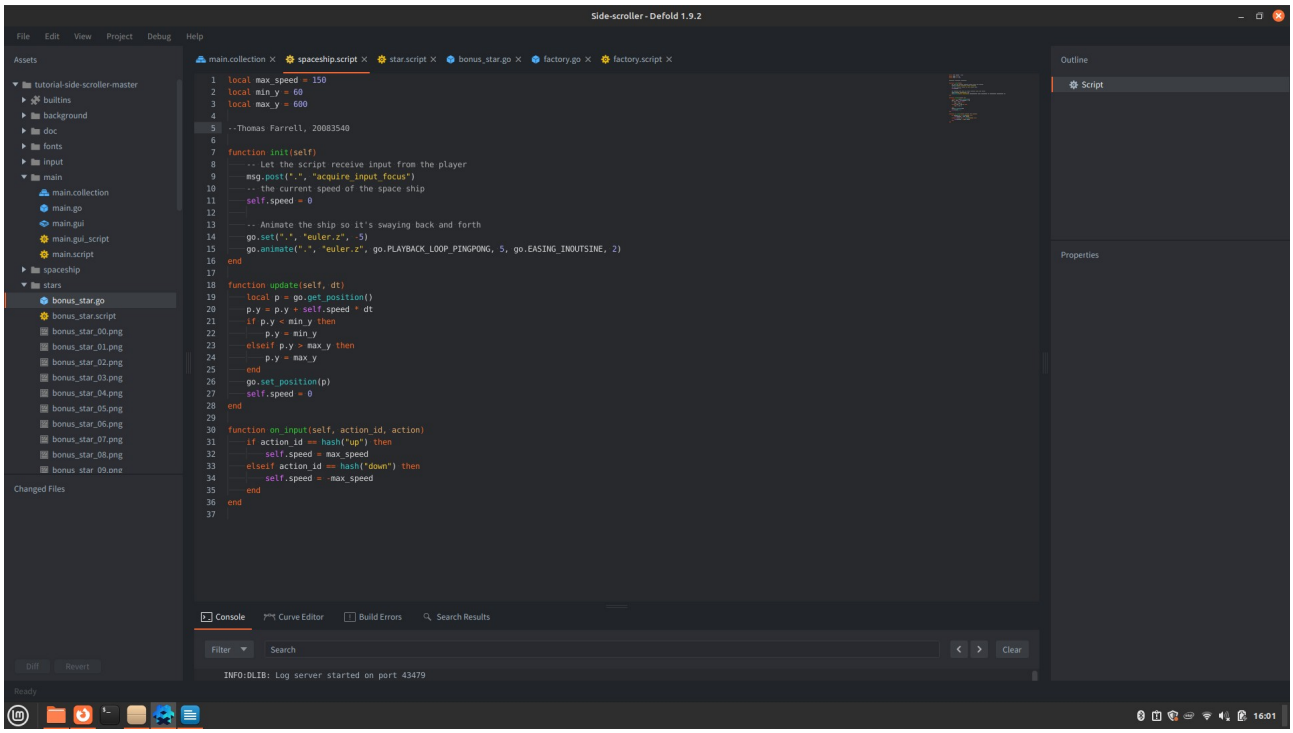
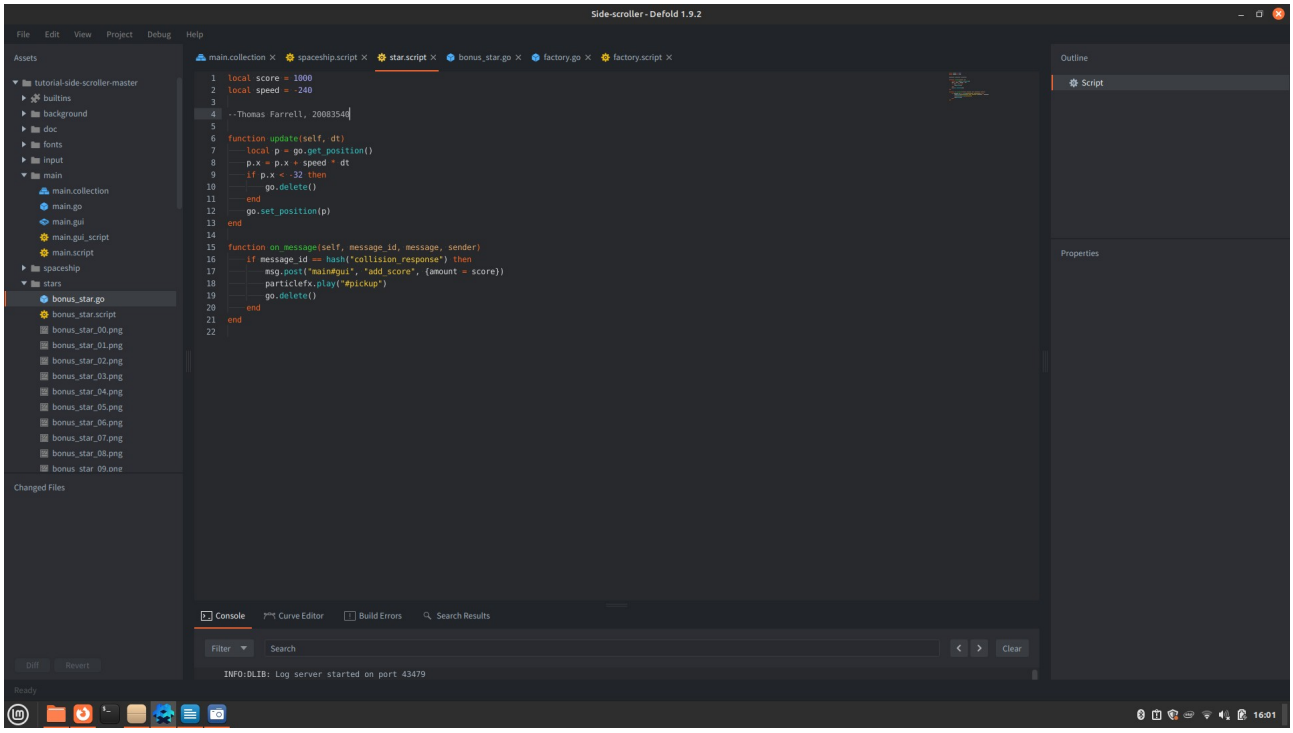
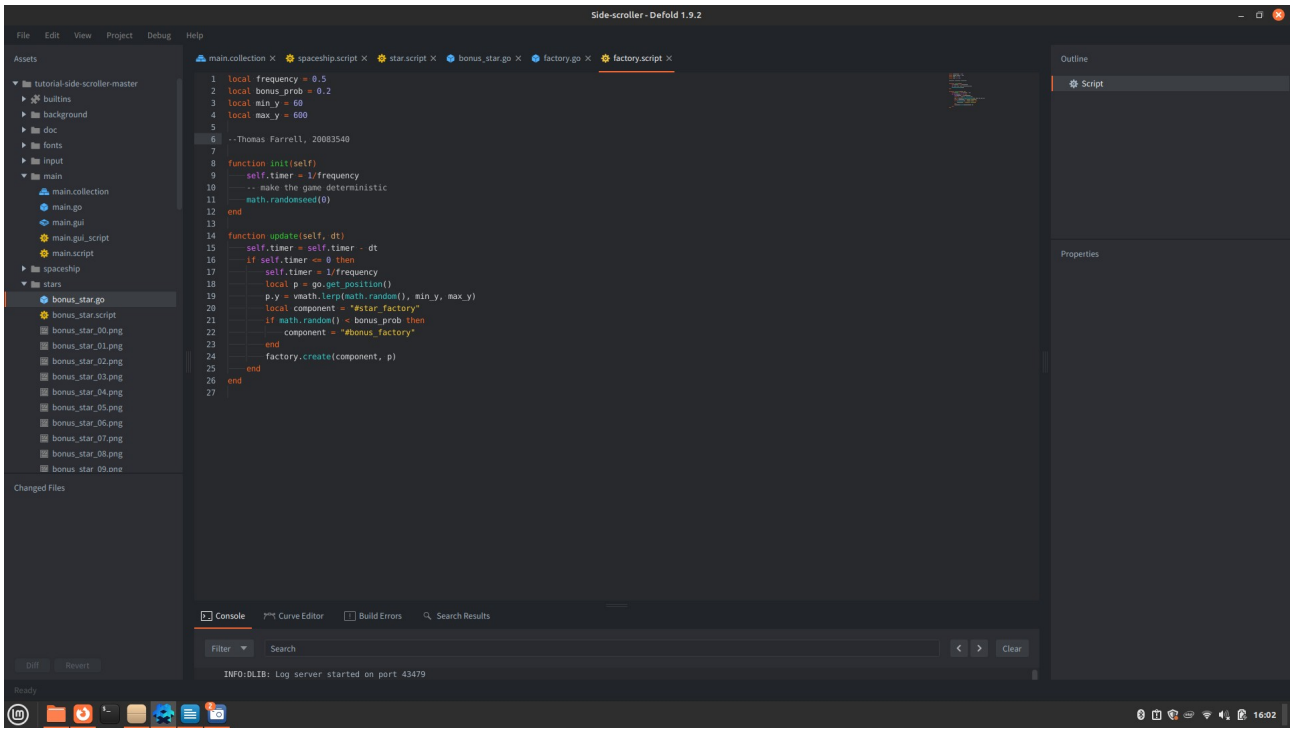
Lab Report

Name:	Thomas Farrell
Student Number:	20083540
Lab (Week) Number:	01
Date:	11/09/2024

Exercise 03.1

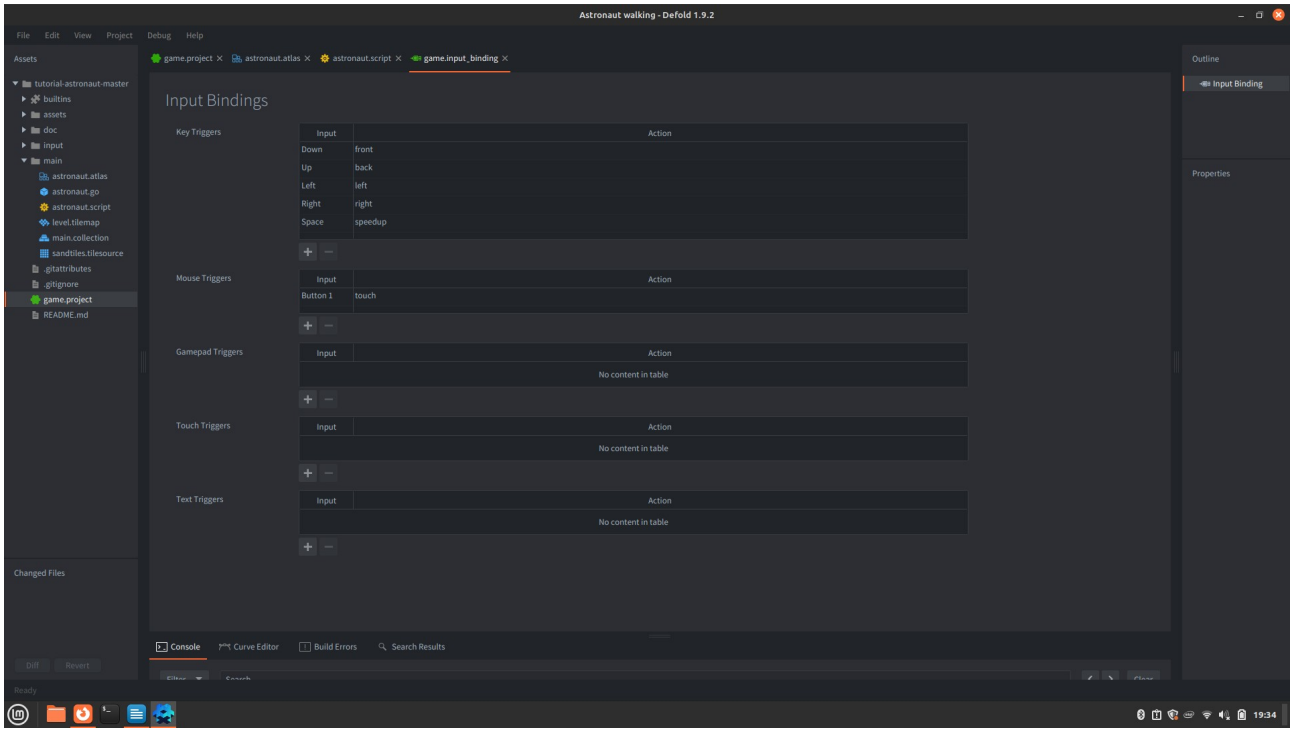


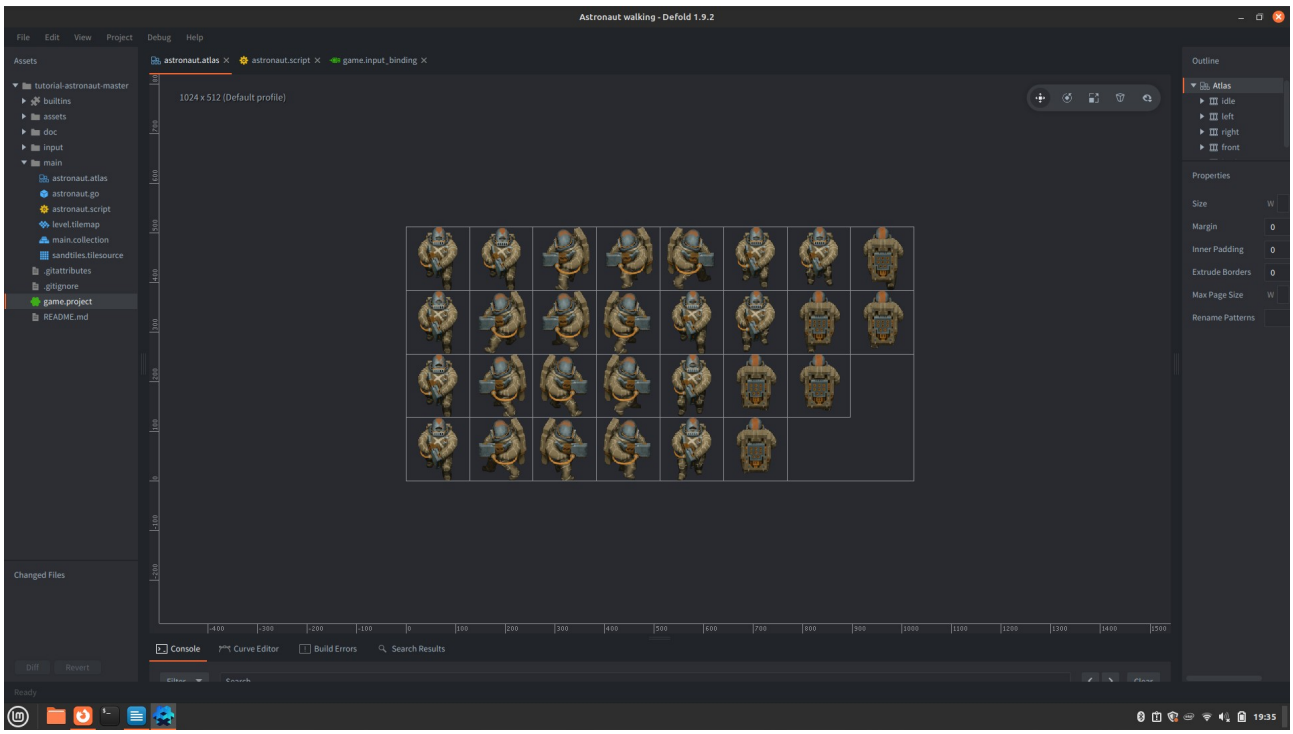

Side-scroller - Defold 1.9.2
File Edit View Project Debug Help
Assets
tutorial-side-scroller-master
builtins
background
doc
fonts
input
main
main.collection
main.go
main.gui
main.gui_script
main.script
spaceship
stars
bonus_star.go
bonus_star.script
bonus_star_00.png
bonus_star_01.png
bonus_star_02.png
bonus_star_03.png
bonus_star_04.png
bonus_star_05.png
bonus_star_06.png
bonus_star_07.png
bonus_star_08.png
bonus_star_09.png
Changed Files
Diff Revert
Ready
main.collection spaceship.script star.script bonus_star.go factory.go factory.script
local score = 1000
local speed = -240
--Thomas Farrell, 20083540
function update(self, dt)
local p = go.get_position()
p.x = p.x + speed * dt
if p.x < -32 then
go.delete()
end
go.set_position(p)
end
function on_message(self, message_id, message, sender)
if message_id == hash("collision_response") then
msg.post("main#gui", "add_score", {amount = score})
particlefx.play("#pickup")
go.delete()
end
end
Outline
Script
Properties
Console Curve Editor Build Errors Search Results
Filter Search Clear
INFO:DLIB: Log server started on port 43479
16:01

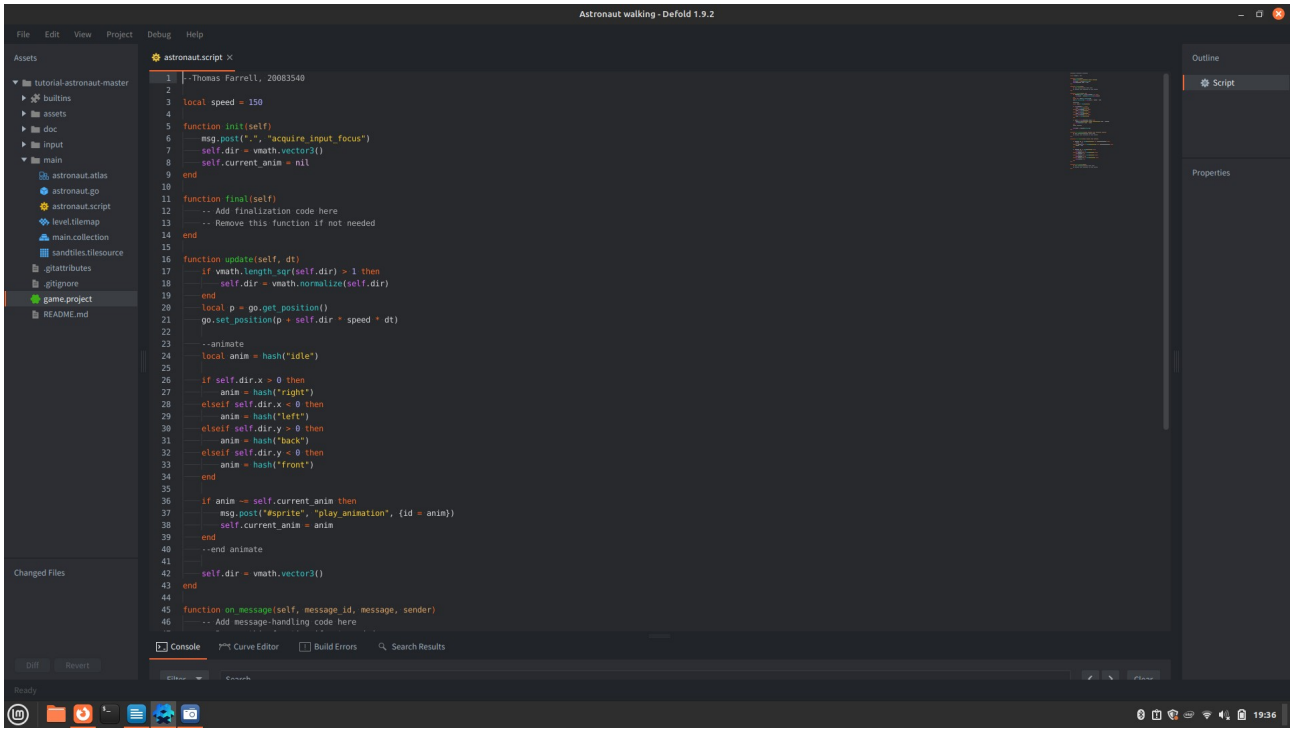

Side-scroller - Defold 1.9.2
File Edit View Project Debug Help
Assets
tutorial-side-scroller-master
builtins
background
doc
fonts
input
main
main.collection
main.go
main.gui
main.gui_script
main.script
spaceship
stars
bonus_star.go
bonus_star.script
bonus_star_00.png
bonus_star_01.png
bonus_star_02.png
bonus_star_03.png
bonus_star_04.png
bonus_star_05.png
bonus_star_06.png
bonus_star_07.png
bonus_star_08.png
bonus_star_09.png
Changed Files
Diff Revert
Ready
main.collection spaceship.script star.script bonus_star.go factory.go factory.script
local frequency = 0.5
local bonus_prob = 0.2
local min_y = 60
local max_y = 600
--Thomas Farrell, 20083540
function init(self)
self.timer = 1/frequency
-- make the game deterministic
math.randomseed(0)
end
function update(self, dt)
self.timer = self.timer - dt
if self.timer <= 0 then
self.timer = 1/frequency
local p = go.get_position()
p.y = vmath.lerp(math.random(), min_y, max_y)
local component = "#star_factory"
if math.random() < bonus_prob then
component = "#bonus_factory"
end
factory.create(component, p)
end
end
Outline
Script
Properties
Console Curve Editor Build Errors Search Results
Filter Search Clear
INFO:DLIB: Log server started on port 43479
16:02

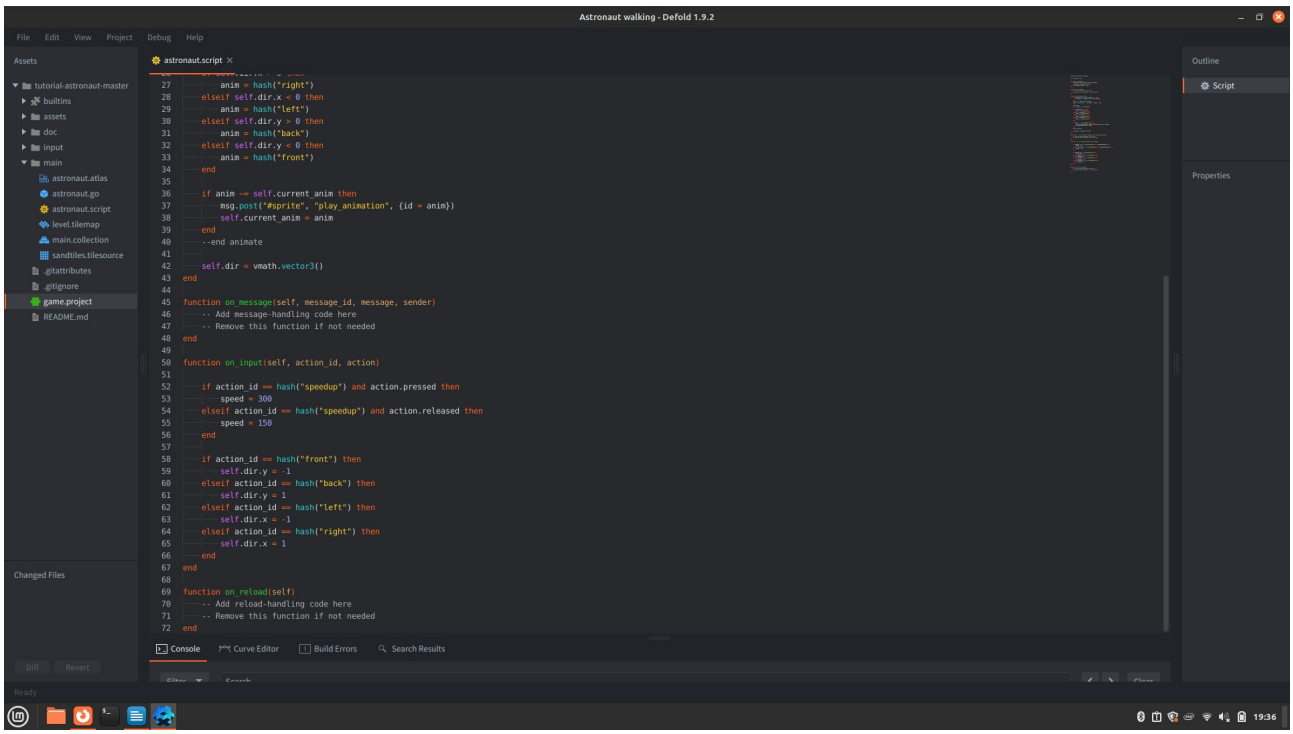
Side-scroller
7000
16:03

Exercise 04.1






Astronaut walking - Defold 1.9.2
File Edit View Project Debug Help
Assets
tutorial-astronaut-master
builtins
assets
doc
input
main
astronaut.atlas
astronaut.go
astronaut.script
level.tilemap
main.collection
sandtiles.tilesource
.gitattributes
.gitignore
game.project
README.md
Changed Files
Diff Revert
Ready
astronaut.script
--Thomas Farrell, 20083540
local speed = 150
function init(self)
msg.post(".", "acquire_input_focus")
self.dir = vmath.vector3()
self.current_anim = nil
end
function final(self)
-- Add finalization code here
-- Remove this function if not needed
end
function update(self, dt)
if vmath.length_sqr(self.dir) > 1 then
self.dir = vmath.normalize(self.dir)
end
local p = go.get_position()
go.set_position(p + self.dir * speed * dt)
--animate
local anim = hash("idle")
if self.dir.x > 0 then
anim = hash("right")
elseif self.dir.x < 0 then
anim = hash("left")
elseif self.dir.y > 0 then
anim = hash("back")
elseif self.dir.y < 0 then
anim = hash("front")
end
if anim ~= self.current_anim then
msg.post("#sprite", "play_animation", {id = anim})
self.current_anim = anim
end
--end animate
self.dir = vmath.vector3()
end
function on_message(self, message_id, message, sender)
-- Add message-handling code here
Console
Curve Editor
Build Errors
Search Results
Outline
Script
Properties
19:36


Astronaut walking - Defold 1.9.2
File Edit View Project Debug Help
Assets
tutorial-astronaut-master
builtins
assets
doc
input
main
astronaut.atlas
astronaut.go
astronaut.script
level.tilemap
main.collection
sandtiles.tilesource
.gitattributes
.gitignore
game.project
README.md
Changed Files
Diff Revert
Ready
astronaut.script
anim = hash("right")
elseif self.dir.x < 0 then
anim = hash("left")
elseif self.dir.y > 0 then
anim = hash("back")
elseif self.dir.y < 0 then
anim = hash("front")
end
if anim ~= self.current_anim then
msg.post("#sprite", "play_animation", {id = anim})
self.current_anim = anim
end
--end animate
self.dir = vmath.vector3()
end
function on_message(self, message_id, message, sender)
-- Add message-handling code here
-- Remove this function if not needed
end
function on_input(self, action_id, action)
if action_id == hash("speedup") and action.pressed then
speed = 300
elseif action_id == hash("speedup") and action.released then
speed = 150
end
if action_id == hash("front") then
self.dir.y = -1
elseif action_id == hash("back") then
self.dir.y = 1
elseif action_id == hash("left") then
self.dir.x = -1
elseif action_id == hash("right") then
self.dir.x = 1
end
end
function on_reload(self)
-- Add reload-handling code here
-- Remove this function if not needed
end
Console
Curve Editor
Build Errors
Search Results
Outline
Script
Properties
19:36

Conclusions (optional)

The game engine itself is exceptionally lightweight, which makes sense since it has compatibility with hardware going back a few decades. The small amount of Lua I've seen during these two tutorials is much more rudimenary than I expected; though I expect that to change at a rapid pace.

Looking forward to learning more of Lua, as I know that it's also heavily used in many modding communities, such as Skyrim, CyberPunk 2077 and even Final Fantasy 14.