ISBN 9798329006933

For The Download Of Project Files & Resources Please Visit:
www.GameMakerBook.com

GAMEMAKER
101 TIPS
&
TRICKS

Table of Contents

1 Keep Instance In Room

This keeps the instance within the room's boundary. Assumes sprite origin as center.

```
x=clamp(x,0+sprite_width/2,room_width-
sprite_width/2);

y=clamp(y,0+sprite_height/2,room_height-
sprite_height/2);


//example movement code

hor = keyboard_check(ord("A"))-
keyboard_check(ord("D"));

ver = keyboard_check(ord("W"))-
keyboard_check(ord("S"));

x=x+hor;

y=y+ver;
```

2 Line Of Sight

This assumes you have three objects obj_player, obj_wall, obj_enemy. This example looks for a line of site from enemy to player. This example assumes sprite origins as center.

Create Event:

```
hidding=false;
```

Step Event:

```
if collision_line(x,y,obj_player.x,
obj_player.y,obj_wall,true,true)
{
  draw_line(x,y,obj_player.x,obj_player.y);
  hidding=true;
}
else
{
  hidding=false;
}
```

Draw Event:

```
draw_self();
if hidding { draw_text(x,y,"Hidden"); }
else
{
 draw_text(x,y,"Can See");
 draw_line(x,y,obj_player.x,obj_player.y);//shows
line - great for testing
}
```

3 Simple Level Complete Save System

To load saved data:

```
ini_open("Settings/savedata.ini");
level1 = ini_read_real("save1", "done", false);
level2 = ini_read_real("save2", "done", false);
level3 = ini_read_real("save3", "done", false);
level4 = ini_read_real("save4", "done", false);
level5 = ini_read_real("save5", "done", false);
level6 = ini_read_real("save6", "done", false);
level7 = ini_read_real("save7", "done", false);
level8 = ini_read_real("save8", "done", false);
ini_close();
```

To save data, where **global.level** is the level that has been completed:

```
ini_open("Settings/savedata.ini");
ini_write_real("save"+string(global.level),
"done", true);
ini_close();
```

4 Seeking Missile

This creates a missile that seeks out an enemy instance.

Create Event:

```
target = instance_nearest(x, y, obj_enemy);
speed=5;
alarm[0]=game_get_speed(gamespeed_fps)*8;
```

Alarm 0 Event:

```
instance_destroy();
```

Step Event:

```
if instance_exists(target)
{
  diff = angle_difference(point_direction(x, y, target.x, target.y), direction);
  direction += sign(diff) * min(abs(diff), 4);
}
image_angle=direction;
```

Collision Event with **obj_enemy**

```
with other instance_destroy();
instance_destroy();
```

5 Draw Health As Bars

Draws Health As Separate Sections.

Draw Event:

```
var number_of_bars, healthbar, width, height,
xpos,ypos, gap;

healthbar=health div 10;//change 10 to value you
wish each bar to represent

width=20;//width of each bar

height=20;//height of each bar

xpos=100;//change this to change draw location

ypos=100;//change this to change draw location

gap=5;

for (number_of_bars = 0; number_of_bars <
healthbar; number_of_bars +=1;)

{

  draw_set_color(c_red);

  draw_rectangle(xpos+width*number_of_bars+
(number_of_bars*gap), 0+ypos,

  xpos+width*number_of_bars+width+
(number_of_bars*gap), height+

  ypos,false)//false fills the rectangle

}
```

6 Ellipse Movement

This code will move in instance in an ellipse.

Create Event:

```
angle=0;

rx=256;

ry=64;

angle_speed=0.05*pi;

xc=xstart;

yc=ystart;
```

Step Event:

```
x=xc+rx*cos(angle); y=yc-ry*sin(angle);
angle+=angle_speed; if(angle>2*pi) angle-=2*pi;
else if(angle<0) angle+=2*pi;
```

7 Draw Contents Of 2D Array

A simple method to draw data from an array. Example **Create Event** code:

```
var i,j;
for (i = 0; i < 10; i++)
{
 for (j = 0;j < 10; j++)
 {
 array[i][j] = i*j;
 }
}
```

Draw Event:

```
cellsize=40;
border=2;
draw_set_colour(c_black);
for (i= 0;i < 10; i++)
{
   for (j= 0; j < 10; j++)
  {
  xpos=i*cellsize;
  ypos=j*cellsize;

draw_rectangle(border+xpos,border+ypos,border+xpo
s+cellsize,border+ypos+cellsize,2);
  draw_text(border+xpos+5,border+ypos+12,array[i]
[j]);
  }
}
```

8 Adaptable Engine Noise

This code changes te pitch of an engine noise based on its speed. Requires an engine noise sound.

Create Event:

```
global.sound=audio_play_sound(engine,1,true);
```

Step Event:

```
play_speed=spd/10;
audio_sound_pitch(global.sound,play_speed);
```

9 Drag-able Objects

Sometimes you may wish the player to be able to click and an instance around the room. Here is a simple solution:

Create Event:

```
dragging=false;
```

Step Event:

```
if position_meeting(mouse_x,mouse_y,id)
{
  if mouse_check_button_pressed(mb_left)
  {
            dragging=true;
            offsetx=x-mouse_x;
            offsety=y-mouse_y;
  }
  if mouse_check_button_released(mb_left)
  {
            dragging=false;
  }
}
```

End Step Event:

```
if dragging
{
x=mouse_x+offsetx
y=mouse_y+offsety
}
```

10 Mini Map

This code draws a mini-map,

Draw Event:

```
var d,a,xx,yy;

xx = obj_player.x;

yy = obj_player.y;

with(obj_wall_parent)

{

  d = point_distance(xx,yy,x,y);

  if(d <= 1000) { d = d/500*75; a =
point_direction(xx,yy,x,y)

  draw_sprite(spr_map, 0, view_xview[0]+75 +
lengthdir_x(d,a), view_yview[0]+75 +
lengthdir_y(d,a));

  }

}

draw_set_color(c_blue);

draw_rectangle(view_xview[0]+75-2,
view_yview[0]+75-
2,view_xview[0]+75+2,view_yview[0]+75+2,0);
```

11 Trail Effect

A simple script for drawing an image trail.

Create Event:

```
t = 0;
tm = 16;
speed = 10 + random(5);
_direction = -5 + random(10);
```

Draw Event:

```
t = min(t + 1, tm);
for (i = t; i > 1; i -= 1)
{
 tx[i] = tx[i-1];
 ty[i] = ty[i-1];
 ta[i] = ta[i-1];
 ti[i] = ti[i-1];
}
tx[1] = x;
ty[1] = y;
ta[1] = direction;
ti[i] = image_index;
direction += _direction;
for (i = 1; i <= t; i += 1)
draw_sprite_ext(sprite_index,ti[i],tx[i],ty[i],im
age_xscale,image_yscale,0,image_blend,sqr(1-i/
t));
```

12 Change Image On Mouse Interaction

This example uses a sprite with 3 different subimages,0=no mouse, 1=mouse over and 2=mouse pressed.

Step Event:

```
if instance_position(mouse_x,mouse_y,id)
{
 if(mouse_check_button(mb_left))
 {
   image_index = 2;
 }
 else
 {
  image_index = 1;
 }
}
else
{
  image_index=0;
}
```

13 Mouse Pointer Point Direction

Draws a sprite as pointer, pointing direction of target and slowly moves.

This example assumes sprite pointing right with origin as center.

Step Event:

```
move_towards_point(mouse_x,mouse_y,point_distance
(x,y,mouse_x,mouse_y)/12);
```

Draw Event:

```
window_set_cursor(cr_none);//hides default

draw_sprite_ext(sprite_index,0,x,y,1,1,point_dire
ction

(xprevious,yprevious,x,y),c_white,1);

//Note: use window_set_cursor(cr_default); to
allow drawing default cursor  again.
```

14 Power Up

A simple solution for managing a power up.

Create Event:

```
power_up=false;
```

Alarm 0 Event:

```
power_up=false;
```

Step Event:

```
if keyboard_check(ord('P')) && power_up=false
{
 power_up=true;
 alarm[0]=game_get_speed(gamespeed_fps)*8;
}
```

Draw Event:

```
if power_up=true
{
  draw_text(50,50,"POWER UP");
}
else
{
  draw_text(50,50,"NO POWER UP");
}
```

15 Push-able Block

A simple pushing system

obj_player **Step Event**:

```
if mouse_check_button(mb_left)x-=5;
if mouse_check_button(mb_right)x+=5;
```

obj_block Step Event:

```
if place_meeting(x+5,y,obj_player) {x-=5}
if place_meeting (x-5,y,obj_player) {x+=5}
```

16 Radar

A great adaptable radar system.

This assumes you have:

Object **obj_player** with sprite assigned

Object **obj_gem** with sprite

Sprite **spr_radar_gem** (6x6 pixels in red)

obj_radar Create Event:

Draw Event:

```
//stop here if no player
if (!instance_exists(Player_Obj))
{
 exit;
}
draw_circle(250,250,240,true);
draw_circle(250,250,160,true);
draw_line(250,10,250,490);
draw_line(10,240,490,240);
draw_set_colour(c_green)
var px=obj_player.x;
var py=obj_player.y;

with (obj_gem)
{
  var dist=point_distance(px,py,x,y)
  if dist<=2000
  {
       dist=dist/2000*250;
```

```
        angle=point_direction(px,py,x,y);


draw_sprite(spr_gem_blip,0,250+lengthdir_x(dist,a
ngle), 250+lengthdir_y(dist,angle));
  }
}
```

17 Random Word From A Text File

This example shows how to select a random word from a text file. Assumes a text file dictionary.txt is in included files.

The script code is:

```
function newWord()

{

  var file;

if(file_exists(working_directory +
"dictionary.txt")){

 //open the dictionary file

 file = file_text_open_read(working_directory +
"dictionary.txt");

 if(file == -1){

 //if loading the file failed return -1

 return -1; //will end the script

 }

 var wordList, wordNumber = 0;

 //make a list containing all words of the
dictionary

 while(!file_text_eof(file)){

 wordList[wordNumber] =
file_text_read_string(file);

 file_text_readln(file);

 wordNumber++;

 }

 file_text_close(file);

 return wordList[irandom(wordNumber-1)]; //return
a random word

}}
```

You can call with `word = newWord();`

18 Real Time Clock Example

Draws the players local time according ro the system clock.

Create Event:

```
minute=0; second=0; hour=0;
```

Step Event:

```
hour=current_hour;

minute=current_minute;

second=current_second;

hours=string(hour);

minutes=string(minute);

seconds=string(second);

seconds=string_repeat("0", 2-
string_length(seconds))+seconds;

minutes=string_repeat("0", 2-
string_length(minutes))+minutes;

hours=string_repeat("0", 2-string_length(hours))
+hours;
```

Draw Event:

```
draw_set_colour(c_black);

draw_text(300,300,hours+":"+minutes+":"+seconds);
```

19 Score With Leading Zeros

A simple method of drawing a score with leading 0;s

Draw Event:

```
str = string(score);

draw_text(x, y, string_repeat("0", 6-
string_length(str))+str);
```

20 Fading Moving Text

Creates moving fading text, great proving info to the player.

Create Event:

```
text[0]="This Text Floats Up";

text[1]="And Fade Outs at Top Of Screen";

text[2]="Great For Providing Inoformation";

text[3]="Or Display Game Credits";

text[4]="When It's Done";

text[5]="Program It To Do Something";

total_lines=array_length_1d(text);

i=room_height;

p=0;

h=1;
```

Draw Event:

```
draw_set_color(c_black);
draw_set_halign(fa_center);
draw_set_valign(fa_middle);
draw_set_font(font_text);///set your font here
h=i/100; draw_set_alpha(h);
draw_text_ext(room_width/2,i,text[p],20,room_widt
h-20);

if keyboard_check(vk_enter) i-=5; else i-=2

if i<0

{

i=(room_height+(string_height(text[p])*2)); p+=1

}

draw_set_alpha(1);

//if p>total_lines then do something, go to
room/restart for example
```

21 Cool Down System

A useful idea that limits how often a player can shoot / attack / or perform other actions.

Create Event:

```
global.hitmeter=0;

global.cooloff=false;
```

Step Event:

```
if mouse_check_button_pressed(mb_left) &&
global.cooloff=true

{

audio_play_sound(snd_neg,1,false)

exit;

}

//put your attack code here

if mouse_check_button_pressed(mb_left) &&
global.cooloff=false

{

  global.hitmeter+=25;

}

if global.hitmeter>100 && global.cooloff=false

{

  global.cooloff=true;

  global.hitmeter=100;


}


if global.cooloff

{

  if global.hitmeter<1

  {
```

```
        global.hitmeter=0;
        global.cooloff=false;
   }
}
global.hitmeter-=0.2;
if global.hitmeter>100 global.hitmeter=100;
if global.hitmeter<0 global.hitmeter=0;
```

Draw Event:

```
draw_self()
yy=500;
  draw_set_color(c_orange)
  draw_rectangle(100,yy-300,800,yy-200,false);
  draw_set_color(c_yellow)
  draw_rectangle(100,yy-300,100+
((700/100)*global.hitmeter),yy-200,false);
  draw_set_colour(c_black);
  draw_rectangle(100,yy-300,room_width-100,yy-
200,true);
  draw_set_halign(fa_center);
  draw_set_valign(fa_middle);
  draw_set_font(font_arcade);
if global.cooloff=false
{
  draw_text(room_width/2-200,yy-250,"CAN ATTACK
AND BLOCK");
}
if global.cooloff=true
{
  draw_text(room_width/2-200,yy-
250,"RECHARGING");
}
```

```
draw_set_colour(c_white);
draw_text(100,100,global.hitmeter);
```

22 Sliding Bar

A useful bit of code that allows a user to select a value. Assumes you have two sprite 64x64 in size, origin as center, **spr_bar** in red and **spr_slider** in green. Sprite bar is assigned to the object below. When you place object obj_slider in the room click **o**n a corner and stretch out as needed

Create Event:

```
minValue=x-(sprite_width/2);

maxValue=x+(sprite_width/2);

curValue=0.5;

length=maxValue-minValue;

isSliding=false;
```

Step Event:

```
if(mouse_check_button_pressed(mb_left)&&position_
meeting(mouse_x,mouse_y,self))

{

  isSliding=true;

}

if(mouse_check_button_released(mb_left))

{

  isSliding=false;

}

if(isSliding)

{

  curValue=(mouse_x-minValue)/length;

  if(mouse_x<minValue)

  {

       curValue=0;

  }
```

```
  if(mouse_x>maxValue)
  {
       curValue=1;
  }
}
```

Draw Event:

```
draw_self();
draw_sprite(spr_slider,0,(length*curValue)
+minValue,y);
draw_set_colour(c_black);
draw_text(x,y+10,string(curValue));
```

23 Slowly Move

This slowly moves to a position, in this example to the mouse's position.

Step Event:

```
movement_speed=25;//Higher Number Makes Slower Speed

target_x=mouse_x;//or other target position

target_y=mouse_y;//or other target position

x +=( target_x-x)/ movement_speed; //target position-current position

y +=( target_y-y)/ movement_speed; //target position-current position
```

24 Rising Smoke Effect

Create a rising and fading smoke effect. Requires a sprite of a smoke.

Create Event:

```
motion_set(90,2);

alp=1;
```

Step Event:

```
alp-=0.02;

if alp<0 instance_destroy();
```

Draw Event:

```
draw_sprite_ext(sprite_index,0,x,y,1,1,0,c_white,
alp);
```

25 Typewriter Text Effect

Draws text, one character at a time.

Create Event:

```
draw_set_colour(c_black);
draw_text(x, y, typewriter_out);
```

Alarm 0 Event:

```
typewriter_out+=string_copy(text_to_write,i,1);
i+=1;
if((i-1)!=string_length(text_to_write))
alarm[0]=5;
```

Draw Event:

```
draw_set_colour(c_black);
draw_text(x,y,typewriter_out);
```

26 Level Unlock System

The currently available levels is the value global.levels, ie a value of 2 would mean levels 1 and 2 unlocked. This needs to be declared before adding the following code, for example in a splash screen at the start of the game or at the end of completed level. This example uses a sprite with 2 subimages,' image 0' unlocked, 'image 1' locked, the sprite origin is the center.

Just duplicate the objects as needed, changing the value of my_id to the level number, also change the code in the left button released event to go to the level you want.

Create Event:

```
my_id=1;
locked=true;
subimage=1;
image_speed=0;
///check if allowed
if (global.level <= my_id-1)
{
 locked=true;
 subimage=1;
}
else
{
 locked=false;
 subimage=0;
}
```

Left Button Pressed Event:

```
///go to level(room) if unlocked if locked=false
{ room_goto(room_level_1); }
```

Draw Event:

```
draw_sprite(spr_levels,subimage,x,y);
draw_set_halign(fa_center);
draw_text(x,y,string(my_id));
```

27 Weapon Management

An example for storing info for multiple weapons. This example assumes you have the required graphics and sound effect referenced in the code below.

Create Event:

```
weapon_no=1;//handgun

global.weapon_info[weapon_no,1]="Hand
Gun";//Weapon Name

global.weapon_info[weapon_no,2]=100;//Starting
Number

global.weapon_info[weapon_no,3]=1;//Strength

global.weapon_info[weapon_no,4]=1;//Reload Speed

global.weapon_info[weapon_no,5]=1;//Cost

global.weapon_info[weapon_no,6]=5;//Aiming Speed

global.weapon_info[weapon_no,7]=spr_handgun;//Gun
Sight for Weapon

global.weapon_info[weapon_no,8]=obj_damage_handgu
n;//Damage Area for Weapon

global.weapon_info[weapon_no,9]=snd_handgun;//
Sound When Firing

global.weapon_info[weapon_no,10]=snd_voice_handgu
n_selected;//snd of voice  weapon selected

global.weapon_info[weapon_no,11]=20//ammo pack
size

weapon_no=2;//rifle

global.weapon_info[weapon_no,1]="Rifle";//Weapon
Name

global.weapon_info[weapon_no,2]=25;//Starting
Number

global.weapon_info[weapon_no,3]=2;//Strength

global.weapon_info[weapon_no,4]=2;//Reload Speed

global.weapon_info[weapon_no,5]=2;//Cost

global.weapon_info[weapon_no,6]=3;//Aiming Speed
```

```
global.weapon_info[weapon_no,7]=spr_rifle;//Gun
Sight for Weapon

global.weapon_info[weapon_no,8]=obj_damage_rifle;
//Damage Area for Weapon

global.weapon_info[weapon_no,9]=snd_rifle;//Sound
When Firing

global.weapon_info[weapon_no,10]=snd_voice_rifle_
selected;//snd of voice weapon selected

global.weapon_info[weapon_no,11]=15//ammo pack
size

global.weapon=1;

score=100;
```

Step Event:

```
if (keyboard_check_released(ord("1")))

{

 global.weapon=1;


audio_play_sound(global.weapon_info[global.weapon
,10],1,0);

}

if (keyboard_check_released(ord("2")))

{

 global.weapon=2;


audio_play_sound(global.weapon_info[global.weapon
,10],1,0);

}


if mouse_check_button_pressed(mb_left)

{

  if (score>=global.weapon_info[global.weapon,5])

  {
```

```
        //take money off score
        score=score-
global.weapon_info[global.weapon,5];
        //increase ammo no by one


global.weapon_info[global.weapon,2]=global.weapon
_info[global.weapon,2]+1;


       //play sound purchase complete


audio_play_sound(snd_voice_purchase_complete,1,0)
;
   }
  else
  {
   //play not enough cash

audio_play_sound(snd_voice_not_enough_cash,1,0);
 }
}
if mouse_check_button_pressed(mb_right)
{
  if (global.weapon_info[global.weapon,2]>0 )
  {

audio_play_sound(global.weapon_info[global.weapon
,9],1,0);
   global.weapon_info[global.weapon,2]-=1;
   instance_create_layer
(mouse_x,mouse_y,"Instances",global.weapon_info[g
lobal.weapon,8])
  }
  else
```

```
  {
   audio_play_sound(snd_voice_no_ammo,1,0);
  }
}
```

Draw Event:

```
draw_sprite(global.weapon_info[global.weapon,7],0
,x,y);

draw_set_colour(c_white);

draw_text(25,360,global.weapon_info[global.weapon
,1]);

draw_text(220,360,"Ammo:");

draw_text(350,360,global.weapon_info[global.weapo
n,2]);

draw_text(450,360,"Damage:");

draw_text(590,360,global.weapon_info[global.weapo
n,3]);

draw_text(100,100,"Cash "+string(score));
```

28 Top Down Character Control

4 direction movement and sprite control.

Create Event:

```
sprite_index=spr_down;
```

Step Event:

```
if (keyboard_check(ord("W")))
{
  sprite_index=spr_up;
  image_speed=1;
  y-=2;
}
else if (keyboard_check(ord("S")))
{
  sprite_index=spr_down;
  image_speed=1;
  y+=2;
}
else
if (keyboard_check(ord("A")))
{
    sprite_index=spr_left;
  image_speed=1;
  x-=2;
}
else
if (keyboard_check(ord("D")))
{
    sprite_index=spr_right;
  image_speed=1;
```

```
  x+=2;
}
else
{
  image_speed=0;
}
```

29 Blood Effect Using Particles

Creates a blood effect using particles.

Create Event:

```
global.particle_system = part_system_create();

part_system_depth(global.particle_system, 0);


// Create the particle type

global.particle_type = part_type_create();

part_type_shape(global.particle_type,
pt_shape_circle);

part_type_size(global.particle_type, 0.1, 0.5,
0.1, 0);

part_type_scale(global.particle_type, 1, 1);

part_type_orientation(global.particle_type, 0,
360, 0, 0,0);

part_type_color1(global.particle_type, c_red);

part_type_alpha3(global.particle_type, 1, 0.5,
0);

part_type_speed(global.particle_type, 2, 4, -0.1,
0);

part_type_direction(global.particle_type, 0, 360,
0, 0);

part_type_life(global.particle_type, 20, 40);

part_type_blend(global.particle_type, true);
```

Global Left Mouse Button Pressed Event:

```
number_of_particles=40

part_particles_create(global.particle_system,
mouse_x, mouse_y, global.particle_type,
number_of_particles);
```

Clean Up Event:

```
part_system_destroy(global.particle_system);
part_type_destroy(global.particle_type);
```

30 Draw Text Info With Background

A Simple script that can be used to draw info text for the player (assumes a room/view size of 1920 by 1080).

```
function scr_tip(message){
draw_set_alpha(0.5)
  draw_set_colour(c_white);
  draw_roundrect(500,200,1420,400,false);
  draw_set_font(font_hud);
  draw_set_valign(fa_middle);
  draw_set_halign(fa_center);
  draw_set_colour(c_white);
  draw_text(960,302,message)
  draw_text(960,298,message)
  draw_text(962,302,message)
  draw_text(958,298,message)
  draw_set_colour(c_black);
  draw_text(960,300,message)
  draw_set_alpha(1)
}
```

Example usage:

Create Event:

```
alarm[0]=game_get_speed(gamespeed_fps)*3;
```

Draw GUI Event:

```
message="Some Example Text\nTo Draw";
scr_tip(message);
```

Alarm 0 Event:

```
instance_destroy();
```

31 Speed Boost

A simple system to allow a temporary increase of an instance's max speed.

Create Event:

```
global.boost=false;

current_speed=4;
```

Collision with Powerup instance:

```
global.boost=true;

alarm[0]=game_get_speed(gamespeed_fps)*10;

with other instance_destroy();
```

An example for the **Step Event**, adapt as needed:

```
if !global.boost

{


move_towards_point(mouse_x,mouse_y,current_speed)
;

}

else

{


move_towards_point(mouse_x,mouse_y,current_speed*
2);

}
```

Alarm 0 Event:

```
global.boost=false;
```

32 Projectile Curved Path

Makes a projectile follow a curved path to target. This example ‘launches’ from bottom right, you can adapt as needed.

Create Event:

```
throws_path=path_add();

path_set_kind(throws_path, 1);

path_set_closed(throws_path, false);

if target_x<room_width/2
path_add_point(throws_path, room_width,
room_height, 50);

else path_add_point(throws_path, 0, room_height,
50);

path_add_point(throws_path, 600, 60, 50);

path_add_point(throws_path, target_x, target_y,
50);

path_start(throws_path, 50,
path_action_stop,true);
```

Path End Event:

```
//Add your code here to do something at path end

path_delete(throws_path);

instance_destroy();
```

33 Draw Power Bar With Image Background

A useful approach that draws power, health, boost etc,using an image as the background.

Very adaptable. Example below:

Create Event:

```
player_power=1;
player_max_power=20;
```

Example **Step Event**:

```
if mouse_check_button(mb_left)
{
  player_power-=0.1;
}
if mouse_check_button(mb_right)
{
  player_power+=0.1;
}
if player_power<0 player_power=0;
if player_power>player_max_power
player_power=player_max_power;
```

Draw Event:

```
draw_self();

draw_set_color(c_lime)

draw_text(100,100,player_power);

section_size=sprite_width/20;

draw_set_colour(c_white);

xx=x+sprite_width;

yy=y;

height=sprite_height;

draw_rectangle(xx,yy,xx-
(section_size*(player_max_power-
player_power)),yy+height,false);

draw_set_colour(c_black);

draw_rectangle(x,y,xx,y+height,true);
```

34 Create Muzzle Flash

This example shows how to create a muzzle flash for a rotating weapon. This system also rotates the muzzle flash with the weapon so it looks more realistic.

Weapon Object obj_player:

Create Event:

```
alarm[0]=game_get_speed(gamespeed_fps)*5;
ang=0;
```

Step Event:

```
if mouse_check_button(mb_left)
{
  ang++;
}
else
{
  ang--;
}
image_angle=ang;
```

Alarm 0 Event:

```
alarm[0]=game_get_speed(gamespeed_fps)*5;
instance_create_layer(x,y,"fx",obj_shoot);
```

Object obj_shoot:

Step Event:

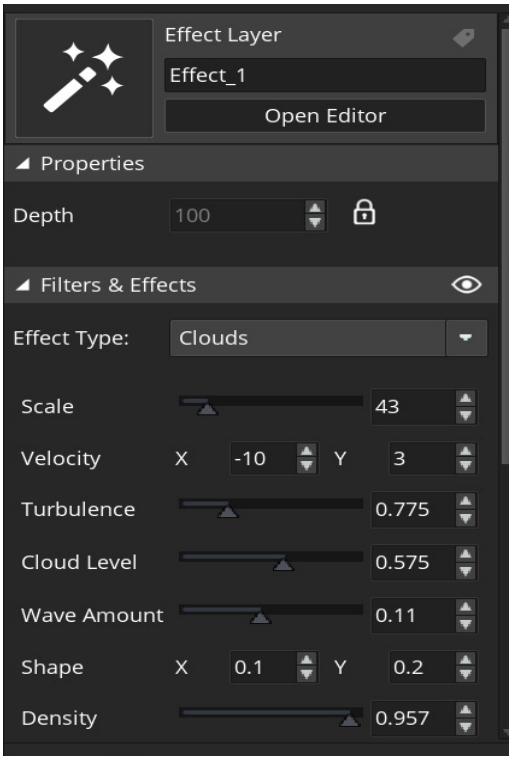
```
x=obj_player.x+lengthdir_x(100,obj_player.ang);
y=obj_player.y+lengthdir_y(100,obj_player.ang);
image_angle=obj_player.ang;
```

Animation End Event:

```
instance_destroy();
```

35 Cloud Effect Using Filters & Effects

This example make use of the powerful filters and effects layer. An example set up is shown below:



36 Missile Smoke Trail

A simple method using built in effects system to create smoke trail effect.

Example usage assumes sprite origin as middle center.

Step Event:

```
move_towards_point(mouse_x,mouse_y,4)
dir=point_direction(x,y,mouse_x,mouse_y);
if direction<dir direction+=0.01;
if direction>dir direction-=0.01;
image_angle=direction;
size=sprite_width/2;
xx=x-lengthdir_x(size,direction);
yy=y-lengthdir_y(size,direction);
effect_create_above(ef_smoke,xx,yy,4,c_blue);
```

37 Selectable Stats

This allows the player to choose a player object with various stats. This example for a driving game, but easily change if for other game styles.

Create Event:

```
alarm[0]=game_get_speed(gamespeed_fps)*3;

global.current=1;

tyres_sprite=[noone,face_1,face_2,face_3,face_4,face_5,face_6,face_7,face_8,face_9];

tyres_name=["","Blue Flash","Neon","Fast","Turning Speed","Attack","Accelerate","Top Speed","Good Allround","Easy Control"];

tyres_accelerate=[0,6,7,4,5,5,10,6,7,7];

tyres_topspeed=[0,7,5,10,8,6,7,10,7,5];

tyres_attack=[0,6,7,3,6,10,5,4,7,5];

tyres_turning=[0,6,4,5,10,3,4,4,7,10];
```

Alarm 0 Event:

```
alarm[0]=game_get_speed(gamespeed_fps)*3;

global.current++;

if global.current=10 global.current=1;
```

Draw Event:

```
draw_set_colour(c_white);

draw_sprite_ext(spr_faces,global.current,400,740,5,5,0,c_white,1);

//
draw_sprite_ext(spr_bikes,global.current,400,900,3,3,0,c_white,1);

draw_set_font(font_arcade);

draw_set_halign(fa_center);

draw_text(400,700,tyres_name[global.current]);
```

```
draw_text(room_width/2,150,"Choose Your
Character");

draw_text(room_width/2,2,"Tap Left Button To
Select");

draw_text(400,600,"Option
"+string(global.current)+" / 9");

draw_set_valign(fa_middle);

draw_set_halign(fa_right);

xx=room_width/3*2;

yy=700

draw_text(xx,yy,"Acceleration: ");

for (var i = 0; i < 10; i += 1)

{

  value=tyres_accelerate[global.current]

  draw_set_color(c_red);

  draw_rectangle(xx,yy-40,xx+500,yy+40,false);

  draw_set_color(c_green);

  draw_rectangle(xx,yy-
40,xx+(500/10*value),yy+40,false);

  draw_set_color(c_white);

  draw_rectangle(xx,yy-40,xx+(500),yy+40,true);

  draw_line(xx+(50),yy-40,xx+(50),yy+40);

  draw_line(xx+(100),yy-40,xx+(100),yy+40);

  draw_line(xx+(150),yy-40,xx+(150),yy+40);

  draw_line(xx+(200),yy-40,xx+(200),yy+40);

  draw_line(xx+(250),yy-40,xx+(250),yy+40);

  draw_line(xx+(300),yy-40,xx+(300),yy+40);

  draw_line(xx+(350),yy-40,xx+(350),yy+40);

  draw_line(xx+(400),yy-40,xx+(400),yy+40);

  draw_line(xx+(450),yy-40,xx+(450),yy+40);

}

//topspeed
```

```
yy=800
draw_text(xx,yy,"Top Speed: ");
for (var i = 0; i < 10; i += 1)
{
  value=tyres_topspeed[global.current]
  draw_set_color(c_red);
  draw_rectangle(xx,yy-40,xx+500,yy+40,false);
  draw_set_color(c_green);
  draw_rectangle(xx,yy-
40,xx+(500/10*value),yy+40,false);
  draw_set_color(c_white);
  draw_rectangle(xx,yy-40,xx+(500),yy+40,true);
  draw_line(xx+(50),yy-40,xx+(50),yy+40);
  draw_line(xx+(100),yy-40,xx+(100),yy+40);
  draw_line(xx+(150),yy-40,xx+(150),yy+40);
  draw_line(xx+(200),yy-40,xx+(200),yy+40);
  draw_line(xx+(250),yy-40,xx+(250),yy+40);
  draw_line(xx+(300),yy-40,xx+(300),yy+40);
  draw_line(xx+(350),yy-40,xx+(350),yy+40);
  draw_line(xx+(400),yy-40,xx+(400),yy+40);
  draw_line(xx+(450),yy-40,xx+(450),yy+40);
}
//attack
yy=900
draw_text(xx,yy,"Attack: ");
for (var i = 0; i < 10; i += 1)
{
  value=tyres_attack[global.current]
  draw_set_color(c_red);
  draw_rectangle(xx,yy-40,xx+500,yy+40,false);
  draw_set_color(c_green);
```

```
  draw_rectangle(xx,yy-
40,xx+(500/10*value),yy+40,false);
  draw_set_color(c_white);
  draw_rectangle(xx,yy-40,xx+(500),yy+40,true);
  draw_line(xx+(50),yy-40,xx+(50),yy+40);
  draw_line(xx+(100),yy-40,xx+(100),yy+40);
  draw_line(xx+(150),yy-40,xx+(150),yy+40);
  draw_line(xx+(200),yy-40,xx+(200),yy+40);
  draw_line(xx+(250),yy-40,xx+(250),yy+40);
  draw_line(xx+(300),yy-40,xx+(300),yy+40);
  draw_line(xx+(350),yy-40,xx+(350),yy+40);
  draw_line(xx+(400),yy-40,xx+(400),yy+40);
  draw_line(xx+(450),yy-40,xx+(450),yy+40);
}
//turning
yy=1000
draw_text(xx,yy,"Turning: ");
for (var i = 0; i < 10; i += 1)
{
  value=tyres_turning[global.current]
  draw_set_color(c_red);
  draw_rectangle(xx,yy-40,xx+500,yy+40,false);
  draw_set_color(c_green);
  draw_rectangle(xx,yy-
40,xx+(500/10*value),yy+40,false);
  draw_set_color(c_white);
  draw_rectangle(xx,yy-40,xx+(500),yy+40,true);
  draw_line(xx+(50),yy-40,xx+(50),yy+40);
  draw_line(xx+(100),yy-40,xx+(100),yy+40);
  draw_line(xx+(150),yy-40,xx+(150),yy+40);
  draw_line(xx+(200),yy-40,xx+(200),yy+40);
```

```
  draw_line(xx+(250),yy-40,xx+(250),yy+40);
  draw_line(xx+(300),yy-40,xx+(300),yy+40);
  draw_line(xx+(350),yy-40,xx+(350),yy+40);
  draw_line(xx+(400),yy-40,xx+(400),yy+40);
  draw_line(xx+(450),yy-40,xx+(450),yy+40);
}
```

And finally a **Step Event** that stores the selected stats in global variables so they can be used in game:

```
if mouse_check_button_pressed(mb_left)
{

global.tyres_sprite=tyres_sprite[global.current];
  global.name=tyres_name[global.current];

global.acceleration=tyres_accelerate[global.curre
nt];
  global.topspeed=tyres_topspeed[global.current];
  global.attack=tyres_attack[global.current];
  global.turning=tyres_turning[global.current];
  global.bike_sprite=global.current;
  room_goto(room_game);
}
```

38 Add Playing Cards & Shuffle

A simple method to shuffle a deck of playing cards.

Create Event:

```
x=room_width/2-(sprite_width/2);

y=500

deck=ds_list_create();

ds_list_add(deck,s_1,s_2,s_3,s_4,s_5,s_6,s_7,s_8,
s_9,s_10,s_11,s_12,s_13);

ds_list_add(deck,c_1,c_2,c_3,c_4,c_5,c_6,c_7,c_8,
c_9,c_10,c_11,c_12,c_13);

ds_list_add(deck,d_1,d_2,d_3,d_4,d_5,d_6,d_7,d_8,
d_9,d_10,d_11,d_12,d_13);

ds_list_add(deck,h_1,h_2,h_3,h_4,h_5,h_6,h_7,h_8,
h_9,h_10,h_11,h_12,h_13);

ds_list_shuffle(deck);

previous=ds_list_create();

//deal card

card=deck[|0];

ds_list_delete(deck,0);

sprite_index=card;

//current

name = sprite_get_name(card)

var _path_parts = string_split(name, "_");

value=real( _path_parts[1]);

suit=string( _path_parts[0]);
```

Draw Event:

```
draw_set_halign(fa_center);
if suit="s" suitis="Spades";
if suit="h" suitis="Hearts";
if suit="d" suitis="Diamonds";
if suit="c" suitis="Clubs";
if value<11 valueis=string(value);
if value=11 valueis="Jack";
if value=12 valueis="Queen";
if value=13 valueis="King";
if value=1 valueis="Ace";
draw_text(200,100,valueis+" Of "+suitis)
draw_sprite(sprite_index,0,x,y)
```

39 Enemy Track Player's Movement

Sometimes you may want an enemy to match the player's movement. This example moves the enemy's y position towards the player.

Step Event:

```
if obj_player.y<y
{
  y--;
}
else
{
  y++;
}
```

40 Tool Tip Pop Up

A useful bit of code that provides extra info to the player on mouse over.

Draw Event:

```
draw_self();
inst=instance_position(mouse_x,mouse_y,id)
if inst!=noone
{
  xx=inst.x;
  yy=inst.y-(sprite_height/2);
  draw_sprite(spr_bubble,0,xx,yy);
  draw_set_colour(c_green);
  draw_set_font(font_text);
  draw_set_halign(fa_center);
  draw_set_valign(fa_middle);
  draw_text(xx-80,yy-100,"Some Useful\nInfo Here");
}
```

41 Pop Up Wobbly Text

A simple text effect, great for using when a player collects a coin, showing the score the player gets.

Create Event:

```
size=0.2;
fade_size=5;//size to reach before fading
fade_speed=0.01;//how quickly to fade
angle=0;
sw=2;//for sine wave
fading=false;
alpha=1;
```

Step Event:

```
size+=0.1;//increase size
y-=0.3;//move up
sw+=0.2;
angle=sin(sw)*5;//for wave effect
if size>fade_size and fading==false//allowq
fading
{
  alpha-=fade_speed;
  if alpha<0 instance_destroy();
}
```

Draw Event:

```
draw_set_font(font_float);

if text<0 draw_set_color(c_red); else
draw_set_color(c_blue);

draw_set_halign(fa_center);

draw_set_valign(fa_bottom);

draw_set_alpha(alpha);

draw_text_transformed(x,y,text,size,size,angle);

draw_set_alpha(1);//set back so noting else
affected
```

You can call with:

```
textmake=instance_create_layer(x,y,"hud",obj_funk
y_text);

textmake.text=”You message here”;
```

42 Grow and Shrink Message Control

This example grows and shrinks an image. Great for giving info to the player. Assumes image origin is middle center.

Create Event:

```
scale=0;
dir="up";
x=room_width/2;
y=room_height/2;
sprite_index=spr_collect;
image_xscale=scale;
image_yscale=scale;
```

Step Event:

```
if dir="up" scale+=0.03;
if dir="up" and scale>1
{
  dir="wait";
  alarm[0]=game_get_speed(gamespeed_fps);
}
if dir="down" scale-=0.03;
if dir="down" and scale<0
{
  instance_destroy();
}
image_xscale=scale;
image_yscale=scale;
```

Alarm 0 Event:

```
dir="down";
```

43 Smoothly Move To Mouse Position

This moves smoothly to a position, in this example the mouse's position.

Create Event:

```
follow_speed=0.1;
target_x=x;
target_y=y;
```

Step Event:

```
target_x=mouse_x;
target_y=mouse_y;
x+=(target_x-x)*follow_speed;
y+=(target_y-y)*follow_speed;
```

44 Bounce Off Room Border

A simple effect, great for sprucing up a menu or pause screen.

Create Event:

```
speed=5;
direction=irandom(360);
```

Step Event:

```
if(x<=0||x>=room_width)
{
direction=180-direction+irandom_range(-10,10);
}
if(y<=0||y>=room_height)
{
direction=360-direction+irandom_range(-10,10);
}
direction=(direction+360)mod 360;
```

45 Pause Music When Sound Effect Plays

This little method pauses the background music whilst an audio effect plays, great for adding atmosphere to your games.

Create Event:

```
music=audio_play_sound(snd_music,1,true);

paused=false;
```

Step Event:

```
if mouse_check_button_pressed(mb_left) &&
paused==false

{

  audio_pause_sound(music);


explosion=audio_play_sound(snd_explosion,1,false)
;

  paused=true;

}

if paused==true

{

  if !audio_is_playing(explosion)

  {

       paused=false;

       audio_resume_sound(music);

  }

}
```

46 Glitch Effect Text

Creates a shaking text effect.

Create Event:

```
text_to_display = "Glitch Effect Example";
```

Draw Event:

```
var x_offset = irandom_range(-2, 2);
var y_offset = irandom_range(-2, 2);
var color_offset = irandom_range(-5, 5);
draw_set_font(font_text);
draw_set_color(c_blue + color_offset);
draw_text(x + x_offset, y + y_offset,
text_to_display);
```

47 Wobbly Text

This makes a cool text floating wobble effect.

Create Event:

```
text="Wobbly Text";
wave_amplitude=45;
wave_frequency=0.2;
wave_speed=0.2;
wave_timer=0;
```

Step Event:

```
wave_timer+=wave_speed;
```

Draw Event:

```
draw_set_font(font_text);
draw_set_colour(c_white);
varlen=string_length(text);
for(vari=0;i<len;i++)
{
  varchar=string_char_at(text,i+1);
varchar_x=x+i*40;//
Adjustthehorizontalspacingasneeded

varchar_y=y+wave_amplitude*sin(wave_frequency*(i+
wave_timer));
  draw_text(char_x,char_y,char);
}
```

48 Choose A Random Instance

This code chooses a random enemy instance.

Create Event:

```
global.target=noone;
```

Step Event:

```
if global.target=noone &&
instance_exists(obj_enemy)

{

  var inst =
instance_find(obj_enemy,irandom(instance_number(o
bj_enemy) - 1));

  global.target=inst;

  state="found_target";

}
```

You could then do something, such as moving towards the target:

```
if instance_exists(global.target) &&
distance_to_point(global.target.x,global.target.y
)>10

{

move_towards_point(global.target.x,global.target.
y,3)

}
```

Set `global.target` to `noone` when done.

49 Draw Mini Healthbar For Enemy

This is a simple method to draw a mini healthbar above an enemy instance. Assume sprite origin as middle center.

For example, in the **Create Event**:

```
maxhl=5;

hl=maxhl;
```

Draw Event:

```
draw_self();

parts=(100/maxhl)*hl;

sw=sprite_width/2;

sh=sprite_height/2;

draw_healthbar(x-sw,y-sh-10,x+sw,y-
sh,parts,c_black,c_red,c_green,0,true,true);
```

50 Fade In And Out

This fades a sprite in and out. Useful for a lot of applications.

Create Event:

```
alpha=0;
fade_speed=0.01;
fading_in=true;
```

Step Event:

```
if(fading_in)
{
  alpha+=fade_speed;
  if(alpha>=1)
  {
       alpha=1;
       fading_in=false;
  }
}
else
{
  alpha-=fade_speed;
  ifalpha<=0)
  {
       alpha=0;
       fading_in=true;
  }
}
```

Draw Event:

```
draw_sprite_ext(sprite_index,0,500,500,1,1,0,c_wh
ite,alpha);
```

51 Rotating Text

A Simple method for rotating text.

Create Event:

```
rotation_speed=1;
rotation_angle=0;
text_to_rotate="Hello, World!";
text_x=x;
text_y=y;
draw_set_font(font_text);
width=string_width(text_to_rotate);
```

Step Event:

```
rotation_angle+=rotation_speed;
xx=text_x+(-lengthdir_x(width/2,rotation_angle))
yy=text_y+(-lengthdir_y(width/2,rotation_angle))
text_x=x;
text_y=y;
```

Draw Event:

```
draw_set_font(font_text);
draw_set_color(c_white);
draw_text_transformed(xx,yy,text_to_rotate,1,1,ro
tation_angle);
```

52 Flashing Text

You can use this method to make text flash and glow, ideal for getting the attention of the player.

Draw Event:

```
color=make_colour_rgb(256,current_time  mod
256,random(256));

draw_set_color(color);

draw_set_font(font_text);

draw_text(x, y, "Your Text Here");
```

53 Play Sound At Selected Volume

Sometimes you may wish to play a sound at a lower or higher volume.

Here is a simple example.

Step Event:

```
if mouse_check_button(mb_left)
{
  //play 20& volume
  audio_play_sound(snd_explosion,1,0,0.2);
}
if mouse_check_button(mb_right)
{
  //play 100% volume
  audio_play_sound(snd_explosion,1,0,1);
}
```

54 Spawn Powerups Control System

An example system for providing the player with a bonus. Works on a percentage system, each with a chance of happening.

Create Event:

```
result="";
value=0;
```

Step Event:

```
if mouse_check_button_pressed(mb_left) //replace
with your bonus trigger
{
  var _chance = floor(irandom(99));
  if(_chance == 0)
  {
       result="Extra Life";
       //Your add life code
  }
  else if(_chance < 15)
  {
       result="Score Boost";
       //Your add score code
  }
  else if(_chance < 40)
  {
       result="Weapon Upgrade";
       //Your weapon upgrade
  }
```

```
else
  {
       result="nothing";
       //do nothing
  }
  value=_chance;//for testing
}
```

Draw Event:

```
draw_text(100,100,result);
draw_text(100,200,value);
```

55 Move Crosshair To Mouse Position

At time you want a player crosshair to move towards the mouse's position.

Here is a simple solution, **Step Event**:

```
targetx=mouse_x;
targety=mouse_y;
move_speed=4;
if distance_to_point(targetx,targety)>move_speed
move_towards_point(targetx,targety,move_speed)
```

56 Checkpoint System

Sometimes you want the player to save it's position at a level checkpoint.

This can be done with:

Create Event:

```
checkpoint_x=x;
checkpoint_y=y;
```

Collision with check point instance:

```
checkpoint_x=x;
checkpoint_y=y;
```

To return to checkpoint when health has run out:

```
x=checkpoint_x;
y=checkpoint_y;
lives--;
health=100;
```

57 Check If Instance Is In View

Sometimes you may want to check if an instance is in view, for example only allowing an enemy to shoot it's projectile when within the current view.

```
function scr_inview(xpos,ypos)
{
  var cam = view_camera[0];
  var x1 = camera_get_view_x(cam);
  var y1 = camera_get_view_y(cam);
  var x2 = x1 + camera_get_view_width(cam);
  var y2 = y1 + camera_get_view_height(cam);
  if( point_in_rectangle( xpos, ypos, x1, y1, x2,
y2)) return true;
  else return false;
}
}
```

Which you call with:

```
is_in_view=scr_inview(x,y);
```

Which will return **true** or **false**

58 Jump Through Platforms

Sometimes you may want to allow a player to be able to jump through a platform. This is one approach.

Create Event:

```
jump_speed=45;
gravity=1;
hsp=8;
```

Step Event:

```
if mouse_check_button(mb_left)
{
  x=x-hsp; image_xscale=-1
}
else
{
  x=x+hsp; image_xscale=1
}
if vspeed >= 0
{
        var _platform = instance_place(x, y +
vspeed, obj_block_parent);
        if (_platform != noone)
        {
                if (bbox_bottom <=
_platform.bbox_top)
                {
                  //do nothing
                }
                 else{vspeed=-jump_speed; }}}
```

59 Play Random Level Music

There may be time when you want to randomise background music for a level. The following is a simple method.

Create Event:

```
audio_stop_all();

global.music=audio_play_sound( choose(snd_music_1
,snd_music_2,snd_music_3,snd_music_4),1,true);
```

Storing the music track as a global variable allows you perform things on it whilst the game is playing. For example, to pause / resume the music.

60 Exploding Effect

If your sprite assets have several parts available here is a simple effect.

For example in a **Step Event**:

```
if mouse_check_button_pressed(mb_left)
{
  parts=sprite_get_number(spr_parts);
  for (var i = 0; i < parts; i += 1)
  {


inst=instance_create_layer(x,y,"explosions",obj_e
xplode);
       inst.sprite_index=spr_parts;
       inst.image_speed=0;
       inst.image_index=i;
       inst.direction=(360/parts)*i;
       inst.speed=4;
  }
}
```

You may wish to destroy **obj_explode** when outside room.

61 Draw Text With Border

Here is a simple script that draws text with a border. Great for sprucing up your GUI text.

```
function draw_txt_border(xx,yy,text,col1,col2)
{
  draw_set_colour(col1);
  draw_text(xx-1,yy-1,text);
  draw_text(xx-1,yy+1,text);
  draw_text(xx+1,yy+1,text);
  draw_text(xx+1,yy-1,text);
  draw_set_colour(col2);
  draw_text(xx,yy,text);;
}
```

Which you can call from a **Draw Event** with:

```
draw_txt_border(100,100, ”Hello
World”,c_black,c_red);
```

62 Move Coin To Score Text

This creates a coin moving effect that moves it towards the HUD text for displaying score. This example assumes the HUD text is in the bottom right on GUI layer.

Create Event:

```
alp=1;
```

Step Event:

```
alp=alp-0.01;
if alp<0.2
{
  alp=0.2
  var cam = view_camera[0];
  var x1 = camera_get_view_x(cam);
  var y1 = camera_get_view_y(cam);
  var x2 = x1 + 1800
  var y2 = y1 + 920
  move_towards_point(x2,y2,5)
  if distance_to_point(x2,y2)<20
  {
       score+=1;
       instance_destroy()
       audio_play_sound(snd_coin,1,false);
  }
}
```

Draw Event:

```
var cam = view_camera[0];

var x1 = camera_get_view_x(cam);

var y1 = camera_get_view_y(cam);

draw_sprite_ext(sprite_index,image_index,x-x1,y-
y1,1,1,0,c_white,alp);
```

63 Tire Track Effects

Draws tyre tracks that fade away. Ideal for top down car or tank games.

Spawning code for car or tank.

Create Event:

```
alarm[0]=5;
```

Alarm 0 Event:

```
alarm[0]=5;

xx=x-(lengthdir_x(120,image_angle));

yy=y-(lengthdir_y(120,image_angle));

inst=instance_create_layer(xx,yy,"tracks",obj_tra
cks);

inst.image_angle=image_angle;

show_debug_message("tracks")
```

obj_tracks Create Event:

```
alp=1;
```

Step Event:

```
alp-=0.01;

if alp<0 instance_destroy();
```

Draw Event:

```
draw_set_alpha(alp);

draw_self();

draw_set_alpha(1);
```

64 Fireworks Display

A cool graphical effect, ideal for when a player completes a level. Just pop the following code in a **Step Event**:

```
xx=irandom_range(0,room_width);

yy=irandom_range(0,room_height);

size=irandom(5);

col=choose(c_white,c_yellow,c_red,c_orange,c_purp
le,c_silver);

effect_create_above(ef_firework,xx,yy,size,col);
```

65 Spawn Bullets From Double Weapon

This allows for spawning bullets in a double-barrelled weapon.

Shooting code:

```
posx=x+lengthdir_x(30,ang-90).

posy=y+lengthdir_y(30,ang-90)

posxx=posx+lengthdir_x(160,ang)

posyy=posy+lengthdir_y(160,ang)

inst=instance_create_layer(posxx,posyy,"bullets",
tower_bullet);

inst.direction=ang;

inst.image_angle=ang;

inst.speed=6;



posx=x+lengthdir_x(30,ang+90)

posy=y+lengthdir_y(30,ang+90)

posxx=posx+lengthdir_x(160,ang)

posyy=posy+lengthdir_y(160,ang)

inst=instance_create_layer(posxx,posyy,"bullets",
tower_bullet);

inst.direction=ang;

inst.image_angle=ang;

inst.speed=6;
```

66 Sprite Drop Shadow

A simple method for creating a drop shadow effect. Great for top down shooter games, such as those with helicopters or planes.

Draw Event code:

```
draw_sprite_ext(sprite_index,0,x-90,y-
20,1,1,image_angle,c_black,0.4);

draw_self();
```

67 Simple Top Down Collision

This code works well in top down games, for example a racing game. Use this and it will bounce off instances.

For example the following in a **Collision Event** with a rock:

```
x+=lengthdir_x(2,point_direction(other.x,other.y,
x,y))
```

```
y+=lengthdir_y(2,point_direction(other.y,other.y,
x,y))
```

68 Spawn Instance With Random Subimage

This example spawns an instance with a random subimage and vertical speed.

Control Object:

Create Event:

```
alarm[0]=game_get_speed(gamespeed_fps)*4;
```

Alarm 0 Event:

```
alarm[0]=game_get_speed(gamespeed_fps)*4;

inst=instance_create_layer(irandom_range(128,room
_width-128),-200,"Instances",obj_gem);
```

obj_gem Create Event:

```
image_speed=0;

image_index=irandom(image_number-1);

vspeed=random_range(1,4);
```

69 Sine Wave Based Movement

A sine wave can be used for a lot of effects, for example smoothly moving an instance up and down, as in this example.

Step Event:

```
ypos=300;//where to anchor
spd=1000;//move speed - smaller is faster
dist=100;//distance to move
y=ypos+sin(current_time/spd)*dist
```

70 Draw Player's Speed On Dial

A simple way to visually show the player's speed.

Create Event:

```
spd=0;
```

An example **Step Event**:

```
if mouse_check_button(mb_left)
{
  spd-=0.1;
}
if mouse_check_button(mb_right)
{
  spd+=0.1;
}
spd=clamp(spd,0,7);
```

Draw Event:

```
draw_set_colour(c_white);
draw_set_halign(fa_center);
draw_set_valign(fa_middle);
draw_set_font(font_dial);
draw_sprite(spr_speed_dial,0,600,420);
ang=0-spd*30
draw_sprite_ext(spr_speed_pin,0,600,420,1,1,ang+1
40,c_white,1);
draw_text(600,600,abs(round(ang)));
```

71 Spawn Trees Border

A simple method of spawning a border of trees, suitable for a top-down vertical scrolling shooter.

Spawn code:

```
size=80;

xx=floor(room_height/size);

for (var i = 0; i < xx; i += 1)

{

  num=choose(1,2);

  if num=1

  {

instance_create_layer(100,i*size,"trees",obj_plan
t);

       instance_create_layer(room_width-
100,i*size,"trees",obj_plant);

  }

  if num=2

  {

instance_create_layer(100,i*size,"trees",obj_plan
t);

instance_create_layer(220,i*size,"trees",obj_plan
t);

       instance_create_layer(room_width-
100,i*size,"trees",obj_plant);

       instance_create_layer(room_width-
220,i*size,"trees",obj_plant);

  }

}
```

72 One Button Controlled Movement

Allows you to add some accessibility to your games, by allow the player to move a crosshair around the room with a single button.

Create Event:

```
rotate:=0;
```

Step Event:

```
if mouse_check_button(mb_left) or
keyboard_check(vk_space)
{
  motion_set(rotate,5);
}
else
{
  rotate+=2.5;
  speed=0;
}
width=sprite_width/2;
height=sprite_height/2
x=clamp(x,width,room_width-width);
y=clamp(y,height,room_height-height);
```

Draw Event:

```
draw_self();
draw_sprite_ext(spr_arrow,0,x,y,1,1,rotate,c_whit
e,1);
```

73 Plane Movement (Side Shooter)

A flexible movement system for a player plane in a side-scrolling shooter. Assumes sprite origin as middle center. **Create Event**:

```
ang=0
speed=3;
```

Step Event:

```
if mouse_check_button(mb_left)
{
  ang++;
}
else
{
  ang--;
}
if ang>80 ang=80
if ang<-80 ang=-80
image_angle=ang;
direction=ang;
if y<100
{ y+=2; ang-=2;}
if y>500{y-=2; ang+=2;}
```

74 Player Power Up Creator

Sometimes you may wish for your player to work hard to earn a powerup. This example requires the player mash the mouse buttons to unlock an upgrade. **Create Event**:

```
spd=0;
target=1000;
target_reached=false;
```

Step Event:

```
if mouse_check_button_pressed(mb_left)
{
  spd+=5;
}
if mouse_check_button_pressed(mb_right)
{
  spd+=5;
}
if !target_reached spd-=0.2;
if spd>target
{
  target_reached=true;
}
if spd>target spd=target;
if spd<0 spd=0;
```

Draw Event:

```
draw_text(100,100,spd);
if target_reached
{
  draw_text(100,150,"Target Reached"); }
```

75 Coin Drop Bonus Effect

Creates a cool looking graphical effect. First you'll need a control object to spawn the coins, for example with the following in a **Step Event:**

```
if mouse_check_button_pressed(mb_left)
{
  repeat(10)
  {

  instance_create_layer(x,y,"Effects",obj_coin);
       show_debug_message("spawned")
  }
}
```

And the code for **obj_coin**: **Create Event**:

```
x=random_range(100,1820);
y=-200
distance=irandom_range(200,1000);
vspeed=random_range(1,3);
```

Step Event:

```
if y>distance
{
  instance_destroy();
  effect_create_above(ef_ring,x,y,5,c_yellow);
effect_create_above(ef_firework,x+12,y+12,5,c_red
);
  effect_create_above(ef_firework,x-12,y-
12,5,c_red);
  effect_create_above(ef_firework,x+12,y-
12,5,c_red);
  effect_create_above(ef_firework,x-
12,y+12,5,c_red);}
```

76 Endless Levels

By using a static screen without views you can create an endless level by spawning instances off screen and for example moving them down. Example assumes a sprite

Spawner object, Create Event:

```
alarm[0]=game_get_speed(gamespeed_fps)*3;
```

Alarm 0 Event:

```
alarm[0]=game_get_speed(gamespeed_fps)*3+random(game_get_speed(gamespeed_fps)*2);

instance_create_layer(irandom_range(50,room_width-50),-300,"Island",obj_island);
```

obj_island Create Event:

```
sprite_index=choose(Water_Object_3,Water_Object_4,Water_Object_5,Water_Object_6,Water_Object_7,Water_Object_8);

vspeed=1;
```

Step Event:

```
if y>room_height+sprite_height
instance_destroy();
```

You can use a similar approach for clouds, enemies etc.

77 Mini Healthbar With Segments

Draws a healthbar with segments.

Create Event:

```
hp=5;

max_hp=hp;
```

Draw Event:

```
draw_self();

draw_set_color(c_black)

draw_healthbar(x-100,y-60,x+100,y-80,(100/
max_hp)*hp,c_white,c_blue,c_yellow,0,true,true);

for (var i = 0; i < max_hp; i += 1)

{

  draw_line(x-100+(200/max_hp)*i,y-60,x-
100+(200/max_hp)*i,y-80)

}
```

78 Outline Shader

Creates a white border around an instance.

```
function outline_init()

{

  uni_size = shader_get_uniform(sh_outline,
"size");

  uni_thick =
shader_get_outline_init()uniform(sh_outline,
"thick");

  uni_color = shader_get_uniform(sh_outline,
"oColor");

  uni_acc = shader_get_uniform(sh_outline,
"accuracy");

  uni_tol = shader_get_uniform(sh_outline,
"tol");

  uni_uvs = shader_get_uniform(sh_outline,
"uvs");

}

function outline_start() {

  var _spr;

  if (argument_count<=2) _spr = sprite_index;

  else _spr = argument[2];

  shader_set(sh_outline);

  var _tex = sprite_get_texture(_spr,
image_index);

  var _w = texture_get_texel_width(_tex);

  var _h = texture_get_texel_height(_tex);

  shader_set_uniform_f(uni_size, _w, _h);

  shader_set_uniform_f(uni_thick, argument[0]);

  shader_set_uniform_f(uni_color,
color_get_red(argument[1])/255,
color_get_green(argument[1])/255,
color_get_blue(argument[1])/255);
```

```
  var acc;
  if (argument_count<=3) acc = 16;
  else acc = argument[3];
  shader_set_uniform_f(uni_acc, acc);
  var tol;
  if (argument_count<=4) tol = 0;
  else tol = argument[4];
  shader_set_uniform_f(uni_tol, tol);
  var uvs = sprite_get_uvs(_spr, image_index);
  shader_set_uniform_f(uni_uvs, uvs[0], uvs[1],
uvs[2], uvs[3]);


}
function outline_start_surface() {
  var _sur = argument[2];
  shader_set(sh_outline);
  var _tex = surface_get_texture(_sur);
  var _w = texture_get_texel_width(_tex);
  var _h = texture_get_texel_height(_tex);
  shader_set_uniform_f(uni_size, _w, _h);
  shader_set_uniform_f(uni_thick, argument[0]);
  shader_set_uniform_f(uni_color,
color_get_red(argument[1])/255,
color_get_green(argument[1])/255,
color_get_blue(argument[1])/255);
  var acc;
  if (argument_count<=3) acc = 16;
  else acc = argument[3];


  shader_set_uniform_f(uni_acc, acc);
  var tol;
  if (argument_count<=4) tol = 0;
```

```
  else tol = argument[4];
  shader_set_uniform_f(uni_tol, tol);
  var uvs = texture_get_uvs(_sur);
  shader_set_uniform_f(uni_uvs, uvs[0], uvs[1],
uvs[2], uvs[3]);
}
function outline_end()
{
  shader_reset();
}
```

Shader vsh:

```
//
// Simple passthrough vertex shader
//
attribute vec3 in_Position;                  //
(x,y,z)
//attribute vec3 in_Normal;                  //
(x,y,z)     unused in this shader.
attribute vec4 in_Colour;                    //
(r,g,b,a)
attribute vec2 in_TextureCoord;              //
(u,v)


varying vec2 v_vTexcoord;
varying vec4 v_vColour;


void main()
{
    vec4 object_space_pos = vec4( in_Position.x,
in_Position.y, in_Position.z, 1.0);
    gl_Position =
gm_Matrices[MATRIX_WORLD_VIEW_PROJECTION] *
object_space_pos;
```

```
    v_vColour = in_Colour;
    v_vTexcoord = in_TextureCoord;
}
```

Shader vsh:

```
//
// Simple passthrough fragment shader
//
varying vec2 v_vTexcoord;
varying vec4 v_vColour;

uniform vec2 size;
uniform float thick;
uniform vec3 oColor;
uniform float accuracy;
uniform float tol;
uniform vec4 uvs;

const float rad_circle = 6.28319;

void main()
{
    gl_FragColor = v_vColour * texture2D( gm_BaseTexture, v_vTexcoord );
    bool outline = false;

    for(float i=1.0; i<=thick; i++){
        for(float d=0.0; d<rad_circle; d+=rad_circle/accuracy){
            vec2 check_pos = v_vTexcoord + i*vec2(cos(d)*size.x, -sin(d)*size.y);
```

```
            vec4 datPixel =  v_vColour *
texture2D( gm_BaseTexture, check_pos);


            bool out_bound = check_pos.x < uvs.r
|| check_pos.y < uvs.g || check_pos.x > uvs.b ||
check_pos.y > uvs.a;


            if (datPixel.a>tol &&
gl_FragColor.a<=tol && !out_bound){
                outline = true;
                break;
            }
        }
       if (outline) break;
    }


    if (outline) gl_FragColor = vec4(oColor.r,
oColor.g, oColor.b, 1.0);
}
```

Example usage:

```
outline_start(12,c_white);
draw_self();
outline_end();
```

79 Move Crosshair To Target

This automatically selects a target and moves the crosshair to the target.

Create Event:

```
target=noone;
alarm[0]=game_get_speed(gamespeed_fps)/2;
image_speed=0;
image_index=0;
```

Step Event:

```
x=clamp(x,0,room_width);
y=clamp(y,0,room_height);
if place_meeting(x,y,obj_parent)
{
  image_index=1;
}
else
{
  image_index=0;
}
if target=noone
{
  with (obj_parent)
  {
      other.target=id;
  }
}
```

Alarm 0 Event:

```
with (obj_parent) {
  if (in_view) {
    other.target=id;
  }
}
with id active=true;
```

80 Spawn Items With Gap

Sometimes you may want to place multiple instances in your room, which can laborious if you have multiple items. Below shows one method to place multiple instances with a gap inbetween.

Create Event:

```
pos=300;

repeat (8)

{

  instance_create_layer(x-
pos,y,"items",choose(obj_big,obj_pole,obj_round))
;


instance_create_layer(x+pos,y,"items",choose(obj_
big,obj_pole,obj_round));

  pos+=200;

}
```

81 Add Scores To A List

There may be times when you want to keep track of player score, for example in a darts game. This code keeps track of data. It could also be used to keep track of words.

Create Event:

```
global.last=ds_list_create();
```

Step Event:

```
//limit to 12 entries
size=ds_list_size(global.last);
if size>12
{
  ds_list_delete(global.last,0);
}
//for testing
if mouse_check_button_pressed(mb_left)
{
  ds_list_add(global.last,irandom(1000));
}
```

Draw Event:

```
draw_set_colour(c_white);
var size=ds_list_size(global.last);
var i;
for (i = 0; i < size; i += 1)
{
    draw_text(900,60+(32*i),global.last[|i]);
}
```

82 Cloud Effect

Create a cloud scrolling effect, great for above and below various layers in your game.

Create Event:

```
image_speed=0;

image_index=choose(0,1,2);

dir=choose(-1,-2,1,2);
```

Step Event:

```
x=x+dir;

if x<-sprite_width x=room_width+sprite_width;

if x>room_width+sprite_width x=-sprite_width;
```

And a **Draw Event,** so the cloud has some transparency:

```
draw_sprite_ext(sprite_index,image_index,x,y,1,1,
0,c_white,0.5);
```

Just pop a few instances in your room and you're good to go.

83 Ball Bouncing Off Instances

A Simple method to make a ball bounce off of walls. This example requires a ball, plus two additional objects for top and bottom walls, plus one for side walls. This setup also gives a bit of randomisation with the angle when the ball bounces.

Create Event:

```
spd = 3;

motion_set(choose(130,310),spd);
```

Collision Event with Top & Bottom walls:

```
move_bounce_all(true);
```

Collision Event with Sides walls:

```
move_bounce_all(true);
```

Collision Event with **obj_ball**:

```
move_bounce_all(true);
```

84 Keeping A Value In A Range

There will be time when you wish to keep a value within a given range. This can be done with `clamp`. For example in a driving game when you wish stop the car leaving the room.

Some example code in a **Step Event**, keeping the x value between 100 and 1000:

```
if mouse_check_button(mb_left)
  {
       x-=5;
  }
  else if mouse_check_button(mb_right)
  {
       x+=5;
  }

  x=clamp(x, 100, 1000);
```

85 Film Style Scrolling Credits

Create scrolling text in a film style. Ideal for showing credits on game complete.

obj_spawner Create Event:

```
list=ds_list_create();
ds_list_add(list,
"Example Text",
"Add Your Own",
"",
"Game Credits"
);
alarm[0]=game_get_speed(gamespeed_fps)*3;
```

Alarm 0 Event:

```
if ds_list_size(list)=0 exit;
alarm[0]=game_get_speed(gamespeed_fps)*3;
text=list[|0];
ds_list_delete(list,0);
instance_create_layer(x,y,"text",obj_text,
{text : text})
```

obj_text Create Event:

```
size=1;
x=room_width/2;
y=room_height+100;
```

Step Event:

```
y--;
if y<room_height size-=0.001;
if size<0.01 instance_destroy();
```

Draw Event:

```
draw_set_font(font_text);

draw_set_halign(fa_center);

draw_set_colour(c_white);

draw_text_ext_transformed(x,y,text,4,1000,size,si
ze,0);
```

86 Rotate Room View

This method provides a great method for giving additional visual feedback to the player. Idea for a driving or flying game. Assumes the room has view 0 set up.

Create Event:

```
ang=0;
```

Step Event:

```
if mouse_check_button(mb_left)
{
  ang--;
  if ang<-60 ang=-60
}
else
{
  ang++;
  if ang>60 ang=60;
}
camera_set_view_angle(view_camera[0],ang);
```

87 Toggle Full Screen

There may be times when you want to allow the player to choice of full screen or windowed. The following allows the play to swap between the two using a key press.

Step Event:

```
if keyboard_check(ord("W"))
{
  if window_get_fullscreen()=true
  {
       window_set_fullscreen(false);
  }
else if window_get_fullscreen()=false
  {
       window_set_fullscreen(true);
  }
}
```

88 Progress Bar

Sometimes it's nice to show the player their level progress. It's quite simple to do this visually. This example is for a side-scroller. Assumes starting x position is 0 and the target is the right end of the room.

Draw Event:

```
current_y=obj_player.y+250;

_percent=(((room_height-450)/(room_height-
450))*current_y)/(room_height-450)

_percent=100-((1-_percent)*100)

{

  draw_set_color(c_black)

  draw_healthbar(180,1080-80,1580,1080-140,100-
_percent,c_white,c_yellow,c_yellow,0,true,true);

  for (var i = 0; i < 100; i += 1)

  {

       draw_line(180+(1400/100)*i,1080-
80,180+(1400/100)*i,1080-140)

  }


  draw_rectangle(180,1080-80,1580,1080-140,true);

}

draw_text(80,1080-110,"PROGRESS");;
```

89 Flash Player To Show Damage

This example shows one method of visually showing a player that they have taken damage. This changes the player's character to red.

Create Event:

```
can_hit=true;
```

Collision Event with whatever instance causes damage:

```
if can_hit
{
   health-=5;
   can_hit=false;
    alarm[0]=game_get_speed(gamespeed_fps);
}
```

Alarm 0 Event:

```
can_hit=true;
```

Draw Event:

```
if can_hit=false
{

draw_sprite_ext(sprite_index,image_index,x,y,1,1,
0,c_red,1);
}
else
{
  draw_self();
}
```

90 Fade Between Two Images

This fades in and out two subimages. Great for making appealing looking text without the overhead of having to use multiple images.

Create Event:

```
alp=0;
dir="up";
```

Step Event:

```
if dir="up"
{
  alp+=0.01;
}
if dir="down"
{
  alp-=0.01;
}
if alp>1 dir="down";
if alp<0 dir="up";
```

Draw Event:

```
draw_sprite_ext(spr_logo,0,x,y,1,1,0,c_white,1-
alp);
draw_sprite_ext(spr_logo,1,x,y,1,1,0,c_white,alp)
;
```

91 Draw A Path As Circles

A useful method of drawing a path as circles. Ideal for giving hints to the player. Assumes a path with id **path** has been created.

Draw Event:

```
var _path = path;
var _pathLength = path_get_length(_path);
var _spacing = 30;
var _dots = _pathLength / _spacing;
var i = 0;
repeat(_dots)
{
	var _pos = 1 - (i / _dots);
	draw_circle(path_get_x(_path, _pos), path_get_y(_path, _pos), 8, false);
	++i;
}
```

92 Simple Top Down Movement Control

A simple top down movement control that uses a single mouse button. The good thing is that this leaves the right mouse button free for attacking.

Create Event:

```
dir=0
speed=6;
```

Step Event:

```
if mouse_check_button(mb_left)
{
  dir+=1;
}
else
{
  dir-=1;
}
direction=dir;
image_angle=direction;
```

93 Laser To Target

This draws a laser from the instance to a target,which in this example is the mouse's x position.

Create Event:

```
size=0;

segments=0;

segment_size=64;

remainder=0;

final_xpos=0;

subimage=0;
```

Step Event:

```
size=0;

segments=0;

segment_size=64;

remainder=0;

final_xpos=0;

subimage=0;
```

Draw Event:

```
draw_self();

for (var i = 0; i < segments; i += 1)

{

  draw_sprite(spr_laser,subimage,x+
(segment_size*i),y)

}

final_xpos=segments*segment_size;

draw_sprite_part(spr_laser,subimage,0,0,remainder
,55,x+final_xpos,y-(27));
```

94 Bubble Explosion Effect

Here's a simple effect that makes a load of bubbles great for an underwater themed game (or use something other than bubbles to match your theme).

Spawn code:

```
repeat(20)
{
inst=instance_create_layer(x,y,"bubbles",
obj_bubble);
 inst.direction=irandom(360);
 inst.speed=random_range(1,4);
}
obj_bubble:
```

Create Event:

```
gravity=-0.1;
alarm[0]=game_get_speed(gamespeed_fps)+
(random(game_get_speed(gamespeed_fps)));
```

Alarm 0 Event:

```
snd=choose(bub1,bub2,bub3,bub4,bub5,bub6,bub7);
audio_play_sound(snd,1,false);
instance_destroy();
```

95 Add Instances To A Grid & Move To A Target

Below is a simple method of making a grid with areas that are not allowed. It then makes a path to a random instance.

Create Event:

```
grid = mp_grid_create(0, 0, room_width / 128, 
room_height / 128, 128, 128);

  mp_grid_add_instances(grid, obj_block, true);

  path = path_add();

    mp_grid_path(grid, path, x, y, obj_melon.x, 
obj_melon.y, 1);

  path_start(path,3,path_action_stop,true);
```

You could then create another target and reuse the above code.

96 Image Scale, Fade and Rotate Effect

An effect that can be used for a variety reasons.

Assumes a sprite assigned that is 512x512 in size and origin as middle center.

Example spawn code:

```
if mouse_check_button_pressed(mb_left)
{
instance_create_layer(room_width/2,room_height/2,"Effects",obj_effect);
}
```

obj_effect

Create Event:

```
size=0.1;
alp=1;
dir=choose(-1,1);
angle=0;
```

Step Event:

```
size+=0.05;
alp-=0.01;
angle+=dir*3;
if alp<0 instance_destroy();
```

Draw Event:

```
draw_sprite_ext(sprite_index,0,(room_width/2)+90,room_height/2,size,size,angle,c_white,alp);
```

97 Player Streak

Keep tracks of a player's streak, for example the number of enemies killed without missing.

Create Event:

```
streak=0;
```

Step Event (testing example):

```
if mouse_check_button_pressed(mb_right)
{
  //call the folling to add to streak
  streak++;
}
if mouse_check_button_pressed(mb_left)
{
  //call the folling to reset streak
  streak=0
}
```

Draw Event example:

```
draw_set_colour(c_black);
draw_text(200,200,"Streak: "+string(streak));
```

98 Numbers As Text

This script will change digits to text, for example **16783** to s**ixteen thousand seven hundred and eighty three**. Works upto 999999

```
function integer_to_english(int) {

    // Lookup tables

    static _digits_ = ["", "one", "two", "three",
"four", "five", "six", "seven", "eight", "nine"];

    static _teens_ = ["ten", "eleven", "twelve",
"thirteen", "fourteen", "fifteen", "sixteen",
"seventeen", "eighteen", "nineteen"];

    static _tens_ = ["", "", "twenty", "thirty",
"forty", "fifty", "sixty", "seventy", "eighty",
"ninety"];


    // Decompose digits

    var thousands = (int div 1000) mod 1000;

    var hundreds = (int div 100) mod 10;

    var tens = (int div 10) mod 10;

    var units = int mod 10;


    // Accumulator

    var str = "";


    // Add thousands

    if (thousands > 0) {

        str += integer_to_english(thousands) + "
thousand";

    }
```

```
// Add hundreds
   if (hundreds > 0) {
       if (str != "") {
           str += " ";
       }
       str += _digits_[hundreds] + " hundred";
   }

   // Add tens and digits
   if (int mod 100 > 0) {
       if (str != "") {
           str += " and ";
       }
       switch (tens) {
           case 0:
               str += _digits_[units];
           break;
           case 1:
               str += _teens_[units];
           break;
           default:
               str += _tens_[tens];
               if (units > 0) {
                   str += " " + _digits_[units];
               }
           break;
       }
   }
   // Done
   return str;}
```

99 Draw Lives As Images

This draws the players lives as images.

Create Event:

```
lives=8;
```

Step Event for testing:

```
if mouse_check_button_pressed(mb_left)
{
  lives--;
}
if mouse_check_button_pressed(mb_right)
{
  lives++;
}
```

Draw Event:

```
for (var i = 0; i < lives; i += 1)
{
  draw_sprite(spr_heart,0,100+(i*140),100);
}
```

100 Coin Explosion Effect

The following is a cool explosion effect, great when collecting coins.

```
for (var i = 0; i < 36 i += 1)

{

inst=instance_create_layer(x,y,"effect",obj_coin_
effect);

  inst.direction=i*10

  inst.speed=1;

}
```

And in outside_room Event:

```
instance_destroy();
```

101 Text With Gradient

This script draws gradient text with a border. Great for game titles, game-over screen etc..

```
function
draw_text_gradient(xpos,ypos,mess,sep,wid,bordcol
,gradcol1,gradcol2,alp)

{

  draw_text_ext_colour(xpos+2, ypos+2,mess, sep,
wid,bordcol,bordcol,bordcol,bordcol,alp);

  draw_text_ext_colour(xpos+2, ypos-2,mess, sep,
wid,bordcol,bordcol,bordcol,bordcol,alp);

  draw_text_ext_colour(xpos-2, ypos+2,mess, sep,
wid,bordcol,bordcol,bordcol,bordcol,alp);

  draw_text_ext_colour(xpos-2, ypos-2,mess, sep,
wid,bordcol,bordcol,bordcol,bordcol,alp);

  draw_text_ext_colour(xpos, ypos,mess, sep,
wid,gradcol1,gradcol2,gradcol1,gradcol2,alp)


}
```

Which you can call from a **Draw Event** with:

```
draw_text_gradient(room_width/2-300,300,"Hello
World",124,1900,c_black,c_yellow,c_red,1)
```

AVAILABLE TO ORDER

GAMEMAKERBOOK.COM