# Creating script files for the Roblox project and zipping them for download.

import os, zipfile, textwrap, json, pathlib

BASE = "/mnt/data/roblox\_coinobby\_scripts"

os.makedirs(BASE, exist\_ok=True)

files = {

"README.txt": textwrap.dedent("""\

Roblox Coin Obby - Ready-to-Copy Script Files

=============================================

Files included here map to exact locations inside Roblox Studio (use the same names).

How to use:

1. Open Roblox Studio -> File -> New -> Baseplate.

2. Open Explorer and Properties.

3. Create the structure described below and paste each script's content into a Script/ModuleScript/LocalScript at the matching location.

Workspace:

- Folder: Platforms (your obby parts)

- Folder: Coins (place coin Parts here; each coin Part named exactly 'Coin')

- Part: Finish

- Folder: Checkpoints (Parts named Checkpoint1, Checkpoint2, ...)

- SpawnLocation: StartSpawn

ReplicatedStorage:

- Folder: RemoteEvents

- RemoteEvent: BuyItem

ServerScriptService:

- Leaderstats (Script)

- CoinManager (Script)

- FinishHandler (Script)

- CheckpointServer (Script)

- ShopServer (Script)

- ShopItems (ModuleScript)

StarterGui:

- HUDGui (ScreenGui) -> TextLabel named CoinsLabel

- ShopGui (ScreenGui) -> Frame -> TextButton named BuyHatButton (LocalScript inside)

StarterPlayer > StarterPlayerScripts:

- (none required for this minimal build)

Notes:

- Paste the contents of each .lua file into a Script/ModuleScript/LocalScript in the Studio tree using the same filenames as the target objects above.

- For LocalScripts, place them under StarterGui elements (e.g., BuyButtonClient inside the BuyHatButton).

- Test using Play and Start Server/Start Player.

- If you want accessory-giving items, edit ShopItems.lua to Insert/clone accessories stored in ServerStorage.

"""),

"Leaderstats.lua": textwrap.dedent("""\

-- Leaderstats (ServerScriptService/Leaderstats)

local Players = game:GetService("Players")

Players.PlayerAdded:Connect(function(player)

local leaderstats = Instance.new("Folder")

leaderstats.Name = "leaderstats"

leaderstats.Parent = player

local coins = Instance.new("IntValue")

coins.Name = "Coins"

coins.Value = 0

coins.Parent = leaderstats

local time = Instance.new("IntValue")

time.Name = "Time"

time.Value = 0

time.Parent = leaderstats

-- Store checkpoint CFrame for respawn

local checkpointVal = Instance.new("CFrameValue")

checkpointVal.Name = "Checkpoint"

-- default checkpoint: StartSpawn location (set when Spawn exists and character loaded)

checkpointVal.Parent = player

end)

Players.PlayerRemoving:Connect(function(player)

-- optional: persist data or cleanup

end)

"""),

"CoinManager.lua": textwrap.dedent("""\

-- CoinManager (ServerScriptService/CoinManager)

local Players = game:GetService("Players")

local CoinsFolder = workspace:WaitForChild("Coins")

local COIN\_VALUE = 1

-- keep a weak table for debounce per coin

local touchedCoins = {}

local function onCoinTouched(coin, otherPart)

if not coin or not coin.Parent then return end

if touchedCoins[coin] then return end

local char = otherPart.Parent

if not char then return end

local player = Players:GetPlayerFromCharacter(char)

if not player then return end

local leaderstats = player:FindFirstChild("leaderstats")

if leaderstats then

local coins = leaderstats:FindFirstChild("Coins")

if coins then

touchedCoins[coin] = true

coins.Value = coins.Value + COIN\_VALUE

-- optional visual/sound

if coin:IsA("BasePart") then

coin.CanCollide = false

coin.Transparency = 1

end

-- destroy coin after tiny delay

wait(0.08)

if coin.Parent then

coin:Destroy()

end

touchedCoins[coin] = nil

end

end

end

-- Connect existing coins

for \_, coin in pairs(CoinsFolder:GetChildren()) do

if coin:IsA("BasePart") and coin.Name == "Coin" then

coin.Touched:Connect(function(hit) onCoinTouched(coin, hit) end)

end

end

-- If you add coins later during play, connect them

CoinsFolder.ChildAdded:Connect(function(child)

if child:IsA("BasePart") and child.Name == "Coin" then

child.Touched:Connect(function(hit) onCoinTouched(child, hit) end)

end

end)

"""),

"FinishHandler.lua": textwrap.dedent("""\

-- FinishHandler (ServerScriptService/FinishHandler)

local Players = game:GetService("Players")

local finish = workspace:WaitForChild("Finish")

local startSpawn = workspace:WaitForChild("StartSpawn") -- spawn location

local function giveRewardAndRespawn(player)

local stats = player:FindFirstChild("leaderstats")

if stats and stats:FindFirstChild("Coins") then

stats.Coins.Value = stats.Coins.Value + 10 -- finish reward

end

-- teleport player to start spawn

local char = player.Character

if char and char:FindFirstChild("HumanoidRootPart") and startSpawn then

char.HumanoidRootPart.CFrame = startSpawn.CFrame + Vector3.new(0,3,0)

end

end

finish.Touched:Connect(function(hit)

local player = Players:GetPlayerFromCharacter(hit.Parent)

if not player then return end

giveRewardAndRespawn(player)

end)

"""),

"CheckpointServer.lua": textwrap.dedent("""\

-- CheckpointServer (ServerScriptService/CheckpointServer)

local Players = game:GetService("Players")

local checkpointsFolder = workspace:WaitForChild("Checkpoints")

-- When player touches a checkpoint, save CFrame to player.Checkpoint

local function onCheckpointTouched(checkpointPart, hit)

local player = Players:GetPlayerFromCharacter(hit.Parent)

if not player then return end

local cfValue = player:FindFirstChild("Checkpoint")

if cfValue and checkpointPart:IsA("BasePart") then

cfValue.Value = checkpointPart.CFrame + Vector3.new(0,3,0)

end

end

-- connect existing

for \_, cp in pairs(checkpointsFolder:GetChildren()) do

if cp:IsA("BasePart") then

cp.Touched:Connect(function(hit) onCheckpointTouched(cp, hit) end)

end

end

checkpointsFolder.ChildAdded:Connect(function(child)

if child:IsA("BasePart") then

child.Touched:Connect(function(hit) onCheckpointTouched(child, hit) end)

end

end)

-- On spawn/respawn, teleport to last checkpoint if set

Players.PlayerAdded:Connect(function(player)

player.CharacterAdded:Connect(function(char)

local cfVal = player:WaitForChild("Checkpoint", 5)

if cfVal and cfVal.Value ~= CFrame.new(0,0,0) then

if char:FindFirstChild("HumanoidRootPart") then

char.HumanoidRootPart.CFrame = cfVal.Value

else

-- wait for HRP

local hrp = char:WaitForChild("HumanoidRootPart", 5)

if hrp then hrp.CFrame = cfVal.Value end

end

else

-- default: Send to StartSpawn

local startSpawn = workspace:FindFirstChild("StartSpawn")

if startSpawn and char:FindFirstChild("HumanoidRootPart") then

char.HumanoidRootPart.CFrame = startSpawn.CFrame + Vector3.new(0,3,0)

end

end

end)

end)

"""),

"ShopItems.lua": textwrap.dedent("""\

-- ShopItems (ServerScriptService/ShopItems)

local ShopItems = {}

-- format: itemName = {cost = number, apply = function(player) end}

ShopItems.Items = {

Hat1 = {

cost = 5,

apply = function(player)

-- Example effect: temporary speed boost 30s

local char = player.Character

if not char then return end

local humanoid = char:FindFirstChildOfClass("Humanoid")

if humanoid then

local originalSpeed = humanoid.WalkSpeed

humanoid.WalkSpeed = originalSpeed + 8

delay(30, function()

if humanoid and humanoid.Parent then

humanoid.WalkSpeed = originalSpeed

end

end)

end

end

},

}

return ShopItems

"""),

"ShopServer.lua": textwrap.dedent("""\

-- ShopServer (ServerScriptService/ShopServer)

local ReplicatedStorage = game:GetService("ReplicatedStorage")

local RemoteFolder = ReplicatedStorage:WaitForChild("RemoteEvents")

local BuyItem = RemoteFolder:WaitForChild("BuyItem")

local ShopItems = require(script.Parent:WaitForChild("ShopItems"))

BuyItem.OnServerEvent:Connect(function(player, itemName)

if type(itemName) ~= "string" then return end

local itemDef = ShopItems.Items[itemName]

if not itemDef then return end

local stats = player:FindFirstChild("leaderstats")

if not stats then return end

local coins = stats:FindFirstChild("Coins")

if not coins then return end

if coins.Value >= itemDef.cost then

coins.Value = coins.Value - itemDef.cost

-- apply the item effect server-side

local success, err = pcall(function()

itemDef.apply(player)

end)

if not success then

warn("Failed to apply shop item:", err)

end

else

-- Optionally: tell client insufficient funds (not implemented here)

end

end)

"""),

"BuyButtonClient.lua": textwrap.dedent("""\

-- BuyButtonClient (LocalScript inside BuyHatButton)

local ReplicatedStorage = game:GetService("ReplicatedStorage")

local RemoteFolder = ReplicatedStorage:WaitForChild("RemoteEvents")

local BuyItem = RemoteFolder:WaitForChild("BuyItem")

script.Parent.MouseButton1Click:Connect(function()

-- ask server to buy Hat1

BuyItem:FireServer("Hat1")

end)

"""),

"HUDClient.lua": textwrap.dedent("""\

-- HUDClient (StarterGui/HUDGui/HUDClient)

local Players = game:GetService("Players")

local player = Players.LocalPlayer

local coinsLabel = script.Parent:WaitForChild("CoinsLabel") -- ensure this exists

local function bindCoins()

local leaderstats = player:WaitForChild("leaderstats", 10)

if leaderstats then

local coins = leaderstats:WaitForChild("Coins", 10)

if coins then

coinsLabel.Text = "Coins: " .. tostring(coins.Value)

coins.Changed:Connect(function()

coinsLabel.Text = "Coins: " .. tostring(coins.Value)

end)

end

end

end

if player.Character then

bindCoins()

end

player.CharacterAdded:Connect(bindCoins)

"""),

}

# write files

for name, content in files.items():

path = os.path.join(BASE, name)

with open(path, "w", encoding="utf-8") as f:

f.write(content)

# create zip

zip\_path = "/mnt/data/roblox\_coinobby\_scripts.zip"

with zipfile.ZipFile(zip\_path, "w", compression=zipfile.ZIP\_DEFLATED) as zf:

for name in files:

zf.write(os.path.join(BASE, name), arcname=name)

# list created files and zip

created = {"files": list(files.keys()), "zip": zip\_path}

created