Making Small-Scale 2D Games with LÖVE and Lua

Objectives

Objectives

how to design a small-scope game
that can be created within fourteen days

Objectives

how to get started using the LÖVE 2D framework to create your 2D game

Objectives

how to avoid the three landmines of small-scope game design.

How to design a small-scope game that can be created within fourteen days

Scope

Scope

--[[how, then, do you make sure that you are not overcommitting to a game scope that exceeds your capabilities and available time?]]--

Scope

function stepOne()

--Take stock of what **game dynamic** is at the core of your desire to create a game.

end

Game Dynamic

Game Dynamic

Territorial Acquisition;
Prediction;
Spatial Reasoning;
Survival;
Destruction;
Building;
Collection;
Chasing or Evading;
Trading; and
Race to the End.

(Romero & Scheiber, 2017, pp.5-8)

Game Dynamic

Scope

```
function stepTwo()
    --Take stock of the game
    that you would like to
    create.
end
```

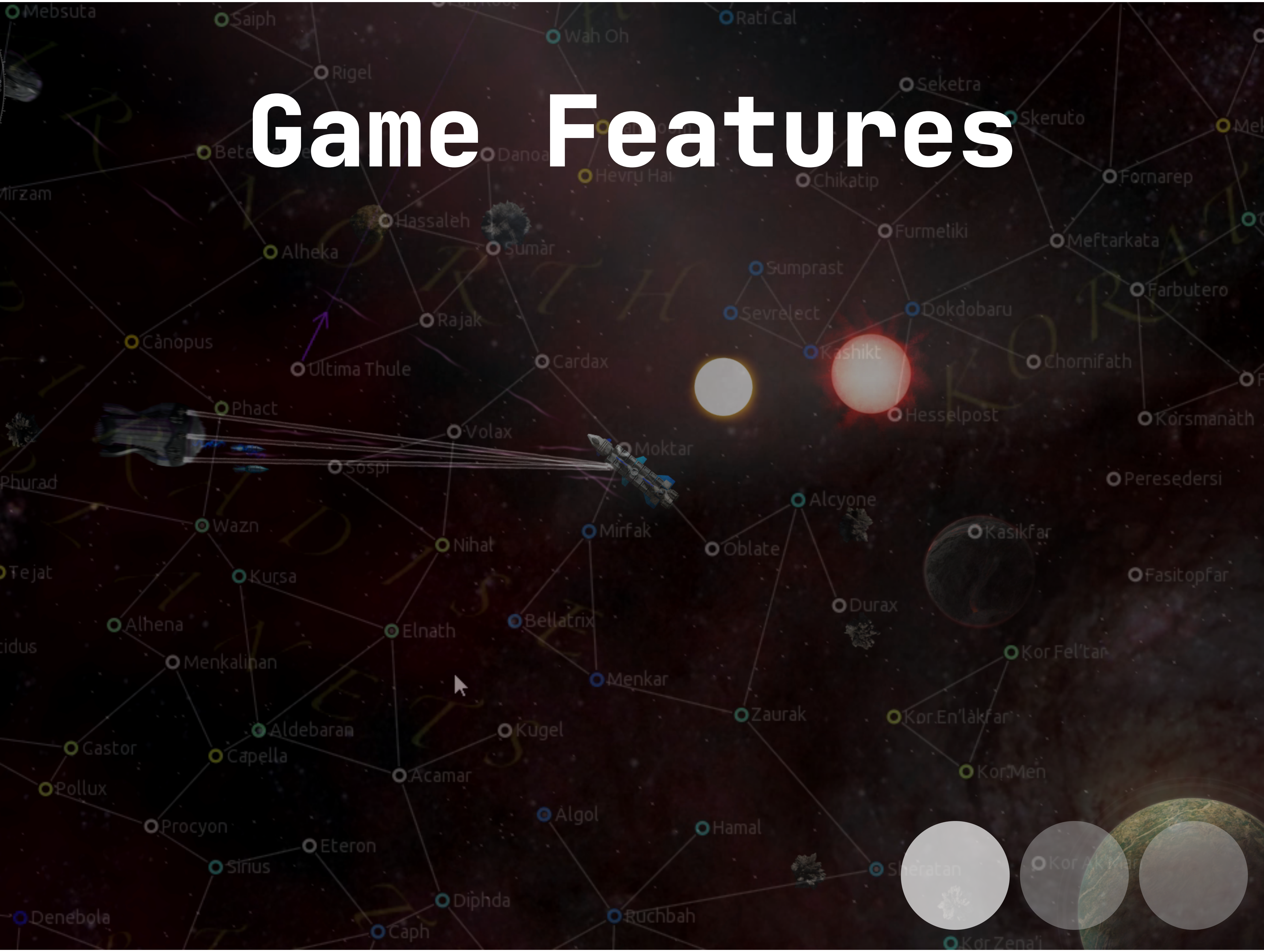
Game Features

Game Features

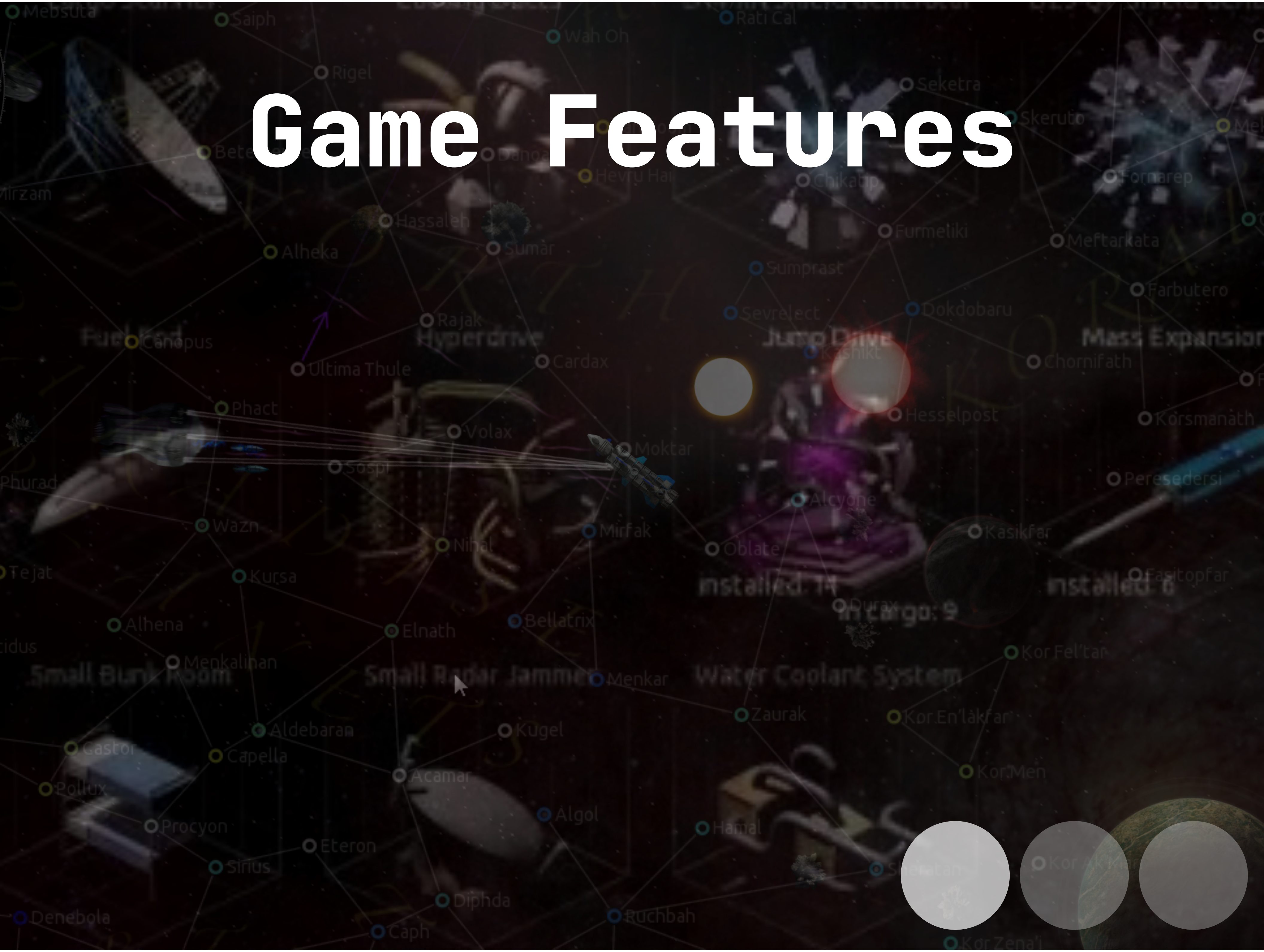
The player;
Environment;
Characters;
Artwork;
Abilities;
Levels;
Power-Ups; and
Scoring.

Game Features

Game Features



Game Features



Game Features

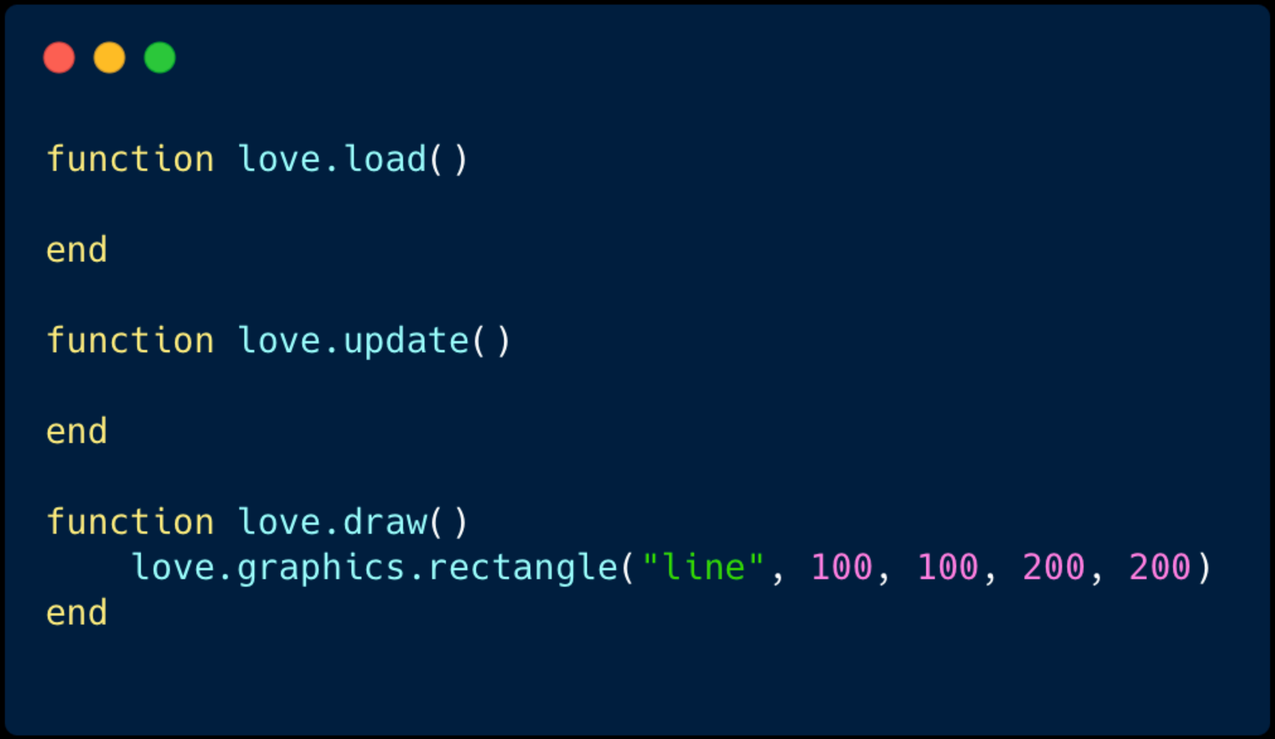
Scope

```
function stepThree()
    --Reduce the game down
    to a Minimal Viable
    Product.
end
```

How to get started using the LÖVE 2D framework to create your 2D game

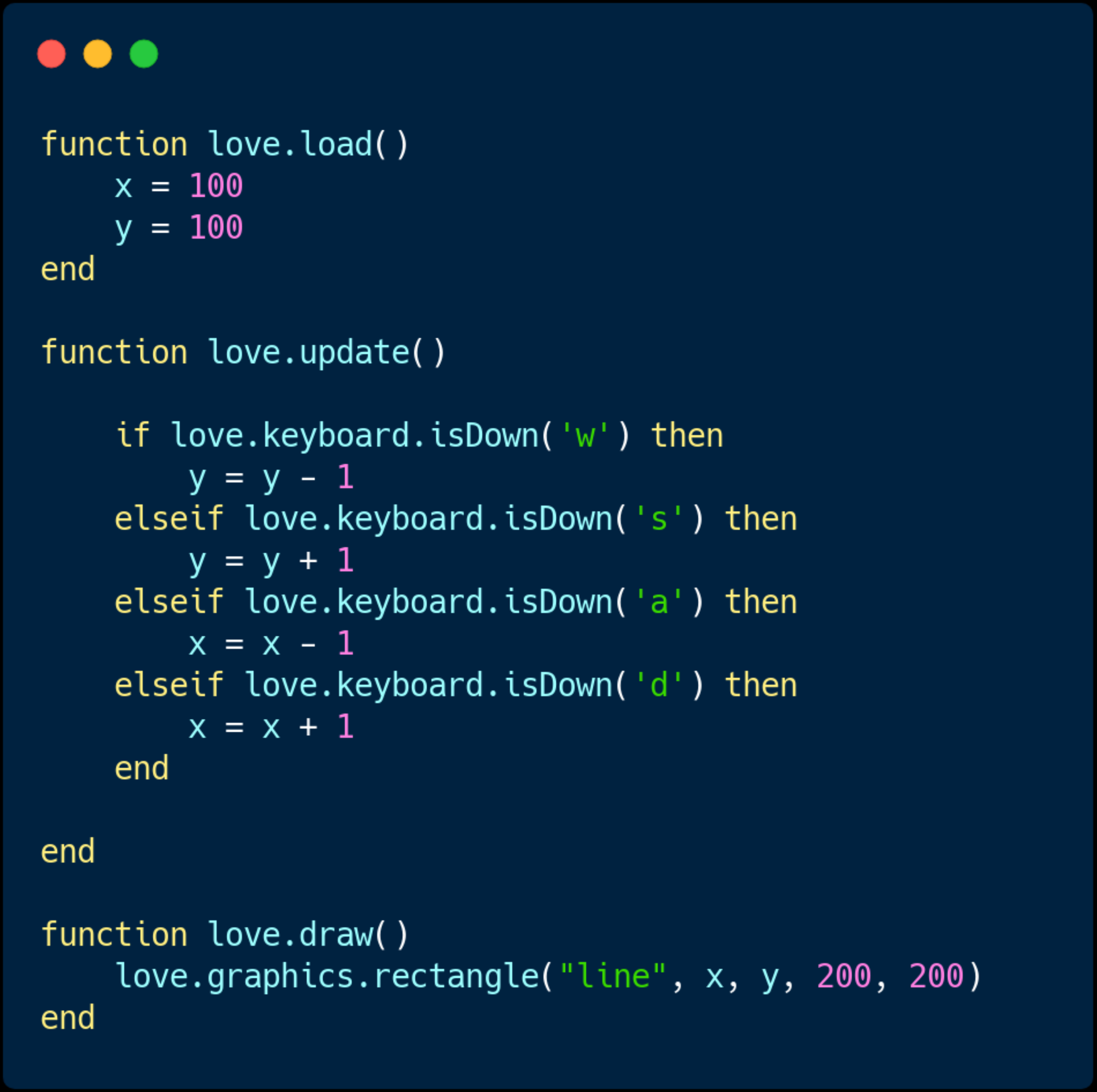
LÖVE Basics

LÖVE Basics



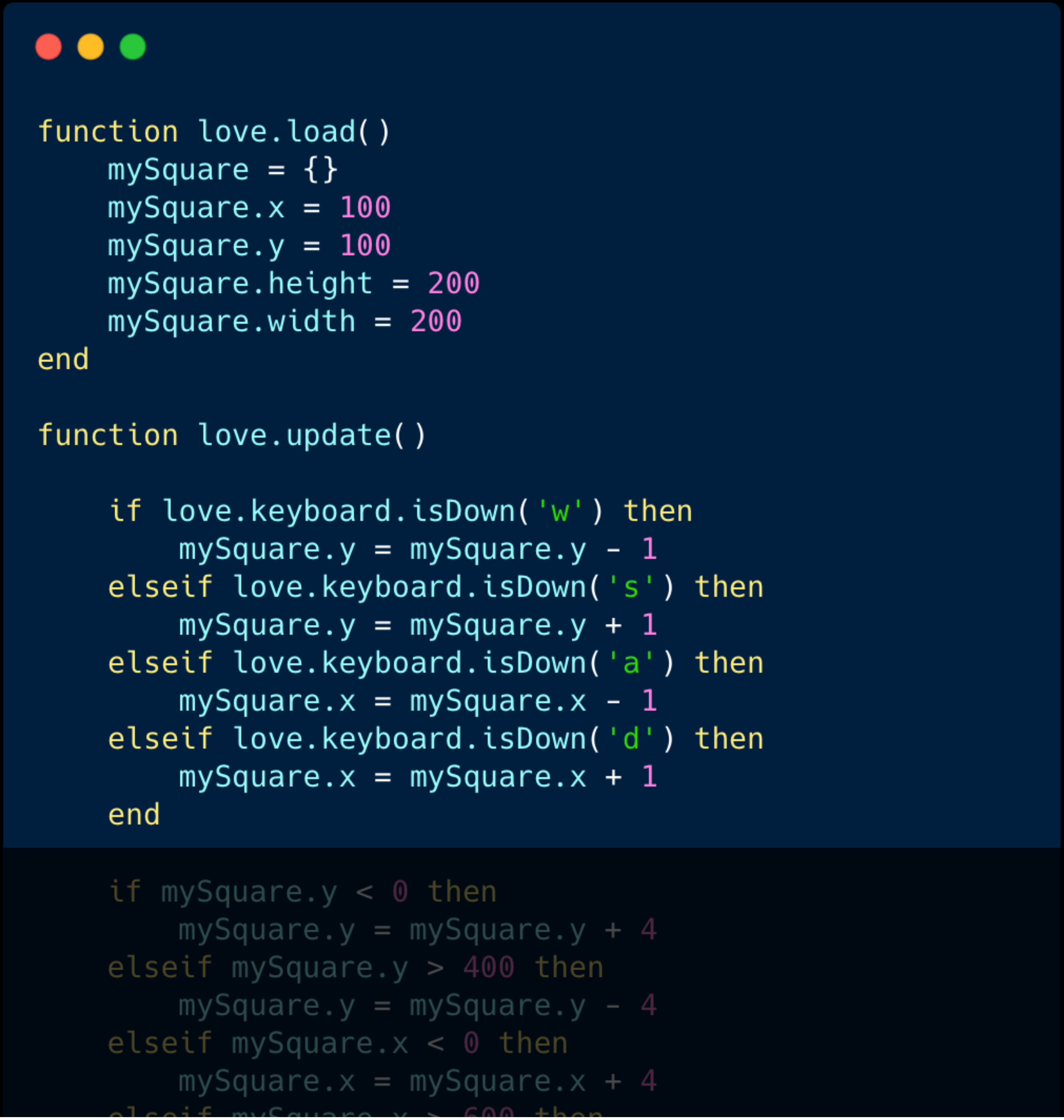
```lua
function love.load()

end

function love.update()

end

function love.draw()
    love.graphics.rectangle("line", 100, 100, 200, 200)
end
```

LÖVE Basics



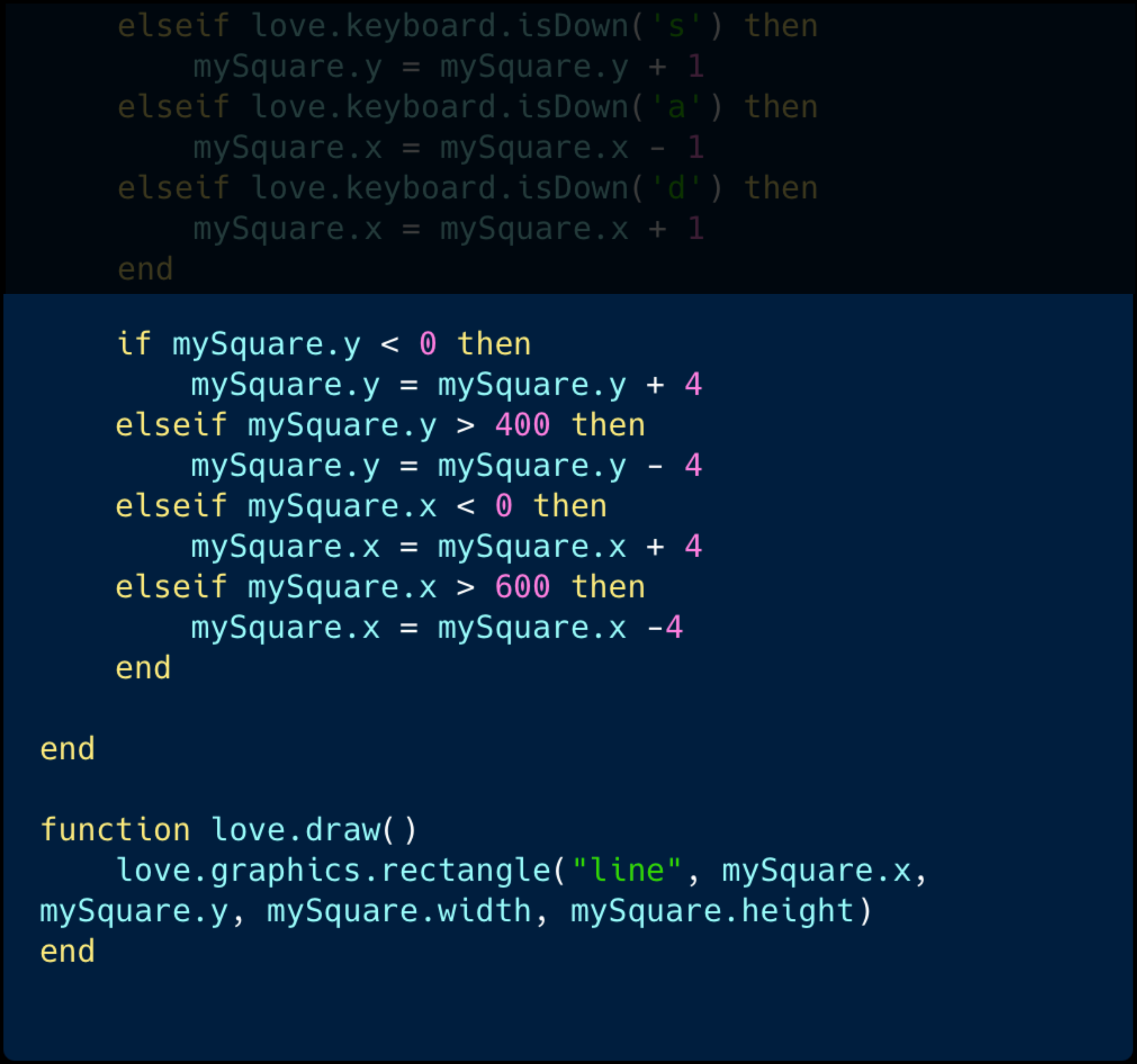
```lua
function love.load()
    x = 100
    y = 100
end

function love.update()

    if love.keyboard.isDown('w') then
        y = y - 1
    elseif love.keyboard.isDown('s') then
        y = y + 1
    elseif love.keyboard.isDown('a') then
        x = x - 1
    elseif love.keyboard.isDown('d') then
        x = x + 1
    end

end

function love.draw()
    love.graphics.rectangle("line", x, y, 200, 200)
end
```

LÖVE Basics



```lua
function love.load()
    mySquare = {}
    mySquare.x = 100
    mySquare.y = 100
    mySquare.height = 200
    mySquare.width = 200
end

function love.update()

    if love.keyboard.isDown('w') then
        mySquare.y = mySquare.y - 1
    elseif love.keyboard.isDown('s') then
        mySquare.y = mySquare.y + 1
    elseif love.keyboard.isDown('a') then
        mySquare.x = mySquare.x - 1
    elseif love.keyboard.isDown('d') then
        mySquare.x = mySquare.x + 1
    end

    if mySquare.y < 0 then
        mySquare.y = mySquare.y + 4
    elseif mySquare.y > 400 then
        mySquare.y = mySquare.y - 4
    elseif mySquare.x < 0 then
        mySquare.x = mySquare.x + 4
    elseif mySquare.x > 600 then
```

LÖVE Basics



```
    elseif love.keyboard.isDown('s') then
        mySquare.y = mySquare.y + 1
    elseif love.keyboard.isDown('a') then
        mySquare.x = mySquare.x - 1
    elseif love.keyboard.isDown('d') then
        mySquare.x = mySquare.x + 1
    end

    if mySquare.y < 0 then
        mySquare.y = mySquare.y + 4
    elseif mySquare.y > 400 then
        mySquare.y = mySquare.y - 4
    elseif mySquare.x < 0 then
        mySquare.x = mySquare.x + 4
    elseif mySquare.x > 600 then
        mySquare.x = mySquare.x -4
    end

end

function love.draw()
    love.graphics.rectangle("line", mySquare.x,
mySquare.y, mySquare.width, mySquare.height)
end
```

Phases of building a LÖVE game

Best resources to help you get started

Best resources to help you get started

Sheepolution’s “How to LÖVE”

https://sheepolution.com/learn/book/contents

Best resources to help you get started

Simple Game Tutorials

https://simplegametutorials.github.io/love/

Best resources to help you get started

Brenda Romero and Ian Schreiber's "Challenges for Game Designers."

Best resources to help you get started

Extra Credits

https://www.extracredits.site

How to avoid the three landmines of small-scope game design. For less comfortable and more comfortable programmers alike

```
function landmineOne()
  --skipping prototyping
end
```

```
function landmineTwo()
  --over-scoping
end
```

```
function landmineThree()
    --focusing on features
    over function
end
```