

Welcome to creating mini games with 3D Game studio A8.

Written by : René Pol 31-07-2017©

More more more to do with issue number 5!

Welcome back. This issue we will add some more stuff to our mini game. We begin with placing some more enemies. In this example the game will have 2 enemies. We know how to place them at game start right ?

```
function game_start()
{
set(start_pan,SHOW);
snd_play(start_snd,100,0);
wait(-1);
ent_create("sid.mdl", vector(1.597,0, 19.507),walking_sid);
ent_create("shyguy.mdl", vector(205.857,160.444, 11.476),bouncing_enemy);
ent_create("shyguy.mdl", vector(105.857,160.444, 11.476),bouncing_enemy);
countdown();
reset(start_pan,SHOW);
}
```

You will notice that every time the enemies bump into each other they will play the hit sound and 10 points get subtracted from the score.

To prevent this we will give our player model a skill. In the enemy collide function we script it so only when players skill is hit the sound must play as well as the subtraction of the 10 points.

```
action walking_sid()
{
set(my,SHADOW);
sid = me;
c_setminmax(me);
var standing_percentage;
var walking_percentage;
var jumping_percentage;
on_space = my_jump;
my.skill1 =200;
while(1)
{
my_gravity();
while(key_space)
{
if(jumping ==0){
snd_play(jump_snd,100,0);
jumping =1;
}
ent_animate(my, "jump", jumping_percentage, 0);
jumping_percentage += 4 * time_step;
c_move (my, vector (10 * time_step, 0, absdist.z), nullvector, IGNORE_PASSABLE | IGNORE_ME | GLIDE);
wait(1);
}
if(key_cul){
my.pan =90;
}
if(key_cur){
my.pan =-90;
}
if(key_cud){
my.pan =-180;
}
if(key_cuu){
my.pan =0;
}
if(!key_cuu - key_cud - key_cul - key_cur)
{
standing_percentage %= 100;
standing_percentage += 3 * time_step;
```

Welcome to creating mini games with 3D Game studio A8.

Written by : René Pol 31-07-2017©

```
ent_animate(my, "stand", standing_percentage, ANM_CYCLE);
c_move (my, vector (0 * time_step, 0, absdist.z), nullvector, IGNORE_PASSABLE | IGNORE_ME | GLIDE);
wait(1);
}
else
{
walking_percentage %= 100;
walking_percentage += 10 * time_step;
ent_animate(my, "walk", walking_percentage, ANM_CYCLE);
c_move (my, vector (10 * time_step, 0, absdist.z), nullvector, IGNORE_PASSABLE | IGNORE_ME | GLIDE);
wait(1);
}
}
```

The enemy collision function is changed so it only works if player gets hit.

```
function enemy_collides()
{
if(event_type == EVENT_BLOCK)
{
vec_to_angle(my.pan,bounce);
my.pan += 10 - random(20);
my.tilt = 0;
}
if(event_type == EVENT_ENTITY)
{
if(you.skill1 ==200)
snd_play(auch_snd,100,0);
vec_to_angle(my.pan,bounce);
my.pan += 40 - random(20);
my.tilt = 0;
if(score >0)
score -=10;
}
}
```

Voila ! Problem solved you can now add as many enemies if you want without any problems.



Adding music to your games.

What would gaming be without nice musical pieces to go with it ? In order to handle music in games we use variables. This way we can make the music play,loop,stop and pause whenever we wanted.

I created a nice in game tune called ingame.mp3. Let's make good use of this.

First we create a variable for the music track.

```
var = soundtrack01_handle;
```

In the main function we make it play in a loop by using media_loop and the variable we created.

```
function main()
{
video_set(1024,768,32,0);
sky_color.red = 0;
sky_color.green = 0;
sky_color.blue = 0;
level_load("BMS.WMB");
mouse_map = pointer_png;
mouse_mode = 3;
soundtrack01_handle = media_loop("ingame.mp3",NULL,100);
}
```

Now when the game starts the music will play in a loop right away until we want it to stop. In this game when the timer hits zero we make the soundtrack stop by using its variable.

```
function game_finish()
{
media_stop(soundtrack01_handle);
set(finish_pan,SHOW);
snd_play(finish_snd,100,0);
set(sid,PASSABLE | INVISIBLE);
sid.event = NULL;
wait(1);
ent_remove(sid);
wait(-1);
reset(finish_pan,SHOW);
game_begin = 0;
set(menu_pan,SHOW);
mouse_mode = 3;
}
```

To make it more interesting we will make a finished sound play. We define the sound file to use.

```
SOUND* finito_snd = "finito.wav";
```

We add new lines to the game_finish function so the sound will be played.

```
function game_finish()
{
media_stop(soundtrack01_handle);
set(finish_pan,SHOW);
snd_play(finish_snd,100,0);
set(sid,PASSABLE | INVISIBLE);
sid.event = NULL;
wait(-1);
snd_play(finito_snd,100,0);
ent_remove(sid);
wait(-1);
reset(finish_pan,SHOW);
game_begin =0;
wait(-2.5);
soundtrack01_handle = media_loop("ingame.mp3",NULL,100);
set(menu_pan,SHOW);
mouse_mode = 3;
}
```

Pause a game.

The next piece of script you can use in any of your project. It's a pause game function. On a key press the game pauses completely until you press they key again. Later you could use a button for this but the function remains the same. I created a pause game panel that will be used. I will use the P key for pause and continue the game.

```
function pause()
{
if(freeze_mode == 2) // all frozen??
{
freeze_mode = 0; // go on
reset(break_pan,SHOW | TRANSLUCENT);
}
else
{
freeze_mode = 2; // pause
set(break_pan,SHOW | TRANSLUCENT);
}
}
```

To make it work on the key press P we need to add one line to the main function. (It should look familiar to you.)

```
on_p = pause;
```

Done. Every time you press the P key the game will pause and the pause panel will show until you press space again.

You will still hear the in game music play. If you want it to pause the music to we add 2 more lines to the pause function.

```
function pause()
{
if(freeze_mode == 2) // all frozen??
{
freeze_mode = 0; // go on
media_start(soundtrack01_handle);
reset(break_pan,SHOW | TRANSLUCENT);
}
else
{
freeze_mode = 2; // pause
media_pause(soundtrack01_handle);
set(break_pan,SHOW | TRANSLUCENT);
}
}
```

You see we used the variable we created for the soundtrack and use media commands to control it. So now you have a pause game function you can use for all of your mini games.

We have done enough for this issue as there will be more to come in the future.

Until then Happy game creation !

Rene Pol aka Realspawn.

