

Time for some basic movement stuff. In the mentioned mini games a lot of movement is used. Sometimes very simple movement but sometimes a more expanded kind of movement that includes gravity and jumping.

- Left and right movement.
- Up and down movement.
- Rotating movement.
- Left, right, up, down, jump movement.

In this issue we will do the movement step by step so you can alter it later to your own wishes.

We use Sid, a 3D character model created by otter from the 3d game studio community. Sid has a lot of animations that we will use and can be used freely to test your games with.

Ready let's go.



We will use the script created in issue 01. The first thing I'll do is create a so called pointer for the player model. This pointer is a unique name for the model so it can be called in script to activate any action or function.

```
////////////////////////////////////Entity pointers////////////////////////////////////  
ENTITY* sid;
```

Next is to create an action that makes our character move by using the arrow (cursor) keys.

```
////////////////////////////////////sid walking action////////////////////////////////////
```

```
action walking_sid()  
{  
  set(my,SHADOW);  
  sid = me;  
  c_setminmax(me);  
  var standing_percentage;  
  var walking_percentage;  
  while(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;  
      ent_animate(my, "stand", standing_percentage, ANM_CYCLE);  
      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, 0), nullvector, GLIDE);  
      wait(1);  
    }  
  }  
}
```

Assing this action to the player model and run your level. Now you can walk each direction and move around in the level by using the arrow keys.

So we can use movement now. In order to use jumping we need to add some more scripts.

First I create some variables used for jumping and then add two new functions that will handle the jumping and gravity during movement.

```
//////////////////////////////////Jumping and gravity//////////////////////////////////
```

```
VECTOR absdist = 50;  
var jump_time = 1;  
var jump_height = 30;  
var my_height = 50;
```

```
function my_jump()  
{  
  if(jump_time != 1){  
    return;  
  }  
  while(jump_time > -1){  
    jump_time -= 0.2 * time_step;  
    absdist.z = jump_height * jump_time * time_step;  
    wait(1);  
  }  
  while(my_height > 5)  
  {  
    wait(1);  
  }  
  jump_time = 1;  
}
```

```
function my_gravity()  
{  
  VECTOR temp;  
  vec_fill(temp.x, 0);  
  vec_set(temp.x, my.x);  
  temp.z -= 2000;  
  trace_mode = IGNORE_PASSABLE | IGNORE_ME | USE_BOX;  
  my_height = c_trace(my.x, temp.x, trace_mode);  
  if(my_height > 5 || my_height <= 0)  
  {  
    accelerate(absdist.z, -20 * time_step, 0.3);  
  }  
  else  
  {  
    if(jump_time == 1){  
      absdist.z = 0;  
    }  
  }  
}
```

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

Written by : René Pol 30-07-2017©

Now it's time to make some final changes in the walking_sid action to make him use the jump as we see fit. On key press space bar the my_jump function will be activated. And gravity is used during movement.

```
////////////////////////////////////sid walking action////////////////////////////////////

action walking_sid()
{
set(my,SHADOW);
sid = me;
c_setminmax(me);
var standing_percentage;
var walking_percentage;
on_space = my_jump;
while(1)
{
my_gravity();
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;
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);
}
}
}
```

Done deal !. We have movement with jumping in action Well done !

Two more things to add. A sound when jump is used and the jump animation should be used also. First we define the sound used for the jump like we learned before.

```
SOUND* jump_snd = "jump.wav";
```

Then we make some changes to the jump action. We add a variable that makes sure the sound is only played once during jump.

```
//////////////////////////////////Jumping and gravity//////////////////////////////////
```

```
VECTOR absdist = 50;  
var jump_time = 1;  
var jump_height = 30;  
var my_height = 50;  
var jumping =0;  
  
function my_jump()  
{  
if(jump_time != 1){  
return;  
}  
while(jump_time > -1){  
jump_time -= 0.2 * time_step;  
absdist.z = jump_height * jump_time * time_step;  
wait(1);  
}  
while(my_height > 5)  
{  
wait(1);  
}  
jumping =0;  
jump_time = 1;  
}
```

Last thing to do is to add script to our walking_sid action so during jumping the sound will be played.



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

Written by : René Pol 30-07-2017©

```
action walking_sid()
{
set(my,SHADOW);
sid = me;
c_setminmax(me);
var standing_percentage;
var walking_percentage;
var jumping_percentage;
on_space = my_jump;

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;
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);
}
}
}
```

Jumping with sound and animation is working now.

What about the basic and simple movement like moving only left and right, or up and down. Here are some script parts you can assign that will do the trick. You can use a models animation during the movement if you want.

Moving left right using arrow keys.

```
action moving_leftright()
{
set(my,SHADOW);
sid = me;
c_setminmax(me);
var my_speed =10;
while(1)
{
c_move(my, vector(0, my_speed * (key_cul - key_cur) * time_step, 0), nullvector, GLIDE | IGNORE_SPRITES);
wait(1);
}
}
```

Moving forward and backward using the arrow keys.

```
action moving_foreback()
{
set(my,SHADOW);
sid = me;
c_setminmax(me);
var my_speed =10;
while(1)
{
c_move(my, vector(my_speed * (key_cuu - key_cud) * time_step,0, 0), nullvector, GLIDE | IGNORE_SPRITES);
wait(1);
}
}
```

Moving up and down using the arrow keys.

```
action moving_updown()
{
set(my,SHADOW);
sid = me;
c_setminmax(me);
var my_speed =10;
while(1)
{
c_move(my, vector(0,0,my_speed * (key_cuu - key_cud) * time_step), nullvector, GLIDE | IGNORE_SPRITES);
wait(1);
}
}
```

Rotating left and right using the arrow keys.

```
action rotate_leftright()
{
set(my,SHADOW);
sid = me;
c_setminmax(me);
var my_speed =10;
while(1)
{
c_rotate (me, vector(my_speed *(key_cul - key_cur)* time_step, 0, 0), GLIDE | IGNORE_SPRITES);
wait(1);
}
}
```

You can combine these codes yourself in any way you see fit. You know enough of movement now to create your own mini games. If you have questions about the first 2 issues send me a mail and I will try to answer all of your questions in featured issues. Have fun with all you learned and I will meet you in next issue.

Happy game creation !
René Pol aka Realspawn.

