

RMVA 跳过启动界面并把启动背景图当作启动 Logo[RGSS][脚本整理]

#制作 by: 洮源芝罘

#本脚本用于直接跳过启动界面 （自动 NEW GAME）

#本脚本还可以将通过属性界面更改的标题背景图作为 logo 在屏幕上显示几秒

```
class Scene_Title < Scene_Base
  #-----
  # * Start Processing
  #-----

  def start
    super
    SceneManager.clear
    Graphics.freeze
    create_background
    create_foreground
    create_command_window
    play_title_music
    loop do
      for i in 1..10000
      end
      command_new_game_sp
      break
    end
  end
end

def command_new_game_sp
  DataManager.setup_new_game
  close_command_window
  fadeout_all
  $game_map.autoplay
  SceneManager.goto(Scene_Map)
end
```

```
  #-----
  # * Get Transition Speed
  #-----

  def transition_speed
    return 20
  end
  #-----
  # * Termination Processing
  #-----

  def terminate
    super
```

```

SceneManager.snapshot_for_background
dispose_background
dispose_foreground
end
#-----
# * Create Background
#-----
def create_background
  @sprite1 = Sprite.new
  @sprite1.bitmap = Cache.title1($data_system.title1_name)
  @sprite2 = Sprite.new
  @sprite2.bitmap = Cache.title2($data_system.title2_name)
  center_sprite(@sprite1)
  center_sprite(@sprite2)
end
#-----
# * Create Foreground
#-----
def create_foreground
  @foreground_sprite = Sprite.new
  @foreground_sprite.bitmap = Bitmap.new(Graphics.width, Graphics.height)
  @foreground_sprite.z = 100
  draw_game_title if $data_system.opt_draw_title
end
#-----
# * Draw Game Title
#-----
def draw_game_title
  @foreground_sprite.bitmap.font.size = 48
  rect = Rect.new(0, 0, Graphics.width, Graphics.height / 2)
  @foreground_sprite.bitmap.draw_text(rect, $data_system.game_title, 1)
end
#-----
# * Free Background
#-----
def dispose_background
  @sprite1.bitmap.dispose
  @sprite1.dispose
  @sprite2.bitmap.dispose
  @sprite2.dispose
end
#-----
# * Free Foreground
#-----

```

```

def dispose_foreground
  @foreground_sprite.bitmap.dispose
  @foreground_sprite.dispose
end
#-----
# * Move Sprite to Screen Center
#-----
def center_sprite(sprite)
  sprite.ox = sprite.bitmap.width / 2
  sprite.oy = sprite.bitmap.height / 2
  sprite.x = Graphics.width / 2
  sprite.y = Graphics.height / 2
end
#-----
# * Create Command Window
#-----
def create_command_window
  @command_window = Window_TitleCommand.new
  @command_window.set_handler(:new_game, method(:command_new_game))
  @command_window.set_handler(:continue, method(:command_continue))
  @command_window.set_handler(:shutdown, method(:command_shutdown))
end
#-----
# * Close Command Window
#-----
def close_command_window
  @command_window.close
  update until @command_window.close?
end
#-----
# * [New Game] Command
#-----
def command_new_game
  DataManager.setup_new_game
  close_command_window
  fadeout_all
  $game_map.autoplay
  SceneManager.goto(Scene_Map)
end
#-----
# * [Continue] Command
#-----
def command_continue
  close_command_window

```

```
    SceneManager.call(Scene_Load)
end
#-----
# * [Shut Down] Command
#-----
def command_shutdown
    close_command_window
    fadeout_all
    SceneManager.exit
end
#-----
# * Play Title Screen Music
#-----
def play_title_music
    $data_system.title_bgm.play
    RPG::BGS.stop
    RPG::ME.stop
end
end
```

