a8;
54     this.MoveMod = a9;
55     this.StmnReq = a10;
56     // New property for ARRA v3.3
57     this.BodyLoc = a11;
```

```

51     // 0=Shield;1=Head;2=Body;3=Arms;4=Legs
52     this.DefenseType = a12;
53     // Chop=C;Slash=S;Smash=S;Thrust=T: "CSST"
54     this.Magic = a13;
55     this.Renown = a14;
56     // Class Inherited Methods:
57 }
58 // End Armor Class
59 // =====
60 // Prep Frame 2
61 function PersonClass(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14,p15,
p16,p17,p18,p19,p20,p21,p22,p23,p24,p25,p26,p27,p28) {
62     this.PID = p1; //default - 0
63     this.CID = p2; //default - 1
64     this.Name = p3; //default - Common Adventurer
65     this.Score = p4; //0
66     this.TempScore = 0;
67     this.Category = p5; //Warrior
68     this.Health = p6; //Healthy
69     this.Race = p7; //Folks
70     this.Stmn = Number(p8); //12
71     this.ModStmn = Number(p9);
72     this.Fatigue = Number(p10);
73     //p11? - future use
74     this.Coor = Number(p12); //12
75     this.Psych = Number(p13); //8
76     this.ModIQ = Number(p14); //8
77     this.Renown = Number(p15); //1
78     this.HGold = Number(p16); //0
79     this.HGem = Number(p17); //0
80     this.Movement = p18; //10
81     this.MegaSQ = 1;
82     this.Room = 6;
83     this.Food = Number(p19); //1
84     this.WSRaw = Number(p20); //2
85     this.WSCmbt = p21; //NO
86     this.BSRaw = Number(p22); //2
87     this.BSCmbt = p23; //NO
88     this.AtkFlag = 0;
89     this.MisFlag = 0;
90     this.PryFlag = 0;
91     this.HitFlag = 0;
92     this.EngFlag = 0;
93     this.MovFlag = 0;
94     this.Target = 6;
95     this.TLoc = 2;
96     this.TotalAP = Number(p24); //2
97     this.Shield = p25; //Shield Name?
98     this.Arrows = Number(p26); //0
99     this.AName = p27; //Body Armor Name
100    this.WName = p28; //Primary Weapon Name
101
102    this.W = [];
103    // Weapons Array for this person
104    this.W[0] = {WID:0,Name:"Short Sword",Damage:2,DamMod:0,DamType:"CSST",
Mode:"MMT",Cost:0,Worth:0,Weight:0,ReqStmn:0,ReqCoor:0,Use:0,Magic:0,
Renown:0};
105    // The Weapon in USE!!
106    this.W[1] = {WID:0,Name:"None",Damage:0,DamMod:0,DamType:"CSST",Mode:

```

```

    "MMT",Cost:0,Worth:0,Weight:0,ReqStmn:0,ReqCoor:0,Use:0,Magic:0,Renown
    :0};
107 // Primary Weapon = [1];Secondary = [2]
108 this.W[2] = {WID:0,Name:"None",Damage:0,DamMod:0,DamType:"CSST",Mode:
    "MMT",Cost:0,Worth:0,Weight:0,ReqStmn:0,ReqCoor:0,Use:0,Magic:0,Renown
    :0};
109 // Secondary Weapon = [2]
110 this.W[3] = {WID:0,Name:"None",Damage:0,DamMod:0,DamType:"CSST",Mode:
    "MMT",Cost:0,Worth:0,Weight:0,ReqStmn:0,ReqCoor:0,Use:0,Magic:0,Renown
    :0};
111 // Backup Weapon = [3]
112 // Armor Array for this person
113 this.A = [];
114 // Shield position
115 this.A[0] = {AID:0,Name:"Small Shield",Protect:1,Stack:"Y",DefType:
    "CSST",BodyLoc:"Shield",Cost:0,Worth:0,Weight:0,CoorMod:0,MoveMod:0,
    ReqStmn:0,Magic:0,Renown:0};
116 // Helm position
117 this.A[1] = {AID:0,Name:"None",Protect:0,Stack:0,DefType:"CSST",
    BodyLoc:"Head",Cost:0,Worth:0,Weight:0,CoorMod:0,MoveMod:0,ReqStmn:0,
    Magic:0,Renown:0};
118 // Body Armor is Leather Jerkin
119 this.A[2] = {AID:0,Name:"Leather Jerkin",Protect:2,Stack:"Y",DefType:
    "CS--",BodyLoc:"Body",Cost:0,Worth:0,Weight:0,CoorMod:2,MoveMod:0,
    ReqStmn:0,Magic:0,Renown:0};
120 // Arm's Armor position
121 this.A[3] = {AID:0,Name:"None",Protect:0,Stack:0,DefType:"CSST",
    BodyLoc:"Arms",Cost:0,Worth:0,Weight:0,CoorMod:0,MoveMod:0,ReqStmn:0,
    Magic:0,Renown:0};
122 // Leg's Armor position
123 this.A[4] = {AID:0,Name:"None",Protect:0,Stack:0,DefType:"CSST",
    BodyLoc:"Legs",Cost:0,Worth:0,Weight:0,CoorMod:0,MoveMod:0,ReqStmn:0,
    Magic:0,Renown:0};
124
125 // PersonClass Inherited Methods:
126 this.ModMove = function () { return this.Movemnt-(this.A[0].MoveMod+
    this.A[2].MoveMod);};
127 this.ModCoor = function () { return this.Coor-(this.A[0].CoorMod+this.
    A[2].CoorMod);};
128 this.Level = function () { return (((this.Stmn+this.Coor+this.Psych)-
    26)/6);};
129 this.WS = function () { return ((this.Stmn*2)+(this.WSRaw*5));};
130 this.BS = function () { return ((PersonClass.prototype.ModCoor*2)+(
    this.BSRaw*5));};
131 this.PS = function () { return ((PersonClass.prototype.ModCoor*2)+(
    this.WSRaw*5));};
132 }
133 // End PersonClass
134 // =====
135 //Prep Scene Frame 1
136 function WeaponClass(w1, w2, w3, w4, w5, w6, w7, w8, w9, w10, w11, w12,
    w13, w14) {
137     this.WID = w1;
138     this.Name = w2;
139     this.DamPossible = w3;
140     this.DamMod = w4;
141     this.Cost = w5;
142     this.Worth = w6;
143     // New property. Current armor value to sell

```

```
144     this.Weight = w7;
145     this.StmnReq = w8;
146     this.CoorReq = w9;
147     this.Use = w10;
148     // 1=One handed; 2=Both hands
149     this.Mode = w11;
150     // M=Melee;Missile=M;T=Thrown: "MMT"
151     this.DamType = w12;
152     // Chop=C;Slash=S;Smash=S;Thrust=T: "CSST"
153     this.Magic = w13;
154     this.Renown = w14;
155     this.DamageDone = GetWpnDamage();
156     // Damage Delt per call
157     // Class Inherited Methods:
158 }
159 // End Weapons Class
160 // =====
161 /**TODO**
162 //Prep Scene Frame 1
163 // Root TimeLine Variables (Global for this timeline)
164 //var SecondaryWpns = new WeaponClass(10, "Dagger", 1, 0, 50, 50, 1, 1,
165 4, 1, "M-T", "CS-T", 0, 0);
166 //var PrimaryWpns = new WeaponClass(23, "Short Sword", 2, -1, 200, 200,
167 3, 10, 4, 1, "M--", "CS-T", 0, 0);
168 //var NoWpns = new WeaponClass(0, "None", 0, 0, 0, 0, 0, 0, 0, 0, "----",
169 "----", 0, 0);
170 //var BodyArm = new ArmorClass(103, "Leather Jerkin", 2, "No", 100, 100,
171 2, -2, -2, 8, 2, "-S--", 0, 0);
172 //var NoneArm = new ArmorClass(0, "None", 0, "No", 0, 0, 0, 0, 0, 0, 0, 0,
173 "----", 0, 0);
174
175 /**TODO**
176 // Set-up MMoG capabilities; Person(1) intialized prior to each Combat
177 as a temporary combat record
178 // The follow could be in external files to initialize the char's combat
179 records
180 // for the campaigne.
181 console.log('MMoG Database and Combat records:\n up to 4 member/player
182 teams!\n Records 5-8 are monster team records.\n Player Avatar is
183 record 1\n Monster encountered uses record 6.\nRefer to book to migrate
184 into a SQLite database or PouchDB.');
```

```
175 var Person = [];
176 Person[1] = {Target:6, TotalAP:0, WS:0, BS:0, PS:0, ModStmn:0, TempScore:0
177 , Renown:0};
178 Person[2] = {Target:6, TotalAP:0, WS:0, BS:0, PS:0, ModStmn:0, TempScore:0
179 , Renown:0};
180 Person[3] = {Target:6, TotalAP:0, WS:0, BS:0, PS:0, ModStmn:0, TempScore:0
181 , Renown:0};
182 Person[4] = {Target:6, TotalAP:0, WS:0, BS:0, PS:0, ModStmn:0, TempScore:0
183 , Renown:0};
184 //below are monster combat records
185 Person[5] = {Target:1, TotalAP:0, WS:0, BS:0, PS:0, ModStmn:0, TempScore:0
186 , Renown:0};
187 Person[6] = {Target:1, TotalAP:0, WS:0, BS:0, PS:0, ModStmn:0, TempScore:0
188 , Renown:0};
189 Person[7] = {Target:1, TotalAP:0, WS:0, BS:0, PS:0, ModStmn:0, TempScore:0
190 , Renown:0};
191 Person[8] = {Target:1, TotalAP:0, WS:0, BS:0, PS:0, ModStmn:0, TempScore:0
192 , Renown:0};
```

```
185
186 //CCP - Current Character Played
187 /**TODO** Set for remote login and data access into members' personal
188 avatar records
189 var CCP = new PersonClass(0,1,"Common Adventurer",0,"Warrior","Healthy",
190 "Folks",20,20,20,20,20,8,8,1,0,0,10,1,2,"N",2,"N",3,"Small Shield",0,
191 "Leather Jerkin","Short Sword");
192 //console.table(CCP);
193
194 //Create Monster Array using MonsterClass object
195 /**TODO** Remote this static database onto database server; otherwise
196 use PouchDB or SQLite
197 var M = [];
198 M[1] = {icon:"barbarian",Race:"Folks",Category:"Warrior",Stmn:15,Coor:14,
199 Psych:8,Renown:3,HGold:47,HGem:1,Food:1,WSRaw:3,BSRaw:1,Move:10,Room:0,
200 Armor:"",TotalAP:0,Weapn:"",WDam:0};
201 M[2] = {icon:"dwarf",Race:"Folks",Category:"Warrior",Stmn:14,Coor:13,Psych
202 :8,Renown:3,HGold:76,HGem:0,Food:0,WSRaw:3,BSRaw:2,Move:10,Room:0,Armor:"",
203 ,TotalAP:0,Weapn:"",WDam:0};
204 M[3] = {icon:"darkPriest",Race:"Dark Elf",Category:"Rogue",Stmn:13,Coor:16
205 ,Psych:9,Renown:3,HGold:89,HGem:2,Food:2,WSRaw:2,BSRaw:3,Move:15,Room:0,
206 TotalAP:0,Armor:"",Weapn:"",WDam:0};
207 M[4] = {icon:"goblin",Race:"Goblin",Category:"Warrior",Stmn:13,Coor:12,
208 Psych:8,Renown:2,HGold:34,HGem:2,Food:0,WSRaw:2,BSRaw:2,Move:10,Room:0,
209 Armor:"",TotalAP:0,Weapn:"",WDam:0};
210 M[5] = {icon:"barbarian",Race:"Hobbit",Category:"Rogue",Stmn:9,Coor:13,
211 Psych:10,Renown:4,HGold:57,HGem:1,Food:4,WSRaw:2,BSRaw:3,Move:8,Room:0,
212 Armor:"",TotalAP:0,Weapn:"",WDam:0};
213 M[6] = {icon:"dwarf",Race:"Dwarf",Category:"Warrior",Stmn:15,Coor:10,Psych
214 :8,Renown:4,HGold:89,HGem:3,Food:1,WSRaw:3,BSRaw:2,Move:8,Room:0,Armor:"",
215 TotalAP:0,Weapn:"",WDam:0};
216 M[7] = {icon:"goblin",Race:"Goblin",Category:"Warrior",Stmn:13,Coor:11,
217 Psych:8,Renown:2,HGold:52,HGem:0,Food:0,WSRaw:2,BSRaw:2,Move:10,Room:0,
218 Armor:"",TotalAP:0,Weapn:"",WDam:0};
219 M[8] = {icon:"orc",Race:"Orc",Category:"Warrior",Stmn:11,Coor:13,Psych:8,
220 Renown:3,HGold:77,HGem:0,Food:1,WSRaw:3,BSRaw:2,Move:10,Room:0,Armor:"",
221 TotalAP:0,Weapn:"",WDam:0};
222 M[9] = {icon:"darkMage",Race:"Dark Mage",Category:"Mage",Stmn:12,Coor:12,
223 Psych:8,Renown:3,HGold:98,HGem:0,Food:2,WSRaw:2,BSRaw:3,Move:10,Room:0,
224 Armor:"",TotalAP:0,Weapn:"",WDam:0};
225 M[10] = {icon:"orc",Race:"Orc",Category:"Warrior",Stmn:13,Coor:11,Psych:8,
226 Renown:4,HGold:37,HGem:0,Food:1,WSRaw:3,BSRaw:3,Move:10,Room:0,Armor:"",
227 TotalAP:0,Weapn:"",WDam:0};
228 M[11] = {icon:"whiteMage",Race:"Goblin Mage",Category:"Mage",Stmn:10,Coor:
229 14,Psych:8,Renown:3,HGold:23,HGem:1,Food:1,WSRaw:3,BSRaw:2,Move:10,Room:0,
230 Armor:"",TotalAP:0,Weapn:"",WDam:0};
231 M[12] = {icon:"orc",Race:"Woodland Orc",Category:"Warrior",Stmn:11,Coor:13
232 ,Psych:8,Renown:4,HGold:93,HGem:1,Food:2,WSRaw:3,BSRaw:3,Move:10,Room:0,
233 Armor:"",TotalAP:0,Weapn:"",WDam:0};
234 M[13] = {icon:"goblin",Race:"Goblin",Category:"Warrior",Stmn:12,Coor:12,
235 Psych:8,Renown:3,HGold:76,HGem:0,Food:1,WSRaw:2,BSRaw:2,Move:10,Room:0,
236 Armor:"",TotalAP:0,Weapn:"",WDam:0};
237 M[14] = {icon:"whitePriest",Race:"Elf Priest",Category:"Rogue",Stmn:9,Coor
238 :17,Psych:10,Renown:6,HGold:97,HGem:2,Food:1,WSRaw:3,BSRaw:4,Move:15,Room:
239 0,Armor:"",TotalAP:0,Weapn:"",WDam:0};
240 M[15] = {icon:"dwarf",Race:"Dwarf Hero",Category:"Warrior",Stmn:13,Coor:13
241 ,Psych:9,Renown:6,HGold:103,HGem:4,Food:2,WSRaw:3,BSRaw:4,Move:8,Room:0,
242 Armor:"",TotalAP:0,Weapn:"",WDam:0};
243 M[16] = {icon:"goblin",Race:"Goblin Hero",Category:"Warrior",Stmn:11,Coor:
```

```

18, Psych:8, Renown:8, HGold:78, HGem:3, Food:1, WSRaw:3, BSRaw:3, Move:10, Room:0,
Armor:"", TotalAP:0, Weapn:"", WDam:0};
210 M[17] = {icon:"orc", Race:"Orc Hero", Category:"Warrior", Stmn:12, Coor:18,
Psych:9, Renown:8, HGold:92, HGem:2, Food:1, WSRaw:4, BSRaw:4, Move:10, Room:0,
Armor:"", TotalAP:0, Weapn:"", WDam:0};
211 M[18] = {icon:"hero2", Race:"Folks Hero", Category:"Warrior", Stmn:13, Coor:16
, Psych:8, Renown:7, HGold:87, HGem:1, Food:0, WSRaw:4, BSRaw:2, Move:10, Room:0,
Armor:"", TotalAP:0, Weapn:"", WDam:0};
212 M[19] = {icon:"mummy", Race:"Mummy", Category:"Undead", Stmn:9, Coor:11, Psych:
7, Renown:4, HGold:0, HGem:0, Food:0, WSRaw:2, BSRaw:2, Move:8, Room:0, Armor:"",
TotalAP:0, Weapn:"", WDam:0};
213 M[20] = {icon:"skeleton", Race:"Skeleton", Category:"Undead", Stmn:8, Coor:11,
Psych:4, Renown:2, HGold:0, HGem:0, Food:0, WSRaw:2, BSRaw:2, Move:8, Room:0, Armor
:"", TotalAP:0, Weapn:"", WDam:0};
214 M[21] = {icon:"skeleton", Race:"Skeleton", Category:"Undead", Stmn:8, Coor:11,
Psych:4, Renown:2, HGold:0, HGem:0, Food:0, WSRaw:2, BSRaw:2, Move:8, Room:0, Armor
:"", TotalAP:0, Weapn:"", WDam:0};
215 M[22] = {icon:"zombie", Race:"Zombie", Category:"Undead", Stmn:10, Coor:11,
Psych:6, Renown:3, HGold:0, HGem:0, Food:0, WSRaw:3, BSRaw:2, Move:10, Room:0,
Armor:"", TotalAP:0, Weapn:"", WDam:0};
216 M[23] = {icon:"zombie", Race:"Zombie", Category:"Undead", Stmn:10, Coor:11,
Psych:6, Renown:3, HGold:0, HGem:0, Food:0, WSRaw:3, BSRaw:2, Move:10, Room:0,
Armor:"", TotalAP:0, Weapn:"", WDam:0};
217 M[24] = {icon:"mummy", Race:"Mummy", Category:"Undead", Stmn:9, Coor:11, Psych:
7, Renown:4, HGold:0, HGem:0, Food:0, WSRaw:2, BSRaw:2, Move:8, Room:0, Armor:"",
TotalAP:0, Weapn:"", WDam:0};
218 M[25] = {icon:"skeleton", Race:"Skeleton", Category:"Undead", Stmn:8, Coor:11,
Psych:4, Renown:2, HGold:0, HGem:0, Food:0, WSRaw:2, BSRaw:2, Move:8, Room:0, Armor
:"", TotalAP:0, Weapn:"", WDam:0};
219 M[26] = {icon:"wyvern", Race:"Young Wyvern", Category:"Mage", Stmn:26, Coor:16
, Psych:14, Renown:12, HGold:890, HGem:14, Food:9, WSRaw:5, BSRaw:4, Move:20, Room:
0, Armor:"", TotalAP:4, Weapn:"", WDam:0};
220
221 //Movement Table; could be extenally loaded
222 /**TODO** Remote this static database onto database server; otherwise
upgrade to PouchDB or SQLite
223 var MT = [];
224 MT[0] = {RmID:'combat', North:0, East:0, South:0, West:0, Up:0, Dn:0, Tres
:0, Mnstr:0, Discovr:0, mxPos:280, myPos:150};
225 MT[1] = {RmID:'R1', North:0, East:0, South:2, West:1, Up:0, Dn:0, Tres:0,
Mnstr:0, Discovr:1, mxPos:280, myPos:150};
226 MT[2] = {RmID:'R2', North:1, East:3, South:3, West:0, Up:0, Dn:0, Tres:0,
Mnstr:0, Discovr:1, mxPos:280, myPos:150};
227 MT[3] = {RmID:'R3', North:2, East:5, South:0, West:2, Up:0, Dn:0, Tres:0,
Mnstr:0, Discovr:1, mxPos:235, myPos:230};
228 MT[4] = {RmID:'R4', North:0, East:0, South:5, West:0, Up:0, Dn:0, Tres:0,
Mnstr:0, Discovr:1, mxPos:220, myPos:160};
229 MT[5] = {RmID:'R5', North:4, East:0, South:0, West:3, Up:15, Dn:13, Tres:0
, Mnstr:0, Discovr:1, mxPos:200, myPos:230};
230 MT[6] = {RmID:'R6', North:0, East:1, South:0, West:0, Up:0, Dn:0, Tres:0,
Mnstr:0, Discovr:0, mxPos:200, myPos:260};
231 MT[7] = {RmID:'R7', North:0, East:0, South:8, West:0, Up:0, Dn:0, Tres:0,
Mnstr:0, Discovr:1, mxPos:220, myPos:240};
232 MT[8] = {RmID:'R8', North:7, East:0, South:10, West:0, Up:9, Dn:13, Tres:0
, Mnstr:0, Discovr:1, mxPos:150, myPos:260};
233 MT[9] = {RmID:'R9', North:0, East:0, South:19, West:0, Up:0, Dn:8, Tres:0,
Mnstr:0, Discovr:1, mxPos:200, myPos:190};
234 MT[10] = {RmID:'R10', North:8, East:11, South:0, West:0, Up:9, Dn:0, Tres:
0, Mnstr:0, Discovr:1, mxPos:265, myPos:215};

```

```
235 MT[11] = {RmID:'R11', North:0, East:0, South:0, West:10, Up:0, Dn:0, Tres:
0, Mnstr:0, Discovr:1, mxPos:450, myPos:290};
236 MT[12] = {RmID:'R12', North:0, East:0, South:0, West:13, Up:0, Dn:0, Tres:
0, Mnstr:0, Discovr:1, mxPos:210, myPos:210};
237 MT[13] = {RmID:'R13', North:0, East:12, South:0, West:0, Up:5, Dn:0, Tres:
0, Mnstr:0, Discovr:1, mxPos:165, myPos:190};
238 MT[14] = {RmID:'R14', North:0, East:17, South:15, West:0, Up:0, Dn:0, Tres
:0, Mnstr:0, Discovr:1, mxPos:175, myPos:150};
239 MT[15] = {RmID:'R15', North:14, East:0, South:0, West:0, Up:0, Dn:5, Tres:
0, Mnstr:0, Discovr:1, mxPos:175, myPos:190};
240 MT[16] = {RmID:'R16', North:17, East:0, South:0, West:19, Up:0, Dn:0, Tres
:0, Mnstr:0, Discovr:1, mxPos:190, myPos:220};
241 MT[17] = {RmID:'R17', North:18, East:0, South:16, West:14, Up:0, Dn:0,
Tres:0, Mnstr:0, Discovr:1, mxPos:235, myPos:240};
242 MT[18] = {RmID:'R18', North:0, East:0, South:17, West:0, Up:0, Dn:0, Tres:
0, Mnstr:0, Discovr:1, mxPos:200, myPos:170};
243 MT[19] = {RmID:'R19', North:9, East:16, South:0, West:0, Up:0, Dn:0, Tres:
0, Mnstr:0, Discovr:1, mxPos:165, myPos:190};
244 MT[20] = {RmID:'Inventory', North:0, East:0, South:0, West:0, Up:0, Dn:0,
Tres:0, Mnstr:0, Discovr:0, mxPos:1600, myPos:200};
245 // End of database structures
246 /**TODO**
247 // NOTE: the following data structures are for development only.
248 // The optimum usage is a local or remote database using PouchDB or
SQLite.
249 // Stay away from IndexedDB since it is deprecated.
250 // =====
251
252 /**TODO** Creating Game namespace called GAMEAPP; refactor GAMEAPP with
your project name
253 window.GAMEAPP = {
254     //US Copr. or Copyright; UTF8 circled c is \u00A9 equal to &copy;
255     Copr: "Copyright © \u00A9 1974-2016, Stephen Gose. All rights
reserved.\n",
256     // reference to the Phaser.Game instance
257     game: null,
258
259     // If there's music in your game, and it needs to play through-out a
260     // few State swaps, then you could reference it below.
261     music: null,
262
263     //Toggle background music theme on or off; starts in "on/true" state
264     musicToggle: true,
265
266     //Your game can check MYGAMEAPP.orientated in the game loops
267     // to know if it should pause or not.
268     //orientated: false,
269
270     /**
271     //Grid Tile-Map configurations
272     /**TODO** Remote this static database onto database server;
otherwise upgrade to PouchDB or SQLite
273     //Tile-maps are the visual express separate from the Movement Tables
metadata.
274     tileSize: 64, //twice the size of an avatar icon
275     numRows: 3, //adjustable for your game
276     numCols: 3, //adjustable for your game
277     tileSpacing: 2, //adjustable for your game
278     tilesArray: [], //one way; thousand more to choose
```



```

279     */
280     //Canvas dimensions: world and viewportHeight
281     /**TODO** adjust for your game deployment
282     viewportWidth: 550, //game view
283     viewportHeight: 440,
284     worldWidth: 550,      //world view
285     worldHeight: 440,
286
287     // =====
288     /**TODO** refactor and adjust for your game deployment
289     // Trash Global Variables (alphabetical)
290     // Here we have some global level vars that persist regardless of
291     // State.
292     // =====
293     CmbtTurn: 1,          // Combat Round counter
294     Cntr: 0,
295     CrntRoom: 6,         // Current Room occupied
296     DeadMonster: 0,
297     doorN:{},
298     doorE:{},
299     doorS:{},
300     doorW:{},
301     doorN2:{},
302     doorDn:{},
303     doorUp:{},
304     enemy:{},           //single player game; array used in MMOG
305     Entrance:{},       //Room #6 only
306     gameState: 'explore',
307     //dynamic toolTip across game states
308     /**TODO** store information text in a database for dynamic language
309     conversion.
310     InfoText: "Score: "+CCP.Score+"\n Click on the Door to enter.",
311     KeyNum: 0,
312     LastRoom: 6,        // Last Room entered, used to remove monsters
313     from current room
314     LastDoor:"East",    // help place avatar properly in new room entrance
315     Logout: "EXIT",     //exit game defeated.
316     movespeed: 10,     //default base movement points; this is not pixels.
317     mxPos: 200,        // Monster _x position
318     myPos: 260,        // Monster _y position
319     Narr1: 'doodah',    // Player Combat Narrative text
320     Narr2: "doodah",    // Monster Combat Narrative text
321     player:{},         //single player game; array used in MMOG
322     RoomAlert:"",      //HUD Room alert info
323     rect:{},
324     Trash: 0,
325     Treasure:{},       //treasure icon
326     //button styling
327     styleBTN: { font: "28px Arial", fill: "#000000", align: "center" },
328     //Combat HUD styling
329     // hero text
330     styleCmbtH: { font: "9px Arial", fill: "#66ffff", align: "center"
331     },
332     styleNarrH: { font: "9px Arial", fill: "#66ffff", align: "left" },
333     // monster text
334     styleCmbtM: { font: "9px Arial", fill: "#66ff66", align: "center" },
335     styleNarrM: { font: "9px Arial", fill: "#66ff66", align: "left" },
336     //HUD styling
337     styleHUD: { font: "14px Arial", fill: "#ff9900", align: "center" },

```



```

334 //Room Alert styling
335 styleRA: { font: "18px Arial", fill: "#ff9900", align: "right" },
336 //toolTip styling
337 styleTT: { font: "11px Arial", fill: "#ff9900", align: "center" },
338 TxPos: 0, // Treasure's default x position
339 TyPos: 0, // Treasure's default y position
340 xPos: 90, // Character default x position
341 yPos: 200, // Character default y position
342
343 // here we will store all game phase/states
344 // state object filled as js files load.
345 state: {},
346
347 // =====
348 // -----
349 // Main game Handler methods
350 // -----
351 //**TODO**:
352 // refactor and adjust for your game deployment
353 // remove console debug information on public deployment
354 // =====
355 main: function(){
356 this.game = new Phaser.Game(window.GAMEAPP.viewportWidth, window.
GAMEAPP.viewportHeight, Phaser.AUTO, document.body, window.GAMEAPP
.state.boot);
357 this._DevelopCR(); //create avatar combat record
358 //console.table(Person); //debug
359 console.log(' - Equip ALL Monsters in Database once per game. ');
360 this._MonsterPrep();
361 this._AllotMonster();
362
363 console.log(' - Creating internal static Monster Database: ');
364 console.table(M);
365
366 console.log(' - Creating internal static Movement Table:\n See
book for dynamically generated Movement Tables and Mazes. ');
367 console.log('Distributing 6 Treasures and 7 Monsters once per
game. ');
368 this._AllotTreasure();
369 console.table(MT);
370 },
371 // =====
372 // -----
373 // Supporting game Function & Classes
374 // -----
375 //**TODO**:
376 // Change namespace from generic GAMEAPP to your project
377 // refactor and adjust for your game deployment
378 // remove console debug information on public deployment
379 // =====
380 _audioMgr: function(mode, game) {
381 switch(mode) {
382 case 'init': {
383 GAMEAPP.Storage.initUnset('GAMEAPP.audio', true);
384 GAMEAPP._audioStatus = GAMEAPP.Storage.get('GAMEAPP.audio'
);
385 // GAMEAPP._soundClick = game.add.audio('audio-click');
386 GAMEAPP._sound = [];
387 GAMEAPP._sound['click'] = game.add.audio('audio-click');

```

```

388         if(!GAMEAPP._soundMusic) {
389             GAMEAPP._soundMusic = game.add.audio('audio-theme',1,
390                 true);
391             GAMEAPP._soundMusic.volume = 0.5;
392         }
393         break;
394     }
395     case 'on': {
396         GAMEAPP._audioStatus = true;
397         break;
398     }
399     case 'off': {
400         GAMEAPP._audioStatus = false;
401         break;
402     }
403     case 'switch': {
404         GAMEAPP._audioStatus =! GAMEAPP._audioStatus;
405         break;
406     }
407     default: {}
408 }
409 if(GAMEAPP._audioStatus) {
410     GAMEAPP._audioOffset = 0;
411     if(GAMEAPP._soundMusic) {
412         if(!GAMEAPP._soundMusic.isPlaying) {
413             GAMEAPP._soundMusic.play('',0,1,true);
414         }
415     }
416 }
417 else {
418     GAMEAPP._audioOffset = 4;
419     if(GAMEAPP._soundMusic) {
420         GAMEAPP._soundMusic.stop();
421     }
422 }
423 GAMEAPP.Storage.set('GAMEAPP.audio',GAMEAPP._audioStatus);
424 game.buttonAudio.setFrames(GAMEAPP._audioOffset+1, GAMEAPP.
425     _audioOffset+0, GAMEAPP._audioOffset+2);
426 },
427 //
428 // =====
429 _AllotTreasure: function() {
430     var RmNum = 2;
431     var Rm;
432     var Cntr = 1;
433     /**TODO**: 5 is a fixed value for demonstation game.
434     // Create and use a variable
435     while (Cntr<5) {
436         // This determines the Room Number
437         /**TODO**:
438         // Notice: using a fixed number here; this
439         // could be substituted with MT.length for dynamic
440         adjustments
441         RmNum = (Math.round(Math.random()*18))+1;
442         if (MT[RmNum].Tres == 0) {
443             var KeyNum = Math.round(Math.random()*100)+10;
444             MT[RmNum].Tres += KeyNum;
445             Cntr += 1;
446             console.log("Room #: "+MT[RmNum].RmID);

```

```

444 //             console.log("- Mnstr: #" + MT[RmNum].Mnstr);
445 //             console.log("- Tresr: $" + MT[RmNum].Tres);
446         }
447     }
448     // Place Treasure in the Private Meeting Room and Treasury
    regardless
449     /**TODO**
450     // Notice: using a fixed number here; this
451     // could be substituted with variable for dynamic adjustments
452     KeyNum = Math.round(Math.random()*100)+50;
453     MT[4].Tres += KeyNum;
454     KeyNum = Math.round(Math.random()*200);
455     MT[16].Tres += KeyNum;
456     // console.log("Room #: " + MT[4].RmID);
457     // console.log("- Mnstr: #" + MT[4].Mnstr);
458     // console.log("- Tresr: $" + MT[4].Tres);
459     // console.log("Room #: " + MT[16].RmID);
460     // console.log("- Mnstr: #" + MT[16].Mnstr);
461     // console.log("- Tresr: $" + MT[16].Tres);
462     // console.log("Treasures Allocated.");
463     console.log("=== Allocate Treasure completed: ===");
464
465 },
466 //
467 // =====
468 // Prep Frame 2
469 _AllotMonster: function() {
470     var RmNum = 1;
471     var Rm;
472     var Cntr = 1;
473     while (Cntr < 7) {
474         // This determines the Room Number
475         /**TODO**
476         // Notice: using a fixed number here; this
477         // could be substituted with MT.length for dynamic
            adjustments
478         RmNum = (Math.round(Math.random()*18))+1;
479
480         if (MT[RmNum].Mnstr == 0) {
481             //Monster static data file has 26 to choose from
482             /**TODO**
483             // Notice: using a fixed number here; this could be the
                variable M.length
484             // upgrade to Access a local PouchDB or SQLite database
485             // upgrade to Access a centralized remote database server
486             var KeyNum = (Math.round(Math.random()*25))+1;
487             MT[RmNum].Mnstr = KeyNum;
488             M[KeyNum].Room = RmNum;
489             Cntr += 1;
490             // console.log("Room #: " + MT[RmNum].RmID);
491             // console.log("- Mnstr: #" + MT[RmNum].Mnstr);
492             // console.log("- Tresr: $" + MT[RmNum].Tres);
493
494         }
495     }
496     // Place Monsters in the Private Meeting Room and Treasury
    regardless
497     //Forced Debug routine for room 1
498     //KeyNum = (Math.round(Math.random()*25))+1;

```

```

499         //MT[1].Mnstr = KeyNum;
500
501         /**TODO**:
```

// Notice: using a fixed number here; this
// could be substituted with variable for dynamic adjustments

```

502         KeyNum = (Math.round(Math.random()*25))+1;
503         MT[4].Mnstr = KeyNum;
504         M[KeyNum].Room = 4;
505         KeyNum = (Math.round(Math.random()*20))+6;
506         MT[16].Mnstr = KeyNum;
507         M[KeyNum].Room = 16;
508         //console.log("Room #: "+MT[1].RmID);
509         //console.log("- Mnstr: #" +MT[1].Mnstr);
510         //console.log("- Tresr: $" +MT[1].Tres);
511         // console.log("Room #: "+MT[4].RmID);
512         // console.log("- Mnstr: #" +MT[4].Mnstr);
513         // console.log("- Tresr: $" +MT[4].Tres);
514         // console.log("Room #: "+MT[16].RmID);
515         // console.log("- Mnstr: #" +MT[16].Mnstr);
516         // console.log("- Tresr: $" +MT[16].Tres);
517         // console.log("=== Monsters Allocated. ===");
518     },
519
520     //
521     //=====
522     //Prep Scene Frame 1
523     _CombatCheck: function(game){
524         //save current game progress when returning from combat state
525         GAMEAPP.LastRoom = GAMEAPP.CrntRoom;
526
527         console.log("Room #: "+GAMEAPP.CrntRoom+"; Opponent collision:
528         combat phase.");
529         this.camera.fade(0x000000, 200, false);
530         this.time.events.add(200, function() {
531             //ARRA Combat rv_3 through rv_8; switched statements used
532             this.game.state.start('Combat');
533         }, this);
534     },
535
536     //
537     //=====
538     // Prep Frame 4
539     //Develop Character Combat Record
540     _DevelopCR: function(){
541         //Character CombtRecord Initialized and UPdated prior to combat-
542         ONCE!!!
543         Person[1].TotalAP = Number(CCP.TotalAP);
544         Person[1].WS = ((Number(CCP.Stmn) * 2) + (Number(CCP.WSRaw) * 5));
545         Person[1].BS = ((Number(CCP.Coor) * 2) + (Number(CCP.BSRaw) * 5));
546         Person[1].PS = ((Number(CCP.Coor) * 2) + (Number(CCP.WSRaw) * 5));
547         Person[1].ModStmn = Number(CCP.ModStmn);
548         Person[1].Renown = Number(CCP.Renown);
549         //console.log("Char Stats:");
550         //console.log(" Total AP: " + Person[1].TotalAP);
551         //console.log(" WS: " + Person[1].WS);
552         //console.log(" BS: " + Person[1].BS);
553         //console.log(" PS: " + Person[1].PS);
554         //console.log(" ModStmn: " + Person[1].ModStmn);
555     }

```

```
556     },
557     //
558     // =====
559     _ExitGame: function() {
560         /**TODO**:
561         // redirect your gamers to your destination
562         window.open("https://leanpub.com/LoRD", "_blank");
563     },
564     // =====
565     //Main ARRA Entrance Door
566     _knockKnock: function(door) {
567
568         //debug
569         console.log("Room #: "+this.CrntRoom+"; Door Clicked: "+door.name)
570
571     },
572     //
573     // =====
574     _InventoryRoom: function(door) {
575
576         GAMEAPP.LastRoom = GAMEAPP.CrntRoom;
577         //debug
578         console.log("Room #: "+GAMEAPP.CrntRoom+"; Inventory Clicked: "+
579         door.name);
580         //visual sfx
581         this.camera.fade(0x000000, 200, false);
582         this.time.events.add(200, function() {
583             this.game.state.start('Inventory'); //ARRA Inventory
584             Scene rv_3 through rv_8
585         }, this);
586     },
587     //
588     // =====
589     //Prep Scene Frame 1
590     // Movement performed
591     // Monster moves in opposite direction of Char
592     _MoveChar: function (cursor,player) {
593         /**TODO**: adjust as required;
594         // more weight carried results in slower movement?
595         // modifications for race?
596         var speed = 250; //in pixels;
597         player.body.velocity.x = 0; //in pixels
598         player.body.velocity.y = 0; //in pixels
599         //monitor player's movement input
600         if (cursor.up.isDown){
601             player.body.velocity.y -= speed;
602         }
603         if (cursor.down.isDown){
604             player.body.velocity.y += speed;
605         }
606         if (cursor.right.isDown){
607             player.body.velocity.x += speed;
608         }
609         if (cursor.left.isDown){
610             player.body.velocity.x -= speed;
611         }
612     },
```

```
613     //
614     // =====
615     // Prep Frame 4
616     _MonsterPrep: function(game){
617         /**TODO**
618         // Notice: using a fixed number here; this
619         //      could be substituted with variable for dynamic adjustments
620         //Equip Monster ONCE per game
621         for(var i=1; i<26; i++){
622             var Trash = (Math.round(Math.random()*6))+1;
623             //Random Armor Names and TotalAP protection as deductions from
             damage
624             if (Trash == 1){
625                 M[i].Armor = "Small Shield";
626                 M[i].TotalAP += 1;
627             }
628             if ((Trash == 2) || (Trash == 0)){
629                 M[i].Armor = "Leather Jerkin";
630                 M[i].TotalAP += 2;
631             }
632             if (Trash == 3){
633                 M[i].Armor = "Shield & Leather Jerkin";
634                 M[i].TotalAP += 3;
635             }
636             if (Trash == 4){
637                 M[i].Armor = "Ring Mail";
638                 M[i].TotalAP += 3;
639             }
640             if (Trash == 5){
641                 M[i].Armor = "Chain Mail";
642                 M[i].TotalAP += 3;
643             }
644             if (Trash == 6){
645                 M[i].Armor = "Ring Mail & Shield";
646                 M[i].TotalAP += 4;
647             }
648
649             Trash = (Math.round(Math.random()*6))+1;
650             //Assign Random Weapons; Weapon Damage is the number of
             6-sided dice.
651             if (Trash == 1){
652                 M[i].Weapn = "Small Ax";
653                 M[i].WDam += 1;
654             }
655             if ((Trash == 2) || (Trash == 0)){
656                 M[i].Weapn = "Short Sword";
657                 M[i].WDam += 2;
658             }
659             if (Trash == 3){
660                 M[i].Weapn = "Mace";
661                 M[i].WDam += 2;
662             }
663             if (Trash == 4){
664                 M[i].Weapn = "Cutlass";
665                 M[i].WDam += 2;
666             }
667             if (Trash == 5){
668                 M[i].Weapn = "Long Bow & Sh. Sword";
669                 M[i].WDam += 2;
```

```
670     }
671     if (Trash == 6){
672         M[i].Weapn = "Short Bow & Ax";
673         M[i].WDam += 1;
674     }
675     if(i == 26){
676         //Set Wyvern Armor
677         M[26].Armor = "Scales";
678         M[26].TotalAP = 4;
679         //Set Wyvern natural weapons
680         M[26].Weapn = "Fangs & Claws";
681         M[26].WDam +=3;
682     }
683     }
684 }
685
686 },
687 //
688 //=====
689 //Prep Scene Frame 1
690 _MoveMonster: function (cursor,monster){
691     /**TODO**: adjust as required;
692     // more weight carred results in slower movement?
693     // modifications for race?
694     var speed = 250; //in pixels
695     monster.body.velocity.x = 0; //in pixels
696     monster.body.velocity.y = 0; //in pixels
697     if (cursor.up.isDown){
698         monster.body.velocity.y += speed;
699     }
700     if (cursor.down.isDown){
701         monster.body.velocity.y -= speed;
702     }
703     if (cursor.right.isDown){
704         monster.body.velocity.x -= speed;
705     }
706     if (cursor.left.isDown){
707         monster.body.velocity.x += speed;
708     }
709 }
710 },
711 //
712 // =====
713 _newGame: function(){
714     /**TODO**: adjust as required
715     // redirect gamers to your site for new game
716     window.open("http://adventurers-of-renown.com/GAMEAPP/", "_blank");
717 },
718 //
719 // =====
720 _playAudio: function(sound) {
721     if(GAMEAPP._audioStatus) {
722         if(GAMEAPP._sound && GAMEAPP._sound[sound]) {
723             GAMEAPP._sound[sound].play();
724         }
725     }
726 },
727 //
728 // =====
```



```

729     _PostMonster: function (game,enemy){
730
731         //create an opponent; Check for monster in this room
732         if(MT[this.CrntRoom].Mnstr > 0){
733
734             this.enemy = this.game.add.sprite(MT[this.CrntRoom].mxPos,
735             MT[this.CrntRoom].myPos, 'avatar');
736             this.enemy.frameName = String(M[MT[this.CrntRoom].Mnstr].
737             icon+'.bmp');
738             this.enemy.body.collideWorldBounds = true;
739             this.enemy.enableBody = true;
740             this.enemy.body.immovable = true;
741             this.game.physics.enable(this.enemy, Phaser.Physics.ARCADE
742             );
743
744             this._PUpdate();
745
746         }
747     },
748     //
749     // =====
750     _PostTres: function (game,Treasure){
751
752         if(MT[this.CrntRoom].Tres > 0){
753             /**TODO**: store information text in a database for dynamic
754             language conversion.
755             InfoText = "There treasure here! See it?!!\n Score: " + CCP.
756             Score;
757             this.Treasure = game.add.sprite(0,0,'avatar');
758             this.Treasure.frameName = 'treasure.bmp';
759             this.Treasure.body.collideWorldBounds = true;
760
761             //Place Treasure Chest icon mid-way between
762             this.Treasure.x = Number((xPos + MT[this.CrntRoom].mxPos)/2);
763             this.Treasure.y = Number((yPos + MT[this.CrntRoom].myPos)/2);
764         }
765     },
766     //
767     // =====
768     // Prep Frame 3: Player Updates Combat Record to current stats
769     /**TODO**: adjust as required to deploy array index 1-4 are
770     player(s); 5-8 are antagonists but could be controlled as
771     // single player for Multi-avatar teams; OR
772     // multi-player single avatar controlled
773     _PUpdate: function (){
774         var CTurnNum = 0;
775         GAMEAPP.CmbtTurn = 0;
776         GAMEAPP.Narr1 = "";
777         GAMEAPP.Narr2 = "";
778         //Character Updated ONCE after Person Combat Record initialized.
779         Person[1].TotalAP = Number(CCP.TotalAP);
780         Person[1].WS = ((Number(CCP.Stmn) * 2) + (Number(CCP.WSRaw) * 5));
781         Person[1].BS = ((Number(CCP.Coar) * 2) + (Number(CCP.BSRaw) * 5));
782         Person[1].PS = ((Number(CCP.Coar) * 2) + (Number(CCP.WSRaw) * 5));
783         Person[1].ModStmn = Number(CCP.ModStmn);
784         Person[1].Renown = Number(CCP.Renown);
785         // console.log("Char Stats Re-read:");
786         // console.log(" Total AP: " + Person[1].TotalAP);

```

```

782 // console.log("      WS: " + Person[1].WS);
783 // console.log("      BS: " + Person[1].BS);
784 // console.log("      PS: " + Person[1].PS);
785 // console.log(" ModStmn: " + Person[1].ModStmn);
786
787 //Monster Updates
788 //setting Computer controlled antagonist;
789 Person[6].TotalAP = (M[MT[this.CrntRoom].Mnstr].TotalAP);
790 Person[6].Coor = Number(M[MT[this.CrntRoom].Mnstr].Coor);
791 Person[6].Food = Number(M[MT[this.CrntRoom].Mnstr].Food);
792 Person[6].Gold = Number(M[MT[this.CrntRoom].Mnstr].HGold);
793 Person[6].Gem = Number(M[MT[this.CrntRoom].Mnstr].HGem);
794 Person[6].WS = ((Number(M[MT[this.CrntRoom].Mnstr].Stmn) * 2) + (
Number(M[MT[this.CrntRoom].Mnstr].WSRaw) * 5));
795 Person[6].BS = ((Number(M[MT[this.CrntRoom].Mnstr].Coor) * 2) + (
Number(M[MT[this.CrntRoom].Mnstr].BSRaw) * 5));
796 Person[6].PS = ((Number(M[MT[this.CrntRoom].Mnstr].Coor) * 2) + (
Number(M[MT[this.CrntRoom].Mnstr].WSRaw) * 5));
797 Person[6].ModStmn = Number(M[MT[this.CrntRoom].Mnstr].Stmn);
798 Person[6].Renown = Number(M[MT[this.CrntRoom].Mnstr].Renown);
799 // console.log("Monster Stats:");
800 // console.log(" Total AP: " + Person[6].TotalAP);
801 // console.log("      WS: " + Person[6].WS);
802 // console.log("      BS: " + Person[6].BS);
803 // console.log("      PS: " + Person[6].PS);
804 // console.log(" ModStmn: " + Person[6].ModStmn);
805 //debug
806 console.table(Person);
807 },
808 //
809 //=====
810 _RenownCheck: function (game){
811 //**TODO**: store information text in a database for dynamic
language conversion.
812 //is a monster in this room?
813 if (MT[this.CrntRoom].Mnstr > 0){
814 //Ran away while monster was still living in the room.
815 CCP.Renown -= 1;
816 GAMEAPP.InfoText = "Score :"+CCP.TempScore+"\n Running from
Combat? \n Renown is " + CCP.Renown;
817 }
818 },
819 //
820 // =====
821 _RmAdmin: function (CrntRoom){
822
823 //add to avatar experience for Room discovery
824 CCP.Score += Number(MT[this.CrntRoom].Discover);
825 MT[this.CrntRoom].Discover = 0;
826
827 },
828 //
829 // =====
830 //Prep Scene Frame 1
831 //**TODO**: store information in a database for dynamic avatar
inventory.
832 _WeaponClass: function (w1, w2, w3, w4, w5, w6, w7, w8, w9, w10, w11,
w12, w13, w14) {
833 this.WID = w1;

```

```

834     this.Name = w2;
835     this.DamPossible = w3;
836     this.DamMod = w4;
837     this.Cost = w5;
838     this.Worth = w6;
839     // New property. Current armor value to sell
840     this.Weight = w7;
841     this.StmnReq = w8;
842     this.CoorReq = w9;
843     this.Use = w10;
844     // 1=One handed; 2=Both hands
845     this.Mode = w11;
846     // M=Melee;Missile=M;T=Thrown: "MMT"
847     this.DamType = w12;
848     // Chop=C;Slash=S;Smash=S;Thrust=T: "CSST"
849     this.Magic = w13;
850     this.Renown = w14;
851     this.DamageDone = this._GetWpnDamage();
852     // Damage Delt per call
853     // Class Inherited Methods:
854 }
855 //
856 // =====
857 // -----
858 // End Main game Handler
859 // -----
860 // =====
861
862 /**
863 //TODO: integration
864 _labelButton: function(game,x,y,key,label,callback,
865     callbackContext, overFrame, outFrame, downFrame, upFrame){
866     Phaser.Button.call(this,game,x,y,key,callback,
867     callbackContext, overFrame, outFrame, downFrame, upFrame);
868     //Style how you wish...
869     this.style = {'font': '10px Arial','fill':'black'};
870     this.anchor.setTo( 0.5, 0.5 );
871     this.label = new Phaser.Text(game, 0, 0, label, this.style);
872     //puts the label in the center of the button
873     this.label.anchor.setTo( 0.5, 0.5 );
874     this.addChild(this.label);
875     this.setLabel( label );
876     //adds button to game
877     game.add.existing( this );
878 }
879 _labelButton.prototype = Object.create(Phaser.Button.prototype);
880 _labelButton.prototype.constructor = _labelButton;
881 _labelButton.prototype.setLabel = function( label ) {
882     this.label.setText(label);
883 }
884 */
885
886 };
887
888 /** DEPRECATED METHOD - NEVER EVER USE THIS!
889 * See Phaser.js Game Design Workbook for complete explanation
890 * http://leanpub.com/phaserjsgamedesignworkbook
891 * window.onload = function () {
892 *     let game = new Phaser.Game(0, 0, Phaser.AUTO, document.body);

```

```
893     *   };
894     */
895
896     //preferred lauch method for BOM.
897     window.addEventListener('DOMContentLoaded', function(){
898         window.GAMEAPP.main();
899     }, false);
900
901     /* End of file */
902     /* Location: ./js/main.js */
903
```