Subject: Re: A notice to anyone planning to write code for scripts 4.0 Posted by saberhawk on Thu, 01 Sep 2011 05:46:56 GMT View Forum Message <> Reply to Message

The color scheme can be messed around with significantly.

Colors!

File Attachments
1) colors.png, downloaded 323 times

Page 1 of 2 ---- Generated from Command and Conquer: Renegade Official Forums

ee tt_VC2010 - Microsoft Visual Studio
<u>F</u> ile <u>E</u> dit <u>V</u> iew VAssist <u>X</u> The <u>m</u> e <u>P</u> roject <u>B</u> uild <u>D</u> ebug Tea <u>m</u> D <u>a</u> ta <u>T</u> ools Ar <u>c</u> hitecture Te <u>s</u> t A <u>n</u> alyze <u>W</u> indow <u>H</u> elp
h PhysDefClass.h ×
(Global Scope) -
4 #include "engine_common.h"
5 #include "engine_vector.h"
6 #include "engine_io.h"
7 <pre>#include "engine_threading.h"</pre>
8 #include "engine_string.h"
9 #include "Definition.h"
10 // If you modify this list, be sure to add the friendly name to GetCollisionGroupName
11 ⊟typedef enum {
12 DEFAULT_COLLISION_GROUP == 0, = // collides with everything
13 UNCOLLIDEABLE_GROUP, // collides with nothing
14 TERRAIN_ONLY_COLLISION_GROUP, TS BULLET COLLISION GROUP, TABLE AND
15 > BULLET_COLLISION_GROUP, ++ > > // collides with everything but itself 16 > TERRAIN_AND_BULLET_COLLISION_GROUP, +> > // collides with terrain and bullets
17 + BULLET_ONLY_COLLISION_GROUP, + + + // collides only with bullets
18 SOLDIER_COLLISION_GROUP, 3 3 3 // collides with everything (but only soldiers u
19 SOLDIER_GHOST_COLLISION_GROUP, A A A // collides with everything but soldiers
20
21 ÷ // new collision groups here. 8 - 14
22 -> C4 COLLISION GROUP, ++ + + + + -> // behaves like DEFAULT but doesn't collide with
23 UNDERGROUND_COLLISION_GROUP, Y Y Collides with TERRAIN and itself
23 > UNDERGROUND_COLLISION_GROUP, > // collides with TERRAIN and itself 24 > SOLDIER_ONLY_COLLISION_GROUP, > // collides only with SOLDIER and SOLDIER_GHOST
25 → SOLDIER_BULLET_COLLISION_GROUP, → → // collides with C4, BULLET, SOLDIER and SOLDIER
26
27 → TERRAIN_COLLISION_GROUP → → = 15, → → // Terrain must be 15
28
29 > // more collision groups here, 16 - 31
30 * WATER_SURFACE_COLLISION_GROUP, * * * // collides with NAVAL_UNIT, BEACHING_UNIT, AMBH
31 * WATER_EDGE_COLLISION_GROUP, ** * * // collides with DEFAULT, SOLDIER, and NAVAL_UNI 32 * WATER_EDGE_ALT_COLLISION_GROUP, ** * // collides only with NAVAL_UNIT
32 WATER_EDGE_ALT_COLLISION_GROUP, // collides only with NAVAL_UNIT 33 BEACH EDGE COLLISION GROUP, // collides only with BEACHING UNIT
24 NAVAL UNIT COLLISION GROUP AS A A A A A A A A A A A A A A A A A A
34 NAVAL_UNIT_COLLISION_GROUP, // behaves like DEFAULT but collides with WATER_ 35 BEACHING_UNIT_COLLISION_GROUP, // behaves like DEFAULT but collides with WATER_
36 HOVER_UNIT_COLLISION_GROUP, 4 // behaves like DEFAULT but collides with WATER_
37 AMPHIBIOUS_UNIT_COLLISION_GROUP, // behaves like DEFAULT but collides with AMPHIB
38 AMPHIBIOUS UNIT FLOOR COLLISION GROUP, A // collides only with AMPHIBIOUS UNIT
39
40 COLLISION_GROUP_MAX = 32, A A A A A A A A A A A A A A A A A A A
41 } Collision_Group_Type;
42
43 inline const char* GetCollisionGroupName(Collision_Group_Type group)
34 WAVAL_UNIT_COLLISION_GROUP, // behaves like DEFAULT but collides with WATER_ 35 BEACHING_UNIT_COLLISION_GROUP, // behaves like DEFAULT but collides with WATER_ 36 HOVER_UNIT_COLLISION_GROUP, // behaves like DEFAULT but collides with WATER_ 37 AMPHIBIOUS_UNIT_COLLISION_GROUP, // behaves like DEFAULT but collides with AMPHIB 38 AMPHIBIOUS_UNIT_FLOOR_COLLISION_GROUP, // collides only with AMPHIBIOUS_UNIT 39
Ready Ln 10 Col 86