Love.load()

Love.update(dt)

Love.draw()

------------> expected in main.lua

Love.graphics.printf(text, x, y, [width], [align]) align left, right or center

Love.window.setMode(width, height, params)

Love.graphics.defaultFilter(min, mag)

Love.keypressed(key)

Love.event.quit()

Love.graphics.newFont(path, size)

Love.graphics.setFont(font)

Love.graphics.clear(r, g, b, a)

Love.graphics.rectangle(mode, x, y, width,height)

CONTROL

Love.keyboard.isDown(key) -> return true or false if key is held down != love.keypressed(key) only fire code once per press

RANDOM

Monte-carlo algorithm

Math.randomseed(num)

Os.time()

Math.random(min, max) – math.random(upper) (int 1 to upper) – math.random() (float 0 to 1)

CLAMP

Math.min(num1, num2)

Math.max(num1, num2)

Love.timer.getFPS()

Love.window.setTitle(title)

AABB collision detection

1/ alix-aligned bounding boxes (no rotation in world space)

If rect1.x is not > rect2.x +rect2.width and

rect1.x + rect1.width is not < rect2.x and

rect1.y is not > rect2.y + rect2.height and

rect1.y + rect1.height is not < rect2.y:

collision is true

else

collision is false

state machine

start

--press Enter--> serve

--press Enter--> play

--first who get 10 score- > over

love.audio.newSource(path, [type])

type: stream(stream from disk when needed) or static (preserved in memory)

large sound effect/ soundtrack -> stream is more effective

love.resize(width, height)