

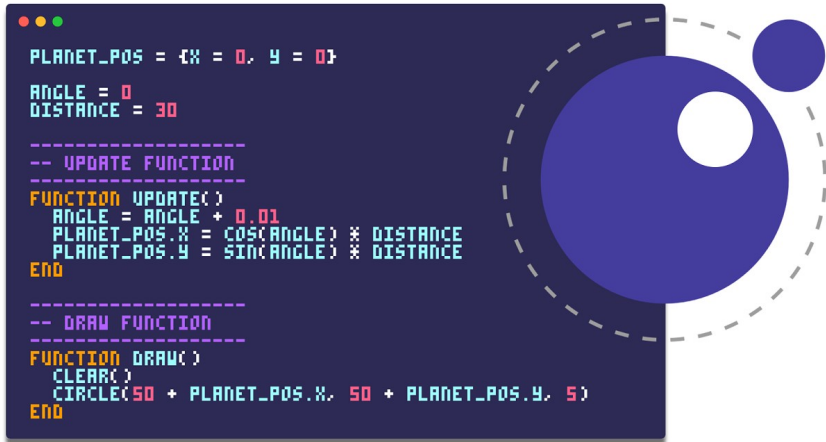
# Lua

Multi-paradigm programming language



# Overview

- Lua is a simple, lightweight, embeddable scripting language.
- Often used in game development, embedded systems, etc.
- Interpreted open source language.



```
PLANET_POS = {X = 0, Y = 0}
ANGLE = 0
DISTANCE = 30

-----
-- UPDATE FUNCTION
FUNCTION UPDATE()
    ANGLE = ANGLE + 0.01
    PLANET_POS.X = COS(ANGLE) * DISTANCE
    PLANET_POS.Y = SIN(ANGLE) * DISTANCE
END

-----
-- DRAW FUNCTION
FUNCTION DRAW()
    CLEAR()
    CIRCLE(50 + PLANET_POS.X, 50 + PLANET_POS.Y, 5)
END
```

# History

- Lua was created in 1993 by Roberto Ierusalimschy, Luiz Henrique de Figueiredo and Waldemar Celes
- Lua's predecessors were the data-description/configuration languages Simple Object Language (SOL) and data-entry language (DEL)



**Roberto Ierusalimschy**

# Key features

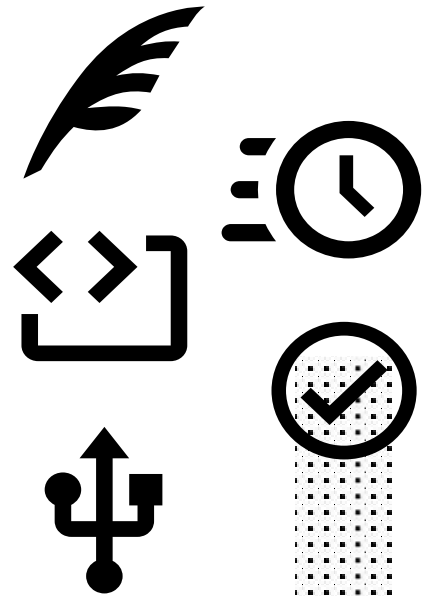
**Lightweight:** Small footprint (about 1MB in size)

**Fast:** Efficient performance, esp. with LuaJIT

**Embeddable:** Easily integrated into other languages

**Simple:** Easy to learn and use

**Portable:** Runs on virtually any platform

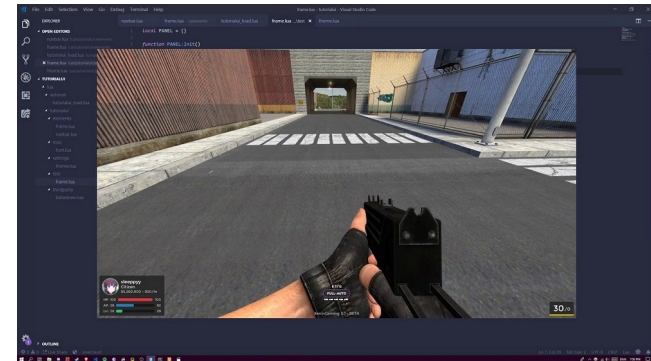
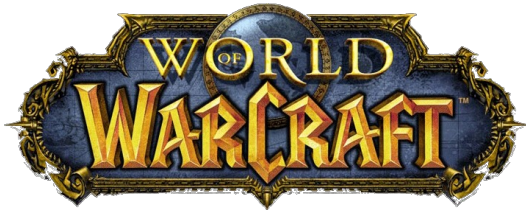


# Example

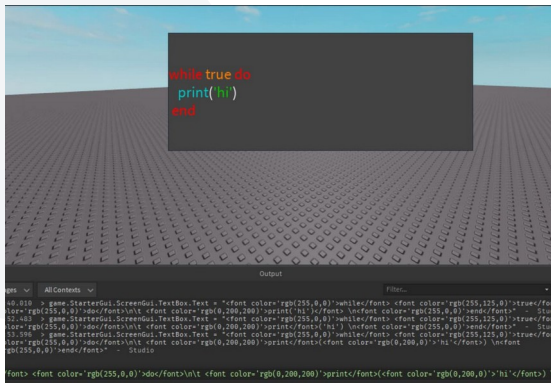
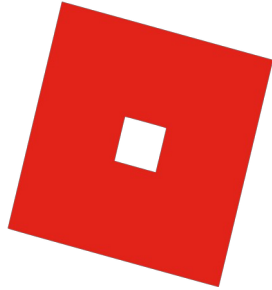
```
-- defines a factorial function
function fact (n)
  if n == 0 then
    return 1
  else
    return n * fact(n-1)
  end
end

print("enter a number:")
a = io.read("*number")      -- read a number
print(fact(a))
```

# Embedded Lua



# Embedded Lua



# Lua Comparison

- Python: More features, but heavier and slower
- JavaScript: Web-focused, not as lightweight or embeddable
- Lua: Minimalistic, fast, and embeddable

When to Use Lua:

- When performance and size matter
- For embedding in applications







**Thank you for your attention.**