

ROTATIO

```
global gradang, cursorDiff, gremnantList, gwhichGame,
gRotaConst, gvolume

on startMovie
  set the soundLevel to gvolume
  set the volume of sound 1 to 250
  set the volume of sound 2 to 250
  set cursorDiff to point(0,0)
  set gremnantList to []
  set gwhichGame to 3
  set the visible of sprite 5 to false
  set gRotaConst to 0
end

on radarCircle

  put the rotation of sprite 11 into rot
  set the rotation of sprite 11 to rot+1
  set the rotation of sprite 12 to rot-10
  set the rotation of sprite 13 to rot-30
  set the rotation of sprite 14 to rot-40
  set the rotation of sprite 15 to rot-50
  set the rotation of sprite 16 to rot-100
  - set the rotation of sprite 17 to (the rotation of sprite
  17)+11
  put rot mod 360 into rota
  case (rota) of
    90:
      set the member of sprite 20 to member(21,1)
      set the loc of sprite 20 to point(320,21)
      set the loc of sprite 21 to (the loc of sprite 20)
    180:
      set the member of sprite 20 to member(22,1)
      set the loc of sprite 20 to point(540,240)
      set the loc of sprite 21 to (the loc of sprite 20)
    270:
      set the member of sprite 20 to member(23,1)
      set the loc of sprite 20 to point(320,460)
      set the loc of sprite 21 to (the loc of sprite 20)
    0:
      set the member of sprite 20 to member(24,1)
      set the loc of sprite 20 to point(100,240)
      set the loc of sprite 21 to (the loc of sprite 20)
  otherwise
    set the member of sprite 21 to member(25,1)
    set the locH of sprite 20 to 1000
    set the rotation of sprite 21 to random(360)
  end case

  phonyCursor
end

on phonyCursor
  put pointToPolar (point(320,240)-the mouseLoc) into gradang
  set the rotation of sprite 100 to the angle of gradang
  set the loc of sprite 100 to the mouseLoc+cursorDiff

end

on centerStamp
  - sound fadeOut 1, 1200
  safepu 0
  updateStage
  if inside(the mouseLoc, the rect of sprite 8) then

    set the visible of sprite 100 to false
    set the rotation of sprite 8 to 0
    updateStage
    set the picture of member (10,1) = the picture of the
    stage
    crop (member (10,1), the rect of sprite 7)
    PolarToPoint
    set the member of sprite 17 to member(random(150),random
    (4)+1)+1)
    set cursorDiff to point(0,0)
    set the visible of sprite 100 to true
  end if
  if gwhichGame=2 then
    put random(255) into ra
    set the foreColor of sprite 5 to ra
    set the foreColor of sprite 16 to ra
  end if
  set cursorDiff to point(1000,1000)
  cursor -1
end

on targetRotate
  set cursorDiff to point(0,0)
  cursor 200
  repeat while the mouseDown
    radarCircle
    set the rotation of sprite 8 to (the rotation of sprite
    8)+the radius of gradang
    updateStage
  end repeat
  set cursorDiff to point(1000,1000)
  cursor -1

end

on fourSounds
  put the angle of gradang into ang
  if ang<90 then
    safepu member(random(3)+200,2)
  else
    if ang<180 then
      safepu member(random(3)+200,3)
    else
      if ang<270 then
        safepu member(random(3)+200,4)
      else
        if ang<360 then
          safepu member(random(3)+200,5)
        end if
      end if
    end if
  end if
  set cursorDiff to point(1000,1000)
  set gRotaConst to 0
  set the ink of sprite 17 to 4
  set the rotation of sprite 17 to 0
end

on safepu x
  puppetSound 1,0
  updateStage
  puppetSound 1,x
  updateStage
end

on PointToPolar thePoint
  - Clockwise; zero at 12:000
  set x to float(the locH of thePoint)
  set y to float(the locV of thePoint)

  set y to (y+(y=0))
  set angle to (atan(x/y) * -57.29577951)
  if y>0 then set angle to float(180 + angle)
  else if x<0 then set angle to float(360 + angle)
  set radius to float(sqrt(x*x + y*y))
  return [#angle: angle, #radius: radius]
end

on PolarToPoint
  put the rotation of sprite 11 into rot
  set the list to [#angle: (rot)+90, #radius: random(150)+70]
  set a to float(-(the angle of theList) / 57.29577951)
  set r to float(the radius of theList)
  set x to float(integer(r * sin(a)))
  set y to float(integer(r * cos(a)))
  set the loc of sprite 17 to point(x,y)+point(320,240)
end

on setHole
  repeat with w= 1 to 600
    set the member of sprite 6 to member(w, 8)
    updateStage
    set the picture of member (w,7) = the picture of the
    stage
  end repeat
end

on selectGame
  if inside(the mouseLoc, the rect of sprite 8) then
    cursor 200
    set cursorDiff to point(0,0)
  else
    if inside(the mouseLoc, the rect of sprite 5) then
      set the visible of sprite 5 to true
      set cursorDiff to point(1000,1000)
      cursor -1
      puppetPalette "vivid"
      set the member of sprite 16 to member(8,1)
      set the trails of sprite 16 to true
      set gwhichGame to 2
    else
      if inside(the mouseLoc,rect(80,0,564,481)) then
        set the visible of sprite 5 to false
        set cursorDiff to point(1000,1000)
        cursor -1
        set the member of sprite 16 to member(37,1)
        set the trails of sprite 16 to true
        set the ink of sprite 17 to 38
        set gRotaConst to -1
        set gwhichGame to 1
      else
        set the stageColor to the stageColor
        set the visible of sprite 5 to false
        set cursorDiff to point(1000,1000)
        cursor -1
        set the member of sprite 16 to member(8,1)
        set the foreColor of sprite 16 to 255
        set the trails of sprite 16 to true
        set gwhichGame to 3
      end if
    end if
  end if
end

on whichGame
  case gwhichGame of
    1:firstGame
    2:secondGame
    3:thirdGame
  end case
end

on firstGame
  put the foreColor of sprite 16 into forc
  if forc<239 then
    set the foreColor of sprite 16 to forc+1
  else
    set the foreColor of sprite 16 to 2
  end if

  set the rotation of sprite 17 to (the rotation of sprite
  17)-gRotaConst
end

on secondGame

end

on thirdGame

  set the foreColor of sprite 16 to 255
end

on rotSmall
  - put 640-the mouseH into mohh
  set the rotation of sprite 32 to (the rotation of sprite
  32)+100
end

on rotSmall2
  - put 640-the mouseH into mohh
  set the rotation of sprite 33 to (the rotation of sprite
  33)-100
end

on stopMovie
  unloadMember
  unload
end
```