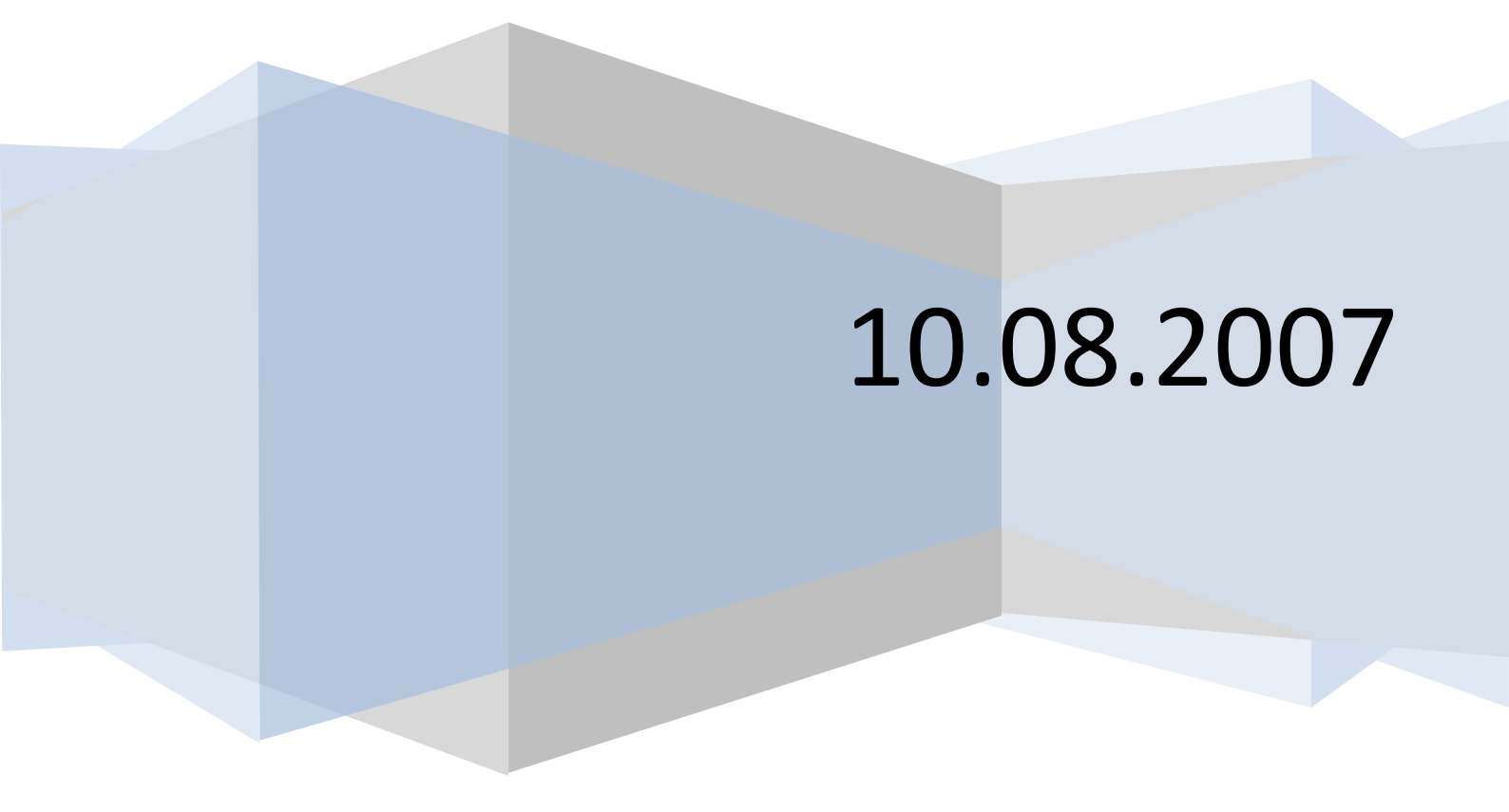


Mouvement pixel par pixel

Scripts de -----

Marcoest



10.08.2007

Auteur: ???

Titre: Game_Character

Description: Votre héros ne bougera plus case par case mais pixel par pixel.

Image(s):



Installation:

Créez un nouveau script avant "Main", appelez-le "Game_Character" puis collez le script suivant dedans.

```
class Game_Player < Game_Character
#-----
# ?œ 'è?'
#-----
TOP_SPACE = BOTTOM_SPACE = 16 # ?ã%º,ì"»'è-³Ž<—î^æ
LEFT_SPACE = RIGHT_SPACE = 16 # ?ŋ%ºE,ì"»'è-³Ž<—î^æ
#-----
# ?œ ŒöŠJfCf"fXf^f"fX•İ?"
#-----
attr_accessor :dot_moving # fhfbfg^Ú"®,İŒü,«?inil:fhfbfg^Ú"®,μ,Ä,¢,È,¢ 0:Ž~,Ü,Ä,Ä,¢,é?j
#-----
# ?œ ?%ºŠú%º»
#-----
def initialize
super
# fhfbfg^Ú"®ŠJŽn?A—$,¿Ž~,Ü,¢,¹,é
@dot_moving = 0
end
#-----
# ?œ ^Ú"®'†"»'è
#-----
def moving?(by_update = false)
# ŒÄ,Ŧ?º,μŒ³,ªupdate^ÈŠO,Ä?A,©,Äfhfbfg^Ú"®'†,İ,Œ,«
if !by_update and @dot_moving
# —$,¿Ž~,Ü,Ä,Ä,¢,È,~,é,İ^Ú"®'†
return @dot_moving != 0
else
# fhfbfg^Ú"®'†,Ä,È,?A,©,Ä~—??Ä•W,ŒŽÄ?Ä•W,ª^á,Ä,Ä,¢,é,İ^Ú"®'†
return (!@dot_moving and (@real_x != @x * 128 or @real_y != @y * 128))
end
end
#-----
# ?œ ^Ú"®f<?[fg,ì<?Š
# move_route : ?V,μ,¢^Ú"®f<?[fg
#-----
def force_move_route(move_route)
# —$,¿Ž~,Ü,¢,¹,é
@dot_moving = 0
super(move_route)
end
#-----
# ?œ ftfŒ?[f€?X?V
#-----
def update
# f??[fJf<•İ?",É?Ä•W,ð<L%º~
last_real_x, last_real_y = @real_x, @real_y
last_x, last_y = @x, @y
# f??[fJf<•İ?",É^Ú"®'†,©,Ç,¤,©,ð<L%º~
last_moving = moving?(true)
# ^Ú"®'†?AfCxf"fƒŽÄ?s'†?A^Ú"®f<?[fg<?Š'†?A
# f?fbfZ?[fWfEfBf"fhfE•Ž'†,ì,¢,,é,Ä,à,È,¢?ê?‡
unless moving?(true) or $game_system.map_interpreter.running? or
@move_route_forcing or $game_temp.message_window_showing
# fhfbfg^Ú"®
dot_move(Input.dir4)
end
super
# fLff%ºfNf^?[,ª%º,É^Ú"®,μ?A,©,Ä%ºæ-Ê?ã,ì^É'u,ª'†%º>,æ,è%º,ì?ê?‡
if @real_y > last_real_y and @real_y - $game_map.display_y > CENTER_Y
# f}fbfv,ð%º,ÉfXfNf??[f<
```

```

$game_map.scroll_down(@real_y - last_real_y)
end
# fLfff%0fNf^?[ ,^?¶,É`Ú“@,μ?A,©,Â%0æ-Ê?ã,Ì`Ê`u,^`+%0>,æ,è?¶,Ì?ê?‡
if @real_x < last_real_x and @real_x - $game_map.display_x < CENTER_X
# f}fbfv,ð?¶,ÉfXfNf??[f<
$game_map.scroll_left(last_real_x - @real_x)
end
# fLfff%0fNf^?[ ,^%0E,É`Ú“@,μ?A,©,Â%0æ-Ê?ã,Ì`Ê`u,^`+%0>,æ,è%0E,Ì?ê?‡
if @real_x > last_real_x and @real_x - $game_map.display_x > CENTER_X
# f}fbfv,ð%0E,ÉfXfNf??[f<
$game_map.scroll_right(@real_x - last_real_x)
end
# fLfff%0fNf^?[ ,^?ã,É`Ú“@,μ?A,©,Â%0æ-Ê?ã,Ì`Ê`u,^`+%0>,æ,è?ã,Ì?ê?‡
if @real_y < last_real_y and @real_y - $game_map.display_y < CENTER_Y
# f}fbfv,ð?ã,ÉfXfNf??[f<
$game_map.scroll_up(last_real_y - @real_y)
end
# `Ú“@`†,Â,Í,È,¢?ê?‡
unless moving?(true)
# ?Â•W,^?X?V,^3,ê,½,©?Afhfbbfg`Ú“@`†,Â,È,‘O%0ñfvfœfCf,,?[ ,^`Ú“@`†,¾,Á,½?ê?‡
if last_x != @x or last_y != @y or (!@dot_moving and last_moving)
# “~`Ê`u,ÌfCfxf“fg,Æ,Ì?Ú?G,É,æ,éCfxf“fg<N“@”»`è
result = check_event_trigger_here([1,2])
# <N“@,μ,½fCfxf“fg,^,È,¢?ê?‡
if result == false
# fffofbfbOf,?[fh,^ ON ,©,Â CTRL fL?[ ,^%0ÿ,^3,ê,Â,¢,é?ê?‡,ð?œ,«
unless $DEBUG and Input.press?(Input::CTRL)
# fGf“fJfEf“fg fJfEf“fgf_fEf“
if @encounter_count > 0
@encounter_count -= 1
end
end
# <N“@,μ,½fCfxf“fg,^, ,é?ê?‡
else
# —$,¿Ž~,Û,é?Afhfbbfg`Ú“@,μ,Â,¢,¢,È,ç,»,Ì,Û,Û
@dot_moving = @dot_moving ? 0 : nil
end
end
end
# C f{f^f“,^%0ÿ,^3,ê,½?ê?‡
if Input.trigger?(Input::C)
# “~`Ê`u,~,æ,Ñ?^3-Ê,ÌfCfxf“fg<N“@”»`è
check_event_trigger_here([0])
check_event_trigger_there([0,1,2])
end
end
end
#-----
# ?œ `Ú“@f^fCfv : fJfXf^f€
#-----
def move_type_custom
# `âŽ~`†,Â,È,~,ê,Ì`†f
if jumping? or moving?
return
end
# `Ú“@fRf}f“fh,ÌfŠfXfg,Ì?Âœã,É“ž`B,·,é,Û,Âf<?[fv
while @move_route_index < @move_route.list.size
# `Ú“@fRf}f“fh,ðŽæ“¾
command = @move_route.list[@move_route_index]
# fRf}f“fhfR?[fh 0 "Ô (fŠfXfg,Ì?Âœã) ,Ì?ê?‡
if command.code == 0
# fLfVfVf‡f“ [“@?i,ðœ],è•Ô,·] ,^ ON ,Ì?ê?‡
if @move_route.repeat
# `Ú“@f<?[fg,ÌfCf“fffbfNfX,ð?Â?%0,É-β,·
@move_route_index = 0
end
# fLfVfVf‡f“ [“@?i,ðœ],è•Ô,·] ,^ OFF ,Ì?ê?‡
unless @move_route.repeat
# `Ú“@f<?[fg<?Š`†,Ì?ê?‡
if @move_route_forcing and not @move_route.repeat
# `Ú“@f<?[fg,Ì<?Š,ð%0ð?œ
@move_route_forcing = false
# fLfŠfWfif<,Ì`Ú“@f<?[fg,ð•œ<A
@move_route = @original_move_route

```

```

@move_route_index = @original_move_route_index
@original_move_route = nil
end
# 'âŽ~fjEf"fg,ðfNfŠfA
@stop_count = 0
end
return
end
# 'Ú"®ŒnfRf}f"fh (%oº,É^Ú"®?'fWfff"fv) ,l?ê?‡
if command.code <= 14
# fhfbfg^Ú"®¿?Š?l—¹
@dot_moving = nil
# fRf}f"fhfR?[fh,Â•ªŠð
case command.code
when 1 # %oº,É^Ú"®
move_down
when 2 # ?¶,É^Ú"®
move_left
when 3 # %oE,É^Ú"®
move_right
when 4 # ?ã,É^Ú"®
move_up
when 5 # ?¶%oº,É^Ú"®
move_lower_left
when 6 # %oE%oº,É^Ú"®
move_lower_right
when 7 # ?¶?ã,É^Ú"®
move_upper_left
when 8 # %oE?ã,É^Ú"®
move_upper_right
when 9 # f%of"f_f€,É^Ú"®
move_random
when 10 # fvfŒfCf,,?[ ,É<β,Ã,
move_toward_player
when 11 # fvfŒfCf,,?[ ,©,ç%o",',©,é
move_away_from_player
when 12 # ^ê•à'O?i
move_forward
when 13 # ^ê•àŒĖā'p
move_backward
when 14 # fWfff"fv
jump(command.parameters[0], command.parameters[1])
end
# flfvfVf‡f" [^Ú"®,Ã,«,È,¢?ê?‡,Í-³Ž<] ,ª OFF ,Ã?A^Ú"®Ž, "s,l?ê?‡
if not @move_route.skippable and not moving? and not jumping?
return
end
end
@move_route_index += 1
return
end
# fEfFfCfkg,l?ê?‡
if command.code == 15
# fEfFfCfkgfjEf"fg,ð?Ý'è
@wait_count = command.parameters[0] * 2 - 1
@move_route_index += 1
return
end
# ŒÛ,«•l?XŒn,lfRf}f"fh,l?ê?‡
if command.code >= 16 and command.code <= 26
# fRf}f"fhfR?[fh,Â•ªŠð
case command.code
when 16 # %oº,ðŒÛ,
turn_down
when 17 # ?¶,ðŒÛ,
turn_left
when 18 # %oE,ðŒÛ,
turn_right
when 19 # ?ã,ðŒÛ,
turn_up
when 20 # %oE,É 90 "x%oñ"]
turn_right_90
when 21 # ?¶,É 90 "x%oñ"]
turn_left_90

```



```

when 22 # 180 "x%0ñ"]
turn_180
when 23 # %0E,©?¶,É 90 "x%0ñ"]
turn_right_or_left_90
when 24 # f%0f"f_f€É•ûÆü"]Š·
turn_random
when 25 # fvfÆfCf,,?[,l•û,ðÆü,
turn_toward_player
when 26 # fvfÆfCf,,?[,l<t,ðÆü,
turn_away_from_player
end
@move_route_index += 1
return
end
# ,»,l'¼,lfRf}f"f_h,l?ê?‡
if command.code >= 27
# fRf}f"f_hfR?[f_h,Å•ªŠð
case command.code
when 27 # fXfCfbf` ON
$game_switches[command.parameters[0]] = true
$game_map.need_refresh = true
when 28 # fXfCfbf` OFF
$game_switches[command.parameters[0]] = false
$game_map.need_refresh = true
when 29 # `Ú"®'¬"x,l•l?X
@move_speed = command.parameters[0]
when 30 # `Ú"®•p"x,l•l?X
@move_frequency = command.parameters[0]
when 31 # `Ú"®ŽžfAfjf? ON
@walk_anime = true
when 32 # `Ú"®ŽžfAfjf? OFF
@walk_anime = false
when 33 # 'âŽ~ŽžfAfjf? ON
@step_anime = true
when 34 # 'âŽ~ŽžfAfjf? OFF
@step_anime = false
when 35 # Æü,«ÆÁ'è ON
@direction_fix = true
when 36 # Æü,«ÆÁ'è OFF
@direction_fix = false
when 37 # ,·,è"²,¬ ON
@through = true
when 38 # ,·,è"²,¬ OFF
@through = false
when 39 # ?Á'O-Ê,É•Ž! ON
@always_on_top = true
when 40 # ?Á'O-Ê,É•Ž! OFF
@always_on_top = false
when 41 # fOf%0ftfBfbfN•l?X
@tile_id = 0
@character_name = command.parameters[0]
@character_hue = command.parameters[1]
if @original_direction != command.parameters[2]
@direction = command.parameters[2]
@original_direction = @direction
@prelock_direction = 0
end
if @original_pattern != command.parameters[3]
@pattern = command.parameters[3]
@original_pattern = @pattern
end
when 42 # •s"§-¾"x,l•l?X
@opacity = command.parameters[0]
when 43 # ?‡?¬•û-@,l•l?X
@blend_type = command.parameters[0]
when 44 # SE,l%0%0't
$game_system.se_play(command.parameters[0])
when 45 # fXfNfŠfvg
result = eval(command.parameters[0])
end
@move_route_index += 1
end
end

```



```

end
#-----
# ?œ ftfŒ?[f€?X?V (^Ú“®)
#-----
def update_move
# fhfbfg^Ú“®’†,Ă-³,¢,È,ç~—??Ă•W?X?V
unless @dot_moving
# ^Ú“®‘¬“x,©,çf}fbfv?Ă•WÆn,Ă,Ì^Ú“®←—£,É•ĬŠ·
distance = 2 ** @move_speed
# ~—??Ă•W,ªŽĂ?Ă•W,æ,è%ºº,Ì?ê?‡
if @y * 128 > @real_y
# %ºº,É^Ú“®
@real_y = [@real_y + distance, @y * 128].min
end
# ~—??Ă•W,ªŽĂ?Ă•W,æ,è?¶,Ì?ê?‡
if @x * 128 < @real_x
# ?¶,É^Ú“®
@real_x = [@real_x - distance, @x * 128].max
end
# ~—??Ă•W,ªŽĂ?Ă•W,æ,è%ººE,Ì?ê?‡
if @x * 128 > @real_x
# %ººE,É^Ú“®
@real_x = [@real_x + distance, @x * 128].min
end
# ~—??Ă•W,ªŽĂ?Ă•W,æ,è?ā,Ì?ê?‡
if @y * 128 < @real_y
# ?ā,É^Ú“®
@real_y = [@real_y - distance, @y * 128].max
end
end
# ^Ú“®ŽžfAfjf?,ª ON ,Ì?ê?‡
if @walk_anime
# fAfjf?fJfEf“fg,ð 1.5 ‘?,ā,·
@anime_count += 1.5
# ^Ú“®ŽžfAfjf?,ª OFF ,Ă?A’āŽ~ŽžfAfjf?,ª ON ,Ì?ê?‡
elsif @step_anime
# fAfjf?fJfEf“fg,ð 1 ‘?,ā,·
@anime_count += 1
end
end
#-----
# ?œ fhfbfg^Ú“®?i’Ç%ººĂ?j
# dir : •ûŒü?i0,2,4,6,8?j
#-----
def dot_move(dir)
@dot_moving = dir
# —$,¿Ž~,Û,Ă,Ā,¢,é,©fhfbfg^Ú“®’†,Ă,È,¢,È,ç,·,®,ÉfŠf^?[f“
if dir == 0 or !dir
return
end
# ?Ă•W,Ì•Û‘¶
last_real_x, last_real_y = @real_x, @real_y
last_x, last_y = @x, @y
# ^Ú“®‘¬“x,©,çf}fbfv?Ă•WÆn,Ă,Ì^Ú“®←—£,É•ĬŠ·
distance = 2 ** @move_speed
# ?Ă•W,ð?X?V?A?i?s•ûŒü‘▯,Ì“ñ←,Ì?Ă•W,ð“¾,é
case dir
when 2
turn_down
@real_y += distance
dx, dy = 0, 1
x1, y1 = @real_x + LEFT_SPACE, @real_y + 127 - BOTTOM_SPACE
x2, y2 = @real_x + 127 - RIGHT_SPACE, y1

when 4
turn_left
@real_x -= distance
dx, dy = -1, 0
x1, y1 = @real_x + LEFT_SPACE, @real_y + TOP_SPACE
x2, y2 = x1, @real_y + 127 - BOTTOM_SPACE
when 6
turn_right
@real_x += distance

```

```
dx, dy = 1, 0
x1, y1 = @real_x + 127 - RIGHT_SPACE, @real_y + TOP_SPACE
x2, y2 = x1, @real_y + 127 - BOTTOM_SPACE

when 8
turn_up
@real_y -= distance
dx, dy = 0, -1
x1, y1 = @real_x + LEFT_SPACE, @real_y + TOP_SPACE
x2, y2 = @real_x + 127 - RIGHT_SPACE, y1

end
# “ñ<÷,ª,¢,,ê,àfLfff%0fNf^,ì`—??À•W,©,ç,í,Ý?o,µ,Ä,¢,é?ê?‡
if (x1 / 128 != @x or y1 / 128 != @y) and (x2 / 128 != @x or y2 / 128 != @y)
# ?i?s•ûŒü,ì“ñ<÷,ª,Æ,à,É’Ê?s%0Â”,©’²,×,é
@x, @y = x1 / 128 - dx, y1 / 128 - dy
pass1 = passable?(@x, @y, dir)
@x, @y = x2 / 128 - dx, y2 / 128 - dy
pass2 = passable?(@x, @y, dir)
# ?Â•W,ðŒ³,É–β,·
@x, @y = last_x, last_y
# ‘Ê?s%0Â”,È?ê?‡
if pass1 and pass2
# ?V,µ,¢ŽÂ?Â•W,ìŒvŽZ
@x, @y = (@real_x + 64) / 128, (@real_y + 64) / 128
# ?Â•W,ª•Ĭ%0»,µ,½,Ê,ç•à?”?X?V
if last_x != @x or last_y != @y
increase_steps
end
# ‘Ê?s•s%0Â”,È?ê?‡
else
# —§,¿Ž~,Û,é
@dot_moving = 0
# ŽÂ?Â•W,ð<,?³,·,é
case dir
when 2
@real_y = [(last_real_y + 64) / 128 * 128, last_real_y].max
# %0E’▣,ÉŠñ,Á,Ä,¢,é,Æ,«
if @real_x / 128 < @x
# %0E,ĭfCfxf“fg,ð?æ,É?^—?
x1, x2 = x2, x1
end
when 4
@real_x = [(last_real_x + 64) / 128 * 128, last_real_x].min
# %0º‘▣,ÉŠñ,Á,Ä,¢,é,Æ,«
if @real_y / 128 < @y
# %0º,ĭfCfxf“fg,ð?æ,É?^—?
y1, y2 = y2, y1
end
when 6
@real_x = [(last_real_x + 64) / 128 * 128, last_real_x].max
# %0º‘▣,ÉŠñ,Á,Ä,¢,é,Æ,«
if @real_y / 128 < @y
# %0º,ĭfCfxf“fg,ð?æ,É?^—?
y1, y2 = y2, y1
end
when 8
@real_y = [(last_real_y + 64) / 128 * 128, last_real_y].min
# %0E’▣,ÉŠñ,Á,Ä,¢,é,Æ,«
if @real_x / 128 < @x
# %0E,ĭfCfxf“fg,ð?æ,É?^—?
x1, x2 = x2, x1
end
end
# ?Ú?GfCfxf“fg,ĭ<N“®”»’è?i^ê,Â<N“®,µ,½,ç,à,▣^ê,Â,ĭ–³Ž<?j
check_event_trigger_touch(x1 / 128, y1 / 128) or
check_event_trigger_touch(x2 / 128, y2 / 128)
end
end
end
end

class Scene_Map
```



```
def update
# f<?[fv
loop do
# f}fbfv?AfCf"f^fvfŠf^?AfvfŒfCf,,?[,ì?‡,É?X?V
# (,±,ì?X?V?‡?~,Í?AfCfxf"fƒg,ðŽÀ?s,·,é?ðŒ?,ª-ž,½,³,ê,Ä,¢,é,Æ,«,É
# fvfŒfCf,,?[,É~ê?u^Ű"@,·,é<@%öï,ð—^,!,È,¢,È,Ç,ì—?—R,Ä?d—v)
$game_map.update
$game_system.map_interpreter.update
$game_player.update
# fVfXfef€(f^fCf)?[)?A%œ-Ê,ð?X?V
$game_system.update
$game_screen.update
# fvfŒfCf,,?[,ì?ê?Š^Ű"@'†,Ä,È,¬,ê,Îf<?[fv,ð'†'f
unless $game_temp.player_transferring
break
end
# ?ê?Š^Ű"@,ðŽÀ?s
transfer_player
# fƒf%of"fWfVf‡f"?^—?'†,ì?ê?‡?Af<?[fv,ð'†'f
if $game_temp.transition_processing
break
end
end
# fXfvf%ofCfƒƒZfbfƒg,ð?X?V
@spriteset.update
# f?fbfZ?[fWfEfBf"fhfE,ð?X?V
@message_window.update
# fQ?[f€fI?[fo?[,ì?ê?‡
if $game_temp.gameover
# fQ?[f€fI?[fo?[%œ-Ê,É?Ø,è'Ö,|
$scene = Scene_Gameover.new
return
end
# f^fCfƒƒ<%œ-Ê,É-ß,·?ê?‡
if $game_temp.to_title
# f^fCfƒƒ<%œ-Ê,É?Ø,è'Ö,|
$scene = Scene_Title.new
return
end
# fƒf%of"fWfVf‡f"?^—?'†,ì?ê?‡
if $game_temp.transition_processing
# fƒf%of"fWfVf‡f"?^—?'†ftf%ofO,ðfNfŠfA
$game_temp.transition_processing = false
# fƒf%of"fWfVf‡f"ŽÀ?s
if $game_temp.transition_name == ""
Graphics.transition(20)
else
Graphics.transition(40, "Graphics/Transitions/" +
$game_temp.transition_name)
end
end
# f?fbfZ?[fWfEfBf"fhfE•Ž!'†,ì?ê?‡
if $game_temp.message_window_showing
return
end
# fGf"fJfEf"fƒfJfEf"fƒg,ª 0 ,Ä?AfGf"fJfEf"fƒƒŠfXfƒg,ª<ó,Á,Í,È,¢?ê?‡
if $game_player.encounter_count == 0 and $game_map.encounter_list != []
# fCfxf"fƒŽÀ?s'†,©fGf"fJfEf"fƒ<ÖŽ~'†,Ä,È,¬,ê,Î
unless $game_system.map_interpreter.running? or
$game_system.encounter_disabled
# fƒf<?[fv,ðŒ~è
n = rand($game_map.encounter_list.size)
troop_id = $game_map.encounter_list[n]
# fƒf<?[fv,ª—LŒØ,È,ç
if $data_troops[troop_id] != nil
# fofƒf<ŒÄ,Ŧ?o,µftf%ofO,ðfZfbfƒg
$game_temp.battle_calling = true
$game_temp.battle_troop_id = troop_id
$game_temp.battle_can_escape = true
$game_temp.battle_can_lose = false
$game_temp.battle_proc = nil
end
end
end
```



```
end
# B f{f^f",ª%Ÿ,³,ê,½?ê?‡
if Input.trigger?(Input::B)
# fCfxf"fġŽĀ?s'†,©f?fjff...?[\ŒŽ~'†,Ā,È,~,ê,Ĭ
unless $game_system.map_interpreter.running? or
$game_system.menu_disabled
# f?fjff...?[ŒĀ,Ń?o,µftf%fo,Æ SE %'tftf%fo,ðfZfbfg
$game_temp.menu_calling = true
$game_temp.menu_beep = true
end
end
# fffofbfOf,?[fh,ª ON ,©,Ā F9 fL?[,ª%Ÿ,³,ê,Ā,¢,é?ê?‡
if $DEBUG and Input.press?(Input::F9)
# fffofbfOĒĀ,Ń?o,µftf%fo,ðfZfbfg
$game_temp.debug_calling = true
end
# fhfbfg^Ů"®'†,Ā,È,fvfŒfCf,,?[,ì^Ů"®'†,Ā, ,é?ê?‡,ð?œ,«
unless !$game_player.dot_moving and $game_player.moving?
# ŠeŽí%œ-Ê,ĬĒĀ,Ń?o,µ,ðŽĀ?s
if $game_temp.battle_calling
call_battle
elsif $game_temp.shop_calling
call_shop
elsif $game_temp.name_calling
call_name
elsif $game_temp.menu_calling
call_menu
elsif $game_temp.save_calling
call_save
elsif $game_temp.debug_calling
call_debug
end
end
end
end
end
```