Code Out of Game3d.cs

Enemy Classes

class cCritterEnemyOne :cCritter3Dcharacter

{

System.Timers.Timer deathAnimation;

int elapsed;

Random rndRip;

int ripState;

public cCritterEnemyOne(cGame pownergame)

: base( pownergame )

{

deathAnimation = new System.Timers.Timer();

elapsed = 0;

rndRip = new Random();

addForce(new cForceGravity(25.0f, new cVector3(0.0f, -1, 0.00f)));

addForce(new cForceDrag(20.0f)); // default friction strength 0.5

Density = 2.0f;

MaxSpeed = 30.0f;

if (pownergame != null) //Just to be safe.

Sprite = new cSpriteQuake(ModelsMD2.Sorb);

// example of setting a specific model

// setSprite(new cSpriteQuake(ModelsMD2.Knight));

if (Sprite.IsKindOf("cSpriteQuake")) //Don't let the figurines tumble.

{

AttitudeToMotionLock = false;

Attitude = new cMatrix3(new cVector3(0.0f, 0.0f, 1.0f),

new cVector3(1.0f, 0.0f, 0.0f),

new cVector3(0.0f, 1.0f, 0.0f), Position);

/\* Orient them so they are facing towards positive Z with heads towards Y. \*/

}

Bounciness = 0.0f; //Not 1.0 means it loses a bit of energy with each bounce.

//Boss is 3.0f, Room 1 is 2.0f, Room 2 is 1.5f

setRadius(1.3f);

MinTwitchThresholdSpeed = 4.0f; //Means sprite doesn't switch direction unless it's moving fast

randomizePosition(new cRealBox3(new cVector3(\_movebox.Lox, \_movebox.Loy, \_movebox.Loz + 4.0f),

new cVector3(\_movebox.Hix, \_movebox.Loy, \_movebox.Midz - 1.0f)));

/\* I put them ahead of the player \*/

randomizeVelocity(0.0f, 30.0f, false);

if (pownergame != null) //Then we know we added this to a game so pplayer() is valid

addForce(new cForceObjectSeek(Player, 0.5f));

int begf = Framework.randomOb.random(0, 171);

int endf = Framework.randomOb.random(0, 171);

if (begf > endf)

{

int temp = begf;

begf = endf;

endf = temp;

}

Sprite.setstate(State.Other, begf, endf, StateType.Repeat);

\_wrapflag = cCritter.BOUNCE;

}

public override void update(ACView pactiveview, float dt)

{

base.update(pactiveview, dt); //Always call this first

//if ( (\_outcode & cRealBox3.BOX\_HIZ) != 0 ) /\* use bitwise AND to check if a flag is set. \*/

//delete\_me(); //tell the game to remove yourself if you fall up to the hiz.

if(distanceTo(Player)<=15)

{

addForce(new cForceObjectSeek(Player, 0.5f));

}

}

public override void die()

{

ripState = rndRip.Next(0, 2);

deathAnimation.Interval = 100;

if (ripState == 0)

Sprite.setstate(State.FallbackDie, 0, 0, StateType.Hold);

else if (ripState == 1)

Sprite.setstate(State.FallForwardDie, 0, 0, StateType.Hold);

deathAnimation.Start();

deathAnimation.Elapsed += new System.Timers.ElapsedEventHandler(interval\_Tick);

Player.addScore(Value);

//base.die();

((cGame3D)Game).decrementMonsterCount();

}

private void interval\_Tick(object sender, EventArgs e)

{

elapsed++;

if (elapsed % 5 == 0)

{

Sprite.ModelState = State.Idle;

elapsed = 0;

deathAnimation.Stop();

base.die();

}

}

public override bool IsKindOf(string str)

{

return str == "cCritter3Dcharacter" || base.IsKindOf(str);

}

public override string RuntimeClass

{

get

{

return "cCritter3Dcharacter";

}

}

}

class cCritterEnemyTwo : cCritterArmed

{

System.Timers.Timer deathAnimation;

int elapsed;

Random rndRip;

int ripState;

public cCritterEnemyTwo(cGame pownergame)

: base(pownergame)

{

deathAnimation = new System.Timers.Timer();

elapsed = 0;

rndRip = new Random();

addForce(new cForceGravity(25.0f, new cVector3(0.0f, -1, 0.00f)));

addForce(new cForceDrag(20.0f)); // default friction strength 0.5

Health = 2;

\_bshooting = false;

WaitShoot = 15;

Armed = true;

Density = 1.0f;

MaxSpeed = 22.0f;

if (pownergame != null) //Just to be safe.

Sprite = new cSpriteQuake(ModelsMD2.Ranger);

// example of setting a specific model

// setSprite(new cSpriteQuake(ModelsMD2.Knight));

if (Sprite.IsKindOf("cSpriteQuake")) //Don't let the figurines tumble.

{

AttitudeToMotionLock = false;

Attitude = new cMatrix3(new cVector3(0.0f, 0.0f, 1.0f),

new cVector3(1.0f, 0.0f, 0.0f),

new cVector3(0.0f, 1.0f, 0.0f), Position);

/\* Orient them so they are facing towards positive Z with heads towards Y. \*/

}

Bounciness = 0.0f; //Not 1.0 means it loses a bit of energy with each bounce.

//Boss is 3.0f, Room 1 is 2.0f, Room 2 is 1.5f

setRadius(1.5f);

MinTwitchThresholdSpeed = 4.0f; //Means sprite doesn't switch direction unless it's moving fast

randomizePosition(new cRealBox3(new cVector3(\_movebox.Lox, \_movebox.Loy, \_movebox.Loz + 4.0f),

new cVector3(\_movebox.Hix, \_movebox.Loy, \_movebox.Midz - 1.0f)));

/\* I put them ahead of the player \*/

randomizeVelocity(0.0f, 30.0f, false);

if (pownergame != null) //Then we know we added this to a game so pplayer() is valid

addForce(new cForceObjectSeek(Player, 0.5f));

int begf = Framework.randomOb.random(0, 171);

int endf = Framework.randomOb.random(0, 171);

if (begf > endf)

{

int temp = begf;

begf = endf;

endf = temp;

}

Sprite.setstate(State.Other, begf, endf, StateType.Repeat);

\_wrapflag = cCritter.BOUNCE;

}

public override void update(ACView pactiveview, float dt)

{

base.update(pactiveview, dt); //Always call this first

//if ( (\_outcode & cRealBox3.BOX\_HIZ) != 0 ) /\* use bitwise AND to check if a flag is set. \*/

//delete\_me(); //tell the game to remove yourself if you fall up to the hiz.

if (distanceTo(Player) <= 16)

{

addForce(new cForceObjectSeek(Player, 0.9f));

}

if(distanceTo(Player)>10)

{

\_bshooting = true;

}

}

// do a delete\_me if you hit the left end

public override void die()

{

ripState = rndRip.Next(0, 2);

deathAnimation.Interval = 100;

if (ripState == 0)

Sprite.setstate(State.FallbackDie, 0, 0, StateType.Hold);

else if (ripState == 1)

Sprite.setstate(State.FallForwardDie, 0, 0, StateType.Hold);

deathAnimation.Start();

deathAnimation.Elapsed += new System.Timers.ElapsedEventHandler(interval\_Tick);

Player.addScore(Value);

//base.die();

((cGame3D)Game).decrementMonsterCount();

}

private void interval\_Tick(object sender, EventArgs e)

{

elapsed++;

if (elapsed % 5 == 0)

{

Sprite.ModelState = State.Idle;

elapsed = 0;

deathAnimation.Stop();

base.die();

}

}

public override bool IsKindOf(string str)

{

return str == "cCritter3Dcharacter" || base.IsKindOf(str);

}

public override string RuntimeClass

{

get

{

return "cCritter3Dcharacter";

}

}

}

class cCritterEnemyBoss : cCritterArmed

{

System.Timers.Timer deathAnimation;

int elapsed;

Random rndRip;

int ripState;

public cCritterEnemyBoss(cGame pownergame)

: base(pownergame)

{

deathAnimation = new System.Timers.Timer();

elapsed = 0;

rndRip = new Random();

addForce(new cForceGravity(25.0f, new cVector3(0.0f, -1, 0.00f)));

addForce(new cForceDrag(20.0f)); // default friction strength 0.5

Density = 2.0f;

Health = 5;

Armed = true;

MaxSpeed = 30.0f;

if (pownergame != null) //Just to be safe.

Sprite = new cSpriteQuake(ModelsMD2.Tyrant);

// example of setting a specific model

// setSprite(new cSpriteQuake(ModelsMD2.Knight));

if (Sprite.IsKindOf("cSpriteQuake")) //Don't let the figurines tumble.

{

AttitudeToMotionLock = false;

Attitude = new cMatrix3(new cVector3(0.0f, 0.0f, 1.0f),

new cVector3(1.0f, 0.0f, 0.0f),

new cVector3(0.0f, 1.0f, 0.0f), Position);

/\* Orient them so they are facing towards positive Z with heads towards Y. \*/

}

Bounciness = 0.0f; //Not 1.0 means it loses a bit of energy with each bounce.

//Boss is 3.0f, Room 1 is 2.0f, Room 2 is 1.5f

setRadius(2.8f);

MinTwitchThresholdSpeed = 4.0f; //Means sprite doesn't switch direction unless it's moving fast

randomizePosition(new cRealBox3(new cVector3(\_movebox.Lox, \_movebox.Loy, \_movebox.Loz + 4.0f),

new cVector3(\_movebox.Hix, \_movebox.Loy, \_movebox.Midz - 1.0f)));

/\* I put them ahead of the player \*/

randomizeVelocity(0.0f, 30.0f, false);

if (pownergame != null) //Then we know we added this to a game so pplayer() is valid

addForce(new cForceObjectSeek(Player, 0.5f));

int begf = Framework.randomOb.random(0, 171);

int endf = Framework.randomOb.random(0, 171);

if (begf > endf)

{

int temp = begf;

begf = endf;

endf = temp;

}

Sprite.setstate(State.Other, begf, endf, StateType.Repeat);

\_wrapflag = cCritter.BOUNCE;

}

public override void update(ACView pactiveview, float dt)

{

base.update(pactiveview, dt); //Always call this first

//if ( (\_outcode & cRealBox3.BOX\_HIZ) != 0 ) /\* use bitwise AND to check if a flag is set. \*/

//delete\_me(); //tell the game to remove yourself if you fall up to the hiz.

if (distanceTo(Player)<=27)

{

addForce(new cForceObjectSeek(Player, 0.4f));

}

}

// do a delete\_me if you hit the left end

public override void die()

{

ripState = rndRip.Next(0, 2);

deathAnimation.Interval = 100;

if (ripState == 0)

Sprite.setstate(State.FallbackDie, 0, 0, StateType.Hold);

else if (ripState == 1)

Sprite.setstate(State.FallForwardDie, 0, 0, StateType.Hold);

deathAnimation.Start();

deathAnimation.Elapsed += new System.Timers.ElapsedEventHandler(interval\_Tick);

Player.addScore(Value);

//base.die();

((cGame3D)Game).decrementMonsterCount();

((cGame3D)Game).setEndGame();

}

public int getHealth()

{

return Health;

}

private void interval\_Tick(object sender, EventArgs e)

{

elapsed++;

if (elapsed % 10 == 0)

{

Sprite.ModelState = State.Idle;

elapsed = 0;

deathAnimation.Stop();

base.die();

}

}

public override bool IsKindOf(string str)

{

return str == "cCritter3Dcharacter" || base.IsKindOf(str);

}

public override string RuntimeClass

{

get

{

return "cCritter3Dcharacter";

}

}