THE LUA PROGRAMMING LANGUAGE BEGINNER'S GUIDE



In this article, we'll cover the basics of the Lua programming language, including:

- What it is
- Main components
- Key features
- Pros & cons
- Use cases
- Lua vs other languages
- Installing Lua in a Windows environment
- Getting it up and running

Let's get started.

What is Lua?



Lua is a robust, lightweight, and embeddable scripting language that supports multiple programming methods, including procedural, objectoriented, functional, and data-driven programming.

As the primary focus on Lua is for scripting, it is rarely used as a standalone programming language.

Instead, it is used as a scripting language that can be integrated (embedded) into other programs written in mainly C and C++. It also supports other programming languages via third-party plugins (<u>NLua</u>/<u>KeraLua</u> for .NET/C#).

Popular use cases for Lua include:

- As a popular component in video game and game engine development. For example, Warframe, World of Warcraft, and CRYENGINE all use Lua.
- As a programming language in many network programs, like CISCO Systems, Nmap, and ModSecurity.
- As a programming language in industrial programs such as Adobe Lightroom and MySQL Workbench.
- As a library that <u>developers</u> can integrate with their programs to enable scripting functionality.

Being a scripting language, Lua does not have its own main application. Instead, it acts exclusively as an embedded part of the host application.

(Explore the most popular programming languages.)

How Lua works

There are two main components of Lua:

- The Lua interpreter
- The Lua virtual machine (VM)

Lua is not directly interpreted through a Lua file like other languages such as <u>Python</u>. Instead, it uses the Lua interpreter to compile a Lua file to bytecode. The Lua interpreter is written in ANSI C, making it highly portable and capable of running on a multitude of devices.

Usually, the compilation is done at the runtime. However, sometimes it can be done before the runtime to increase load times. Then the Lua virtual machine will run this compiled bytecode. The register-based architecture of the Lua virtual machine closely resembles actual hardware architectures, and it will increase the overall performance of the program.

Key features of Lua

So, what's so great about Lua? These are the defining features:



Speed

Lua is considered one of the fastest programming languages among interpreted scripting languages. In particular, Lua can perform large task orders faster than most other programming languages in both benchmarking and real-world scenarios.

For more speed, an independent implementation of Lua called <u>LuaJIT</u> uses a just-in-time compiler that makes Lua even faster.

Size

Lua has a considerably smaller footprint than other programming languages, with its complete source code and documentation taking a mere 1.3 MB. The Lua interpreter with all the standard libraries takes 278K, while the complete Lua library takes only 466K.

This small size is ideal when integrating Lua into multiple platforms, from embedded devices to massive game engines, where every byte is valuable.

Portability & embeddability

With its small size, the portability of Lua is nearly unlimited; any platform that supports the standard C compiler can run Lua out of the box. Lua's speed and size become huge advantages when embedding Lua with another programming language. That's because they can help increase the

speed of the program without hindering any existing functionality.

Importantly, Lua does not require complex rewrites to be compatible with other programming languages. Lua can be used with primary programming languages like C, C++, Java, C#, etc.. and other scripting languages like Perl and Ruby, further extending its usability.

Simplicity

Lua is simple in design yet provides powerful functionality. One of the core features of Lua is metamechanisms which enable developers to implement features—rather than providing a bunch of features directly in the language itself.

Lua also comes with incremental garbage collection, reducing memory usage and implementation complexity. Its sandboxing feature can be used to isolate functions and resources. This increases the security of the program and provides coroutines for multitasking.

All these features come with a simple syntax and easily understandable format so that anyone can easily pick up Lua and use it in their programs.

License

Lua is free and open-source software distributed under the MIT license. This means anyone can use Lua for any purpose without paying any licensing or royalty fees.

Advantages & drawbacks

Like any language, Lua has its pros and cons.

Advantages of Lua

- **Easy app integration.** Its high performance and small size make it easy to integrate Lua into applications.
- **Simple syntax.** Relatively simple syntax structure with around 20 dedicated keywords, which helps to dive into Lua programming easily.
- Flexibility. Without standard libraries, you can customize Lua to meet any need.
- **Cross-platform compatibility and support** for the standard C compiler allows Lua to run virtually anywhere.
- **Dynamic variables** in Lua allow defining variables without defining types, and the type is determined automatically at the runtime.
- Easy debugging. Simple and powerful debug library.
- **Plenty of documentation.** Comprehensive documentation to get Lua projects up and running quickly and the active community.

Disadvantages of Lua

- Limited error handling support can lead to longer debug times to identify the exact errors in a Lua script.
- All variables are created as global variables (global scope), which can lead to errors in variable assignments.

• Limited pattern matching support.

When to use Lua

As a scripting language without major limitations, you can use Lua for any scenario, from a simple backend script in a web server to complex game development.

Lua is highly prevalent in video game development as it can be used to create functionality without contaminating the overall performance while also keeping everything separate.

Another area that Lua excels is embedded programming, where size and performance are major concerns. Lua can be used in everyday applications to extend the existing functionality or create new features and functions.

Some popular games, programs, and services that use Lua are Dark Souls, Fable II, Garry's Mod, Wireshark, VLC, Apache, and Nginx Web Servers.

Lua vs other languages

How does Lua stack up against other languages?

Here's a look at the differences between the high-level general programming language Python, the high-level object-oriented Java, and web-focused Javascript—all compared to Lua.

bmc

Comparing Common Programming Languages

	Lua	Python	Java	JavaScript	
Language Type	Scripting language	General purpose programming language		Web-focused programming language (Supports client- & server-side)	
Usage	Embedded scriptingMultipleMultipleEmbedded scripting(From web development to data analytics)(Desktop/mobile apps to Enterprise apps & services)		Web app development (Frontend & backend)		
Compiled vs Interpreted	Compiled	Interpreted Compiled		Interpreted	
Package Management	kage nagement LuaDist		Maven & Gradle	NPM (Node Package Manager)	
Programing Styles	ing Procedural • Object-oriented • Functional • Data-driven Procedural		 Object-oriented 	 Procedural Object-oriented 	
Ease of use (Syntax)	e of use Simple Simple Co		Complex	Simple	
Speed	Fastest	Relatively slow	Fast	Faster	
Platform Support	Multi-platform	Multi-platform	platform Multi-platform (Web		

Installing Lua

Now, let's see how to set up a development environment in Windows. First, we'll install Lua.

Step 1

Navigate to the <u>Lua.org download page</u>. Here, we will be using a precompiled binary to install Lua in windows. So, click on "get a binary link" as shown in the below screenshot.



Click "Download" on the LuaBinaries page, and you will be redirected to a page with a list of precompiled binaries. Select the appropriate version from that list.

We will be using the latest Lua version for Windows 64 bit.

) LUA 6	inaries Do									\times
ч	N C		Not Sec	ura luchicacies sourceforce act/doumle		0		al bow	- D	niata
		w	• () Horsee				-	<u> </u>		
				1						
				LuaBina	ries					
				Pre-compiled Lua librarie	es and executables.					
Lual	Binarie	S								
Hon	1e Verview		Dowr	nload						
Mo	otivation	n	All th	ne binaries, source code and doc	cumentation are available from the	Source	eFo	rge proj	ect	
lns Llia	stallatio	n	mes	page.						
Cr	edits			https://sourceforge.net/proje	ects/luabinaries/files/					
Co	ntact u	s	But I	here are shortcuts for the most p	opular downloads:					
Con	iuai figurati	on	Lual	Binaries 5.4.2 - Release 1						
Pac	kaging			lua-5.4.2_Sources.tar.gz	Source Code and Makefiles					
Lice	nload			lua-5.4.2_Sources.zip	Source Code and Makefiles					
				lua-5.4.2_Win32_bin.zip	Windows x86 Executables					
				lua-5.4.2_Win32_dllw6_lib.zip	Windows x86 DLL and Includes (MingW-w64 6 Built)					
				lua-5.4.2_Win64_bin.zip	Windows x64 Executables					
				lua-5.4.2_Win64_dllw6_lib.zip	Windows x64 DLL and Includes (MingW-w64 6 Built)					
				lua-5.4.2_MacOS1011_bin.tar.gz	MacOS X Intel Executables					
				lua-5.4.2_MacOS1011_lib.tar.gz	MacOS X Intel Library and Includes					
				lua-5.4.2_Linux54_64_bin.tar.gz	Linux x64 Executables					
				lua-5.4.2_Linux54_64_lib.tar.gz	Linux x64 Library and Includes					
			Lual	Binaries 5.3.6 - Release 1						
				lua-5.3.6_Sources.tar.gz	Source Code and Makefiles					
				lua-5.3.6_Sources.zip	Source Code and Makefiles					
				lua-5.3.6_Win32_bin.zip	Windows x86 Executables					
				lua-5.3.6_Win32_dllw6_lib.zip	Windows x86 DLL and Includes (MingW 6 Built)					
				lua-5.3.6_Win64_bin.zip	Windows x64 Executables					
				lua-5.3.6_Win64_dllw6_lib.zip	Windows x64 DLL and Includes					-
	Lual Hon Ov Man Con Paci Dow Lice	LuaBinarie Home Overview Motivation Installatio History Credits Contact u Manual Configurati Packaging Download License	LuaBinaries Home Overview Motivation Installation History Credits Contact us Manual Configuration Packaging Download License	LuaBinaries Home Down Overview All the files Installation But He files History Credits Contact us But He files Manual Lual Configuration Packaging Download License	LuaBinaries Home Overview Motivation Installation History Credits Contact us Manual Configuration Packaging Download License Download Alt he binaries, source code and door list page: LuaBinaries 5.4.2 - Release 1 LuaBinaries 5.4.2 - Release 1 Lua-5.4.2 _ Sources.tar.gz Lua-5.4.2 _ Win32_bin.zip Lua-5.4.2 _ Win34_bin.zip Lua-5.4.2 _ Win34_bin.zip Lua-5.4.2 _ Win34_dilw6_ib.zip Lua-5.4.2 _ MacOS1011_bin.tar.gz Lua-5.4.2 _ MacOS1011_bin.tar.gz Lua-5.4.2 _ MacOS1011_bin.tar.gz Lua-5.4.2 _ MacOS1011_bin.tar.gz Lua-5.4.2 _ Min34_dilw6_ib.zip Lua-5.4.2 _ Min32_bin.zip Lua-5.3.6 _ Win32_bin.zip Lua-5.3.6 _ Win32_bin.zip Lua-5.3.6 _ Win32_bin.zip Lua-5.3.6 _ Win32_bin.zip Lua-5.3.6 _ Win32_bin.zip	Nome Current Single Constraints Pre-compiled Lua libraries and executables Nome Overview Nistaliation History Corfiguration Packaging Download License Download License License Lua-54.2_Source code and documentation are available from the files page. Distaliation History Contact us Manual Configuration Packaging Download License License Lua-54.2_Sources.tar.gz Source Code and Makefiles Lua-54.2_Win32_bin.zip Windows x86 Executables Lua-54.2_Win32_bin.zip Lua-54.2_Win34_bin.zip Windows x84 Executables Lua-54.2_Win34_bin.zip Windows x84 Executables Lua-54.2_Win84_dilw6_lib.zip Windows x84 Executables Lua-54.2_Linux84_64_bin.tar.gz Linux x84 Executables Lua-54.2_Win84_dilw6_lib.zip Windows x84 Executables Lua-54.2_Linux84_64_bin.tar.gz Linux x84 Executables Lua-54.5_Sources.tar.gz Source Code and Makefiles Lua-54.2_Linux84_64_bin.tar.gz Linux x84 Executables Lua-54.5_Sources.tar.gz Source Code and Makefiles Lua-54.3_Sources.zip Source Code and	Automation Events Pre-compiled Lua libraries and executables Home Overview Motivation Installation History Credits Contact us Manual Configuration Packaging Download License Download Manual Configuration Packaging Download License The binaries, source code and documentation are available from the Source files page. 	Vertexite Proceeding of the project situation of t	Vertices Execute the interview of the intervi	Very Very Monitoria Control Con

will direct the user to a SourceForge page, where the binary will be downloaded.



Move the downloaded Zip file to any location to store the binaries permanently.

Here, we will be using the "D:\Program Files\Lua" as the location. After moving the Zip file, simply extract its content using any compression utility (Ex: Windows Explorer, 7zip, WinRar).



We need to add the location of Lua binaries to the system PATH so that Windows can call Lua from anywhere in the system.

Step 4.1. Navigate to Environment Variables. (Open Windows Explorer, right-click on This PC, and select properties.)



4.2. Click on "Advanced System Settings" in the screen that appears and then click on "Environment Variables".

Settings		- 🗆 X
命 Home	About	
Find a setting	Your PC is monitored and protected.	This page has a few new settings
System	See details in Windows Security	Some settings from Control Panel have moved here, and you can copy your PC info so it's easier to share.
🖵 Display	Device specifications	
11	System Properties X	Related settings
۹७ Souna	Computer Name Hardware Advanced System Protection Remote 3.79	BitLocker settings
Notifications & actions	You must be logged on as an Administrator to make most of these changes.	Device Manager
J Focus assist	Performance Visual effects, processor scheduling, memory usage, and virtual memory	Remote desktop
	Settings	System protection
O Power & sleep	User Profiles	Advanced system settings
📼 Storage	Desktop settings related to your sign-in	Rename this PC (advanced)
	Settings	
	Startup and Recovery	😥 Get help
曰: Multitasking	System startup, system failure, and debugging information	Give feedback
D Projecting to this PC	Settings	
	Environment Variables	
X Shared experiences		
🛱 Clipboard	OK Cancel Apply	
	Experience Windows Feature Experience Pack 120.2212.2020.0	
> ✓ Remote Desktop	Сору	
1 About		
		Step

4.3. In the system variables section, add the location of the Lua executables as a new entry for the Path variable.

er variables for bisin			
Variable	Value		,
ANDROID HOME	E:\Bisina\Downloads\Android\android-sdk		_
ANDROID SDK ROOT	F:\Bisina\Downloads\Android\android-sdk		
BESIEGE_GAME_ASSEMBLIES	E:/Games/SteamLibrary/steamapps/commo	n/Besiege/B	esiege_Dat
BESIEGE_UNITY_ASSEMBLIES	E:/Games/SteamLibrary/steamapps/commo	n/Besiege/B	esiege_Dat
ChocolateyLastPathUpdate	132606530855437616		
ChocolateyToolsLocation	C:\tools		
FileBot	C:\Program Files\FileBot		
	New	Edit	Delete
stem variables			
Variable	Value		1
NUMBER_OF_PROCESSORS	12		
OS	Windows_NT		
Path	C:\Program Files\AdoptOpenJDK\jre-16.0.1.9)-hotspot\bi	in;C:\Progr
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF	;.WSH;.MSC	;.PY;.PYW
POWERSHELL_DISTRIBUTIO	MSI:Windows 10 Pro		
PROCESSOR_ARCHITECTURE	AMD64		
PROCESSOR IDENTIFIER	AMD64 Family 23 Model 113 Stepping 0. Aut	henticAMD	
	New	Edit	Delete
		3	
		OK	Cancel
		OK	Cancel
environment variable		ОК	Cancel
environment variable		ОК	Cancel
environment variable	/H22nec	ОК	Cancel
environment variable C:\Windows\System32\Op	penSSH\	ОК	Cancel New
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV	penSSH\ IDIA Corporation\PhysX\Common	OK	Cancel New
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (DenSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR	OK	Cancel New Edit
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\ProgramData\chocolat	benSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin	OK	Cancel New Edit
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\ProgramData\chocolat C:\Program Files (x86)\Box	benSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\	OK	Cancel New Edit Browse
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\ProgramData\chocolat C:\Program Files (x86)\Box C:\Program Files (x86)\Box	DenSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ benJDK\jdk-11.0.4+11\bin	OK	Cancel New Edit Browse
: environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\AdoptOp C:\Program Files\AdoptOp	benSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ benJDK\jdk-11.0.4+11\bin	OK	Cancel New Edit Browse
: environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files (x86)\Box C:\Program Files\AdoptOp C:\Program Files\dotnet\ C:\Program F	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ penJDK\jdk-11.0.4+11\bin	OK	Cancel New Edit Browse Delete
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Data\chocolat C:\Program Files (x86)\Box C:\Program Files\AdoptOp C:\Program Files\dotnet\ C:\Program Files\dotnet\ C:\Program F	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ penJDK\jdk-11.0.4+11\bin	OK	Cancel New Edit Browse Delete
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\dotnet\ C:\Program F %M2_HOME%\bin C:\ProgramData\chocolat	benSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ benJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven	OK	Cancel New Edit Browse Delete Move Up
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\dotnet\ C:\Program F %M2_HOME%\bin C:\ProgramData\chocolat C:\Program Files\PowerSh	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ penJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven ell\7\	OK	Cancel New Edit Browse Delete Move Up
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\dotnet\ C:\Program F %M2_HOME%\bin C:\Program Data\chocolat C:\Program Files\PowerSh C:\Program Files\PowerSh	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ penJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven iell\7\	OK	Cancel New Edit Browse Delete Move Up Move Down
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files (x86)\Box C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\Din C:\Program Files\PowerSh C:\tools\BCURRAN3 C:\Program Files\Microsof	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin <cryptor\bin\ penJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven ell\7\</cryptor\bin\ 	OK	Cancel New Edit Browse Delete Move Up Move Down
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\Din C:\Program Files\PowerSh C:\tools\BCURRAN3 C:\Program Files\Microsof C:\Program Files\Microsof	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ penJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven ell\7\ ft VS Code\bin rd\	OK	Cancel New Edit Browse Delete Move Up Move Down
: environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\dotnet\ C:\Program Files\Din C:\Program Files\PowerSh C:\tools\BCURRAN3 C:\Program Files\Microsof C:\Program Files\WireGua C:\Program Files\WireGua	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin <cryptor\bin\ penJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven iell\7\ ft VS Code\bin rd\</cryptor\bin\ 	OK	Cancel New Edit Browse Delete Move Up Move Down Edit text
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\Din C:\Program Files\PowerSh C:\tools\BCURRAN3 C:\Program Files\Microsof C:\Program Files\Microsof C:\Program Files\WireGua C:\Program Files\WireGua C:\Program Files\WireGua	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin <cryptor\bin\ penJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven ell\7\ ft VS Code\bin rd\</cryptor\bin\ 	OK	Cancel New Edit Browse Delete Move Up Move Down Edit text
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Data\chocolat C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\Din C:\Program Files\PowerSh C:\tools\BCURRAN3 C:\Program Files\Microsof C:\Program Files\Microsof C:\Program Files\WireGua C:\Program Files\WireGua C:\Program Files\WireGua	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ penJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven ell\7\ ft VS Code\bin rd\	OK	Cancel New Edit Browse Delete Move Up Move Down Edit text
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\Docker\I C:\Program Files\WireGua C:\Program Files\WireGua C:\Program Files\WireGua C:\Program Files\WireGua C:\Program Files\WireGua C:\Program Files\WireGua C:\Program Files\Amazon C:\Program Files\Amazon C:\Program Files\Amazon	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ penJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven ell\7\ ft VS Code\bin rd\ tnet\ \AWSCLIV2\ Docker\resources\bin	OK	Cancel New Edit Browse Delete Move Up Move Down Edit text
environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\NVIDIA (C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\PowerSh C:\Program Files\Microsof C:\Program Files\Docker\I C:\Program Files\Docker\I	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ penJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven eII\7\ ft VS Code\bin rd\ tnet\ AWSCLIV2\ Docker\resources\bin esktop\version-bin	OK	Cancel New Edit Browse Delete Move Up Move Down Edit text
t environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files \NVIDIA O C:\Program Files\NVIDIA O C:\Program Files (x86)\Box C:\Program Files (x86)\Box C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\DocerSh C:\tools\BCURRAN3 C:\Program Files\Microsof C:\Program Files\Microsof C:\Program Files\Microsof C:\Program Files\Microsof C:\Program Files\Microsof C:\Program Files\Microsof C:\Program Files\Microsof C:\Program Files\Docker\I C:\Program Files\Docker\I C:\Program Files\Docker\I C:\Program Files\Docker\I	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ penJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven ell\7\ ft VS Code\bin rd\ tnet\ \AWSCLIV2\ Docker\resources\bin esktop\version-bin		Cancel New Edit Browse Delete Move Up Move Down Edit text
: environment variable C:\Windows\System32\Op C:\Program Files (x86)\NV C:\Program Files\NVIDIA (C:\Program Data\chocolat C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\AdoptOp C:\Program Files\Docker\I C:\Program Files\WireGua C:\Program Files\WireGua C:\Program Files\WireGua C:\Program Files\WireGua C:\Program Files\WireGua C:\Program Files\Amazon C:\Program Files\Amazon C:\Program Files\Docker\I C:\Program Files\Docker\I C:\Program Files\Lua	penSSH\ IDIA Corporation\PhysX\Common Corporation\NVIDIA NvDLISR ey\bin kcryptor\bin\ penJDK\jdk-11.0.4+11\bin ey\lib\maven-snapshot\apache-maven ell\7\ ft VS Code\bin rd\ tnet\ \AWSCLIV2\ Docker\resources\bin esktop\version-bin	OK	Cancel New Edit Browse Delete Move Up Move Down Edit text

Check if the system identifies Lua by opening up a command prompt or a PowerShell window and typing the Lua command (Lua with the version - lua54).



Setting up a Lua development environment

Now that we have installed Lua in the system, we need a development environment to go ahead with coding. For that, we can choose between:

- A dedicated Lua IDE like ZeroBrane Studio
- A general IDE like <u>VSCode</u>

We will be using VSCode for this instance.

Step 1

Let's create a file called "lua_basic.lua" in VSCode and save that file in the desired location. Then we will type some print statements there like the following.

```
print("Hello Lua")
print(10*10)
print("We have multiplied 10 by 10")
```

× -		p lua_basic.lua - Lua_Scripts - Visual Studio Code	– 🗆 X
Ch	EXPLORER ····	₁ lua_basic.lua ×	ho II …
-	✓ OPEN EDITORS	🖬 lua_basic.lua	
	🗙 🖬 lua_basic.lua	1 print("Hello Lua")	(822.**
	∨ LUA_SCRIPTS	2 print(10*10)	1
မို့စ	া lua_basic.lua	3 print("We have multiplied 10 by 10")	
₿			
\geq			
ð			
۲			
8			
£63	> OUTLINE > ZIP EXPLORER		

Our Lua program needs to be compiled before running, so we need to create a Build Task. To do that, click on Terminal Menu, then Run Built Task and select the Configure Build Task Option.

(There will be different build task templates depending on the VSCode configuration. Select "Create tasks.json from template" and finally the "Others" option to define a custom build task.)

Run Build Task



gure Build Task

∢	File Edit	Selection	View	Go	Run	Terminal	Help	lua_basic.lua - Lua_Scripts - Visual Studio Code	– 🗆 X	
ф	EXPLORE	R								
	\sim open ed	ITORS				No bui	ld task to	n found. Configure Build Task		
0	× 🖬	lua_basic.lu	ıa						E224"	
~	\sim LUA_SCR	IPTS						2 print(10*10)	т	
° ~	of lu	a_basic.lua						printij"We have multiplied 10 by 10")	•	
2										
										. ,
\sim										reat
o t	nchc ic	on fra	-m	ton	nnla	Nto				

e tasks.json from template

⋈	File Edit Selection View Go Run	Terminal Help lua_basic.lua - Lua_Scripts - Visual Studio Code	– 🗆 X
Ω	EXPLORER	Select a task to configure	
	✓ OPEN EDITORS	Create tasks.json file from template	
Q	🗙 🖬 lua_basic.lua	Grunt task detection is turned off. Enable grunt task detection	
/~	✓ LUA_SCRIPTS	Gulp task detection is turned off. Enable gulp task detection	т
80	∨ 🛱 .vscode	Jake task detection is turned off. Enable jake task detection	•
25	Iua_basic.lua		
~			
¤ >			Other
×	File Edit Selection View Go Run	Terminal Help lua_basic.lua - Lua_Scripts - Visual Studio Code	– 🗆 X
ф	EXPLORER		ho III …
	✓ OPEN EDITORS	MSBuild Executes the build target	
Q	🗙 🖬 lua_basic.lua	maven Executes common maven commands	622
/~	✓ LUA_SCRIPTS	.NET Core Executes .NET Core build command	т
80	∨ 🛃 .vscode	Others Example to run an arbitrary external command	•
2	Iua_basic.lua		
			Step

2.1. We will add the following code block to configure the task. In that code block, we have defined a task called "Run Lua" that will run on the shell with the command "lua54". That command will take the current file as the argument and carry out a build operation.







Open up the "lua_basic.lua" file again. Then go to the "Terminal" again and click on "Run Build Task" or use the shortcut Ctrl+Shift+B.



will compile the file and provide us with the output.



s it! Now we have a working Lua development environment that can be used to create Lua scripts. We can use the official Lua <u>reference manual</u> to explore the Lua language further.

Lua is powerful

Lua is a powerful scripting language that has limitless potential to add functionality to any program

on a multitude of platforms to suit any use case.

Related reading

- <u>BMC DevOps Blog</u>
- Top 5 Best Practices for Software Development
- Java vs Go: What's The Difference?
- Python Development Tools: Your Python Starter Kit
- DevOps Engineer Roles & Responsibilities
- <u>API/Developer Portals: How To Create Great API Portals</u>