

Create amazing iOS games with Corona,  
Gideros, Moai, and more



Learn  
**Lua for iOS Game  
Development**

Jayant Varma

Apress®

# Learn Lua for iOS Game Development



Jayant Varma

Apress®

# Learn Lua for iOS Game Development

---

Copyright © 2013 by Jayant Varma

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN-13 (pbk): 978-1-4302-4662-6

ISBN-13 (electronic): 978-1-4302-4663-3

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

President and Publisher: Paul Manning

Lead Editor: Michelle Lowman

Developmental Editor: Richard Carey

Technical Reviewers: Mekha Shrestha and Yujan Shrestha

Editorial Board: Steve Anglin, Mark Beckner, Ewan Buckingham, Gary Cornell, Morgan Ertel, Jonathan Gennick, Jonathan Hassell, Robert Hutchinson, Michelle Lowman, James Markham, Matthew Moodie, Jeff Olson, Jeffrey Pepper, Douglas Pundick, Ben Renow-Clarke, Dominic Shakeshaft, Gwenan Spearing, Matt Wade, Tom Welsh

Coordinating Editor: Brigid Duffy

Copy Editor: Damon Larson

Compositor: SPi Global

Indexer: SPi Global

Artist: SPi Global

Cover Designer: Anna Ishchenko

Distributed to the book trade worldwide by Springer Science+Business Media, LLC., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail [orders-ny@springer-sbm.com](mailto:orders-ny@springer-sbm.com), or visit [www.springeronline.com](http://www.springeronline.com).

For information on translations, please e-mail [rights@apress.com](mailto:rights@apress.com), or visit [www.apress.com](http://www.apress.com).

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales—eBook Licensing web page at [www.apress.com/info/bulksales](http://www.apress.com/info/bulksales).

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is available to readers at [www.apress.com](http://www.apress.com).

---

*To my loving parents and my son, Mihir*

---

# Contents at a Glance

---

[About the Author](#)

[About the Technical Reviewers](#)

[Acknowledgments](#)

- [Chapter 1: Introduction to Lua](#)
- [Chapter 2: Lua Libraries](#)
- [Chapter 3: File Operations](#)
- [Chapter 4: Math with Lua](#)
- [Chapter 5: Strings](#)
- [Chapter 6: Threading](#)
- [Chapter 7: Tips and Tricks](#)
- [Chapter 8: Corona SDK](#)
- [Chapter 9: Gideros Studio](#)
- [Chapter 10: Moai](#)
- [Chapter 11: LOVE](#)
- [Chapter 12: Codea](#)
- [Chapter 13: Libraries](#)
- [Chapter 14: Third-Party Applications](#)
- [Chapter 15: Sample Source Code](#)

[Index](#)

# Contents

---

**About the Author**

**About the Technical Reviewer**

**Acknowledgments**

## ■ **Chapter 1: Introduction to Lua1**

What Is Lua?

The History of Lua

Lua Timeline

Starting with Lua

Setting Up Lua

Online Lua Shell

Windows, Mac OS X, and \*nix  
iOS

Features of Lua

Variables

Hello World, the Variable Way

Strings

Numerals

Values and Types

Code Blocks and Scopes

Lua Operators

Arithmetic Operators

Relational Operators

Logical Operators

Other Operators

Summary

## ■ **Chapter 2: Lua Libraries**

Basic Functions

`assert ( v [, message] )`

`collectgarbage ( [opt [,arg]] )`

`dofile ( [filename] )`

`error ( message [,level] )`

[\\_G](#)

---

[getfenv \( \[f\] \)](#)

[getmetatable \( object \)](#)

[ipairs \( t \)](#)

[load \( func \[,chunkname\] \)](#)

[loadstring \( string \[,chunkname\] \)](#)

[next \( table \[,index\] \)](#)

[pairs \( t \)](#)

[pcall \( f, arg1, ... \)](#)

[print\( ... \)](#)

[rawequal \( v1, v2 \)](#)

[rawget \( table, index \)](#)

[rawest \( table, index, value \)](#)

[select \( index, ... \)](#)

[setfenv \( f, table \)](#)

[setmetatable \( table, metatable \)](#)

[tonumber\( e \[,base\] \)](#)

[tostring \( e \)](#)

[type \( v \)](#)

[unpack \( list \[, i \[, j\] \] \)](#)

[\\_VERSION](#)

[xpcall \( f, err \)](#)

## System Libraries

[Table Functions](#)

[OS Functions](#)

## Tables: A Quick Overview

[Tables As Arrays](#)

[Tables As Associative Arrays](#)

## Functions: An Advanced Look

[Tables As Objects](#)

[The Difference Between . and :](#)

## Summary

## ■ Chapter 3: File Operations

### Implicit Functions

[io.close \( \[file\] \)](#)

[io.flush \( \)](#)



io.input ( [file] )

---

io.lines ( [filename] )

io.open ( filename [,mode] )

io.output ( [file] )

io.read ( ... )

io.tmpfile ( )

io.type ( obj )

io.write ( ... )

## Explicit Functions

file:close ( )

file:flush ( )

file:lines ( )

file:read ( [format] )

file:seek ( [whence] [, offset] )

file:setvbuf (mode [, size] )

file:write( ... )

## Uses of File I/O in Gaming

### Saving a Variable

### Grabbing the Data

How the Code Works

Potential Issues

### Saving Data to a Variable

How the Code Works

Potential Issues

### Writing Data to the File

### Saving a Table

### Dynamic Variables

### Summary

## ■ Chapter 4: Math with Lua

### Introduction to Math in Lua

math.abs (x)

math.acos (x)

math.asin (x)

math.atan (x)

`math.atan2 (y,x)`

---

`math.ceil (x)`

`math.cos (x)`

`math.cosh (x)`

`math.deg (x)`

`math.exp (x)`

`math.floor (x)`

`math.fmod (x)`

`math.frexp (x)`

`math.huge`

`math.ldexp (m, e)`

`math.log (x)`

`math.log10 (x)`

`math.max (x, ...)`

`math.min (x, ...)`

`math.modf (x)`

`math.pi`

`math.pow (x,y)`

`math.rad (x)`

`math.random ( [m [,n]] )`

`math.randomseed (x)`

`math.sin (x)`

`math.sinh (x)`

`math.sqrt (x)`

`math.tan (x)`

`math.tanh (x)`

## Practical Uses of Math in Gaming

Assignment

Increasing and Decreasing

Introducing the Point

Conditional Branching

Flipping a Coin

Throwing a Die

Using Flags

Swimming Between the Flags: Multiple Flags

Using Math to Loop

Using a Grid in a Game

## Boolean Math

## Summary

## ■ Chapter 5: Strings

### What Is a String?

- `string.byte ( s [i [j] ] )`
- `string.char ( . . . )`
- `string.dump ( function )`
- `string.find ( s, pattern [,init [,plain] ] )`
- `string.format ( formatString, . . . )`
- `string.gmatch ( s, pattern )`
- `string.gsub ( s, pattern, repl [,n] )`
- `string.len ( s )`
- `string.lower ( s )`
- `string.match ( s, patterns [,init] )`
- `string.rep ( s, n )`
- `string.reverse ( s )`
- `string.sub ( s, i [,j] ] )`
- `string.upper ( s )`

### Patterns

- Character Class
- Pattern Item
- Pattern
- Capture

### Using the String Functions

- Converting Strings to Uppercase
- Converting Strings to Lowercase
- Converting Strings to Title Case
- Padding a String
- CSV Functionality
- Formatting a Number with the Thousand Separator
- Frequency of Letters
- Detecting Whether a String Is a Palindrome
- Splitting a String

### Level Management

## Summary

---

### ■ Chapter 6: Threading

#### Coroutines

- Coroutine Functions

- Creating a New Coroutine

#### Working with Tables

- Customizing the Metatable

- A Useful Example

#### Object-Oriented Lua

#### But What's the Object?

#### Summary

### ■ Chapter 7: Tips and Tricks

#### Generic Lua Functions

- Using the printf Function

- Counting the Number of Elements in a Table

- Using isEmpty

- Knowing the Index of an Element

- Determining Whether a Table Is an Array

- Setting Default Values

- Copying Tables

- Performing a Deep Copy

- Copying Array Components

- Copying Non-Array Components

- Merging Two Tables

- Determining Whether a Value Exists in a Table

- Finding the Difference Between Two Tables

- Getting a Return Value As a Table

- Sorting on Elements of a Table

- Determining Frequency of an Item in a Table

- Converting Numbers into Roman Numerals

- Creating Linked Lists

- Expanding Tokenized Variables

- Padding Zeros to a String

- Getting the Month As a String

- Accessing Strings Like an Array

Finding the Distance Between Two Points in One or Two Dimensions

---

Determining the Angle Between Two Points

Keeping a Number in a Given Range

Performing Linear Interpolation

Getting the Sign of a Value

## Collisions

Using isPointInRect

Using pointInCircle

Determining Whether a Rectangle Is Within Another Rectangle

Determining Whether a Circle Is Within Another Circle

Identifying Overlapping Rectangles

Identifying Overlapping Circles

Determining Whether a Circle Overlaps a Rectangle

Using pointInTriangle

Using pointInPolygon

## Other Generic Functions

Comparing Boolean Values

Translating C/Java Like Loops to Lua

Applying Friction to an Object

Simulating a Jack-in-the-Box

Using a Sine Scroller

Placing a Tile on a Board with No Other Tile in the Same Row and Column

Using Arrays to Output Large Amounts of Text Using Templates

## Parameter Handling

Fixed Parameters

Variable Parameters

Variable Named Parameters

Using Varargs

Parsing a List of Passed Parameters

Making Read-Only Tables

Implementing a Stack

Converting Between Parameters and Tables

## Vector 2D

## Summary

# ■ Chapter 8: Corona SDK

## Setting Up Corona SDK

---

### How Corona SDK Works

### Hello World for Corona SDK

- Graphic Version

- Hello World on the Screen

- Hello World on the Device

### Beyond Hello World: Creating a Rectangle on the Screen

- Groups

- Images

- Event-Based Processing

- Input: Touches

- Physics

### Timing Is Everything

- Frames

- Making a Health Bar

### Animating Using enterFrame

- Back to Health Bars

- Using Transition Functions

- Removing Objects from the Screen

### Creating Sound

- Let's Make Some Noise

- Manipulating the Sound

### Working with Videos

### Creating an Elevator

- Zooming Out for an Overview

- More Events

- Using Maps

### The Internet Browser

### Nothing Lasts Forever

### Enterprise Edition

### Summary

## ■ Chapter 9: Gideros Studio

- License

# Installing Gideros Studio

---

What's Included

Setting the License

First Steps

Running the Code

Configuring the Project

## Architecture

## The Gideros Engine

Hello Birdie

Aligning the Images

Groups

Displaying Text

Drawing Shapes

The Application Object

Adding Sound Effects

Events

Animations

Networking and the Internet

GPS and Compass

Accelerometer

Gyroscope

Physics

## Plug-Ins

## Summary

## ■ Chapter 10: Moai

What Is Moai?

Obtaining Moai

The Moai SDK

Hello Moai

Displaying Text

TrueType Fonts

Bitmap Fonts

Text Attributes

Drawing Vector Primitives

Drawing Lines

---

Drawing Rectangles

Drawing Filled Rectangles

Drawing Circles

Drawing Filled Circles

Drawing Ellipses

Drawing Filled Ellipses

Drawing Polygons

Drawing Points

The Drawing Attributes

Drawing Images

Drawing Custom Images

Loading Images

Copying Images

Saving Images

Resizing Images

Pixel Access for Images

Animation

Tile Decks

Threading

Groups

Handling Input

Keyboard Events

Mouse Events

Touch Events

Sound

Displaying a Dialog

Displaying Video

Device Orientation

Notifications

Networking

Using JSON

Using Base64

Compressing Data



## Physics

---

Box2D Physics

Chipmunk Physics

## Moai Cloud

Creating a Web Service

Consuming the Web Service

## Summary

## ■ Chapter 11: LÖVE

The Architecture

Installing LÖVE

Running LÖVE

The Callback Functions

`love.load ( )`

`love.update ( dt )`

`love.draw ( )`

`love.mousepressed ( x, y, button )`

`love.mousereleased ( x, y, button )`

`love.keypressed ( key, unicode )`

`love.keyreleased ( key )`

`love.focus ( )`

`love.quit ( )`

LÖVE Namespaces

`love.audio`

`love.event`

`love.filesystem`

`love.font`

`love.graphics`

`love.image`

`love.joystick`

`love.mouse`

`love.physics`

`love.sound`

`love.thread`

`love.timer`

The Graphics Module

Images

---

Moving Around

The Active Window

Moving Around with the Keyboard

Turning the Player

Drawing Primitives

Application Settings – conf.lua

Creating Effects

Physics

Displaying Text

Shaders

Making Sound

Sample Game Code

Summary

## ■ Chapter 12: Codea

Getting Codea

Codea Architecture

Hello World

Parameters

Drawing with Codea

Orientation

System Keyboard

Drawing Mode

Background Color

Pen Color

Fill Color

Line Width

Drawing Lines

Anti-Aliasing

Drawing Circles and Ellipses

Drawing Rectangles

Drawing Text

Displaying Images

Drawing Off-screen

---

Loading Sprites into an Image

Saving Images

Saving Data

Drawing Without Color

Transformation Functions

Recording Video

Touch

Drawing with Touch

Multitouch

Making Sound

iOS Hardware

Downloading Data

Displaying Web Pages

Physics

Vector Math

angleBetween

cross

dist

distSqr

dot

len

lenSqr

normalize

rotate

rotate90

Summary

## ■ Chapter 13: Libraries

Particle Candy

Sample Code

Text Candy

Sample Code

Widget Candy

Sample Code

## Corona AutoLAN

---

Sample Code

## Corona Profiler

Sample Code

## Director

Sample Code

## Lime

Sample Code

## RapaNui

Sample Code

## Gideros Illustrator (SVG Library)

Sample Code

## TNT Particles Library

Sample Code

## Busted

Sample Code

## Moses

Sample Code

## Allen

Sample Code

## BhWax

Sample Code

## Summary

## ■ Chapter 14: Third-Party Applications

### IDEs

Glider

CoronaComplete

Lua Studio

ZeroBrane Studio

### Text Editors

Notepad++

TextMate

TextWrangler

BBEdit

Sublime Text 2

---

Textastic Code Editor

## Snippets Collectors

CodeBox

Code Collector Pro

## Versioning

## Bitmap Editors

Pixen

GraphicsGale

Pixelmator

Axialis IconWorkshop

Inkscape

Paint.NET

iConify

## Sprite Animation and Level Creation Tools

SpriteHelper

LevelHelper

Tiled

TexturePacker

PhysicsEditor

Particle Designer

Glyph Designer

Spriteloq

Zoë

TNT Animator Studio

## Sound Tools

Audacity

Bfxr

NodeBeat

## Summary

## ■ Chapter 15: Sample Source Code

Graphics

Setting Up the Environment

Making the Chopper Fly

Using the Accelerometer

---

Moving the Helicopter

Making Sound

Shooting Bullets

Moving the Bullets

Spawning Enemies

Moving the Scenery Objects

Losing a Life

Let There Be Some Color

Displaying Information

Game Over; Play Again?

Collisions

- Shooting Planes and Balloons

- Saving Survivors and Refueling

Putting It All Together

Summary

**Index**

# About the Author

---



**Jayant Varma** is the executive director of OZ Apps, an Australian-based consultancy focused on assisting local and international businesses in mobile product enhancement strategies and specialized services for mobile development and training. He has several apps on the Mac App Store to his credit.

Jayant has an MBA-MIT and is an IT veteran with over 20 years of rich and varied experience in several countries, in sectors including automotive and higher education. Following on with his passion for teaching, he has lectured and developed subject outlines at the James Cook University.

His mobile development journey started in early 2000 with the Microsoft Pocket PC (Compaq iPaq) and has continued to Apple's iOS today. His fascination with technology and games goes back to when he started writing games on his first computer, the ZX Spectrum.

Jayant has been an active member of ACS (Australian Computer Society) and is part of the executive board for the local chapter. He is a founding chairman of the first local YIT (Young IT) chapter. He has conducted iOS development workshops for the AUC (Apple University Consortium) and is enthusiastic with mentoring and advising young developers. He also helps other developers via his blogs on topics related to development.

You can follow Jayant on Twitter at @ozapps, on the blogs at <http://howto.oz-apps.com> and <http://reviewme.oz-apps.com>, and the Lua blogs <http://dev.oz-apps.com> and @LearnLua.

# About the Technical Reviewers

---

**Mekha Shrestha** and **Yujan Shrestha** are founders of the software development company M.Y. Developers, which develops advanced tools for software and game development.

Yujan Shrestha completed his education in biomedical engineering at the University of Texas in Austin, and Mekha Shrestha studied neuroscience at Baylor University. After working on various projects in college, they decided to work together on some innovative projects for research on autism disorders conducted at the University of Texas Medical School at Houston. Following their research, they worked on making simple strategic games. Their first game project was Space Conquest, which is an adventure/action game with realistic physics that enables the player to experience strategic game play. During its development, they recognized the need for tools that improved the quality of game development for indie and professional developers alike, which lead to the making and designing of various tools for Corona SDK—namely, Corona Profiler and Corona Ultimote. They also developed an acclaimed Lua-based IDE called Lua Glider for various platforms, including Corona SDK, Moai SDK, Gideros SDK, and LÖVE 2D.

As a team they aim to develop and improve tools that will enhance the experience of software and game development.



**Patrick Meehan** is the Founder and CEO of Zipline Games, Inc and the architect of Moai SDK.

<http://www.linkedin.com/in/patrickmeehan>

**Atilim Çetin:** Studied at METU Computer Engineering department during 1996–2001. He worked as a software specialist at Makrosam Software, TÜBİTAK UEKAE and Meteksan System companies. During 2006–2009 he worked as a specialist software engineer, technical group leader and three dimensional software group leader in various simulation projects at KaTron Defense, Space and Simulation Technologies company in Koç Group. In 2010 he founded Gideros Mobile and is the design architect and technical leader of the Gideros Studio. He is the author of the book entitled Computer Graphics (ISBN: 975 347 104 0) and also one of the authors of “Three Dimensional Computer Graphics” and “Animation” titles in the Turkish



- [download Los Angeles's Historic Ballparks \(Images of America\)](#)
- [download Italian Family Cooking: Unlocking A Treasury Of Recipes and Stories](#)
- [The Bahir pdf, azw \(kindle\)](#)
- [read online Writing Machines](#)
- [The 8-Hour Diet: Watch the Pounds Disappear Without Watching What You Eat! pdf, azw \(kindle\)](#)
- [click Writing and Being \(Charles Eliot Norton Lectures\)](#)
  
- <http://www.mmastyles.com/books/Well-Fed--Flat-Broke--Recipes-for-Modest-Budgets-and-Messy-Kitchens.pdf>
- <http://www.mmastyles.com/books/Italian-Family-Cooking--Unlocking-A-Treasury-Of-Recipes-and-Stories.pdf>
- <http://www.shreesaiexport.com/library/Out-of-My-Mind.pdf>
- <http://www.freightunlocked.co.uk/lib/Writing-Machines.pdf>
- <http://toko-gumilar.com/books/Shengmai-San--Traditional-Herbal-Medicines-for-Modern-Times-.pdf>
- <http://thewun.org/?library/Slumming--Sexual-and-Social-Politics-in-Victorian-London.pdf>