

Hashicorp Terraform Associate Certification (Exam 003)

*Upskill and certify your IT infrastructure automation skills
with this exam-cum-study guide*

Kimiko Lee



Copyright © 2024 by GitforGits

All rights reserved. This book is protected under copyright laws and no part of it may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without the prior written permission of the publisher. Any unauthorized reproduction, distribution, or transmission of this work may result in civil and criminal penalties and will be dealt with in the respective jurisdiction at anywhere in India, in accordance with the applicable copyright laws.

Published by: GitforGits

Publisher: Sonal Dhandre

www.gitforgits.com

support@gitforgits.com

Printed in India

First Printing: October 2023

ISBN: 9788119177295

Cover Design by: Kitten Publishing

For permission to use material from this book, please contact GitforGits at support@gitforgits.com.

Prologue

This is the beginning of a significant learning journey that will hopefully help you become a master of Terraform. Terraform is a game-changing technology for managing infrastructure as code, and this book will show you what you need to know to become an expert or a certified user. Whether you're on-premises, in the cloud, or using a combination of the two, Terraform gives you the ability to code and manage your infrastructure just like software development. This book will help you get ready for HashiCorp's Terraform Certification, which signifies your competence in this field, by providing both theoretical background and practical tasks.

Skilled Terraform practitioners are in high demand due to the platform's great potential and its widespread adoption in the IT industry. The way we plan for, implement, and oversee infrastructure management has undergone a sea change, and this technology is at the vanguard of that change. Using a declarative language, Terraform eliminates the need for ad hoc scripts or manual setups and instead generates code that is concise, easy to understand, and reusable. This book is an attempt to encapsulate that spirit by providing a systematic, all-encompassing way to learn Terraform, beginning with the fundamentals and on to more complex subjects.

We start by taking a deep dive into Terraform's fundamental architecture, learning all about its providers, resources, and state files. As we go along, we'll use real-life examples and scenarios to bolster the theoretical notions. A lot of them will be set in the framework of CloudOptic, a made-up but totally genuine tech business. The ability to develop code that is both reusable and easy to maintain is essential for any infrastructure project, and you will learn how to use Terraform modules to accomplish just that. Module types and their practical applications are covered extensively, with a special emphasis on root, child, and published modules.

We go beyond providing the bare minimum in terms of infrastructure. The book walks you through the ins and outs of Terraform's more complex features and capabilities, including manipulating states, conditional logic, and dependencies. In order to manage complicated, enterprise-level infrastructures, these subjects are essential. You'll also find out how to deal with configuration drift, a critical but frequently neglected part of infrastructure management. You will not only learn the content, but also master it, thanks to the real-world activities and quizzes that pop up at regular intervals throughout the chapters.

Additionally, the book includes extensive coverage of best practices for scalability, environment isolation, and reusable modules. Finally, we get into how to integrate Terraform with other technologies, such as CDKTF, to provide you even more tools for infrastructure management. Reading this book will open doors to exciting new opportunities in infrastructure management; it's about more than just picking up a tool. You will be well-prepared to become an expert in the field of Terraform after reading this book, which provides practical examples, real-world applications, and an easy-to-understand writing style. Thank you for joining me on this path of self-discovery.

Content

Prologue	iii
Content	iv
Preface	xix
Acknowledgement.....	xxii
Chapter 1: Infrastructure-as-Code and Terraform.....	23
What is Infrastructure as Code (IaC)?	24
The Pre-IaC Era.....	24
Emergence of IaC.....	24
Current Prospects and Adaptability of IaC.....	24
Advantages of IaC	25
Cost-Efficiency.....	25
Improved Collaboration.....	25
Speed and Agility	25
Consistency and Standardization	26
Version Control for Infrastructure.....	26
Improved Disaster Recovery	26
Enhanced Security and Compliance.....	26
Reusability and Modularity.....	26
Documentation and Knowledge Sharing	26
Monitoring and Analytics	27
IaC Landscape	27
Categories of IaC Tools.....	27
<i>Configuration Management Tools</i>	27
<i>Orchestration Tools</i>	27
Imperative vs Declarative Languages.....	27
<i>Imperative Languages</i>	27
<i>Declarative Languages</i>	28
Multi-Cloud and Hybrid Cloud IaC	28
IaC and Containers.....	28
Code Repositories and Version Control	28
IaC Security Tools	28

Infrastructure Monitoring and Telemetry	28
Policy as Code	29
IaC in CI/CD Pipelines	29
Terraform in IaC Ecosystem.....	29
Provider-Agnostic Nature.....	29
Terraform's State Management.....	29
Modular Design and Reusability.....	30
Policy as Code.....	30
Integration with CI/CD Pipelines	30
Collaboration and Workspaces	30
Extensibility and Custom Providers	30
Drift Management.....	30
Key Concepts and Terminologies	31
<i>Terraform Configuration</i>	31
<i>Providers</i>	31
<i>Resources</i>	31
<i>Variables and Outputs</i>	31
<i>State File</i>	31
<i>Backend</i>	32
<i>Plan and Apply Phases</i>	32
<i>Terraform Modules</i>	32
<i>Terraform Workspaces</i>	32
<i>Terraform Cloud</i>	32
<i>Sentinel</i>	32
<i>Terraform Import</i>	32
<i>Drift Management</i>	33
<i>Secret Management</i>	33
<i>Terraform CLI Commands</i>	33
Setting up Terraform on Linux	33
<i>Update Your Package Manager</i>	33
<i>Download Terraform</i>	33
<i>Unzip the Archive</i>	34
<i>Move Binary to /usr/local/bin</i>	34
<i>Verify Installation</i>	34
<i>Configure Environmental Variables</i>	34
<i>Create a Workspace</i>	34
<i>Initialize the Workspace</i>	35

<i>Create a Simple Configuration File</i>	35
<i>Run Terraform Plan</i>	35
<i>Apply the Configuration</i>	35
Summary	36
Test Exercise	36
Chapter 2: Understanding Terraform Basics	39
Understanding Providers	40
Introduction.....	40
How Providers Work?	40
Version control	40
Cross-Provider Support.....	40
Provider Credentials	41
Plugin-Based Architecture	41
Basics	41
How Plugins Are Loaded?	41
Plugin Types	41
<i>Provider Plugins</i>	41
<i>Provisioner Plugins</i>	42
Custom Plugins	42
Dependency Resolution in Plugins.....	42
Security Implications.....	42
Version Management	42
Debugging and Logs	42
Writing Configuration Files for CloudOptic	43
What are Configuration Files?	43
<i>Components of Configuration Files</i>	43
Sample Program: Writing a Configuration File	43
<i>Provider</i>	43
<i>EC2 Resource</i>	44
<i>S3 Resource</i>	44
<i>Variables and Outputs</i>	45
<i>Complete Configuration</i>	45
Resource Blocks and Attributes	47
What Are Resource Blocks?	47
Structure of a Resource Block	47

Understanding Attributes	48
Sample Program: Using Resource Blocks and Attributes	48
Variables and Outputs	49
Terraform Variables	49
<i>Declaration</i>	49
<i>Utilization</i>	50
Outputs in Terraform	50
<i>Declaration</i>	50
<i>Utilization</i>	50
Sample Program: Using Variables and Outputs	51
<i>Variable Configuration</i>	51
<i>Resource Configuration with Variables</i>	51
<i>Output Configuration</i>	52
Benefits of Variables and Outputs	52
Popular Terraform Commands	53
terraform init	53
terraform validate	53
terraform plan	53
terraform apply	53
terraform destroy	54
terraform state	54
terraform import	54
terraform fmt	54
terraform workspace	54
terraform output	55
Sample Program: Using Commands	55
Understanding Terraform State	55
Mechanics of Terraform State	56
terraform.tfstate	56
Components of State File	57
Viewing and Modifying Terraform State	58
Viewing Terraform State	58
<i>Terraform Show</i>	58
<i>JSON Output</i>	59
<i>State List</i>	59
Modifying Terraform State	59

<i>Moving Resources</i>	60
<i>Removing Resources</i>	60
<i>Importing Resources</i>	60
Summary	60
Test Exercise	61
Chapter 3: Terraform Workflow	63
Terraform Workflow Overview	64
Terraform Workflow Phases	65
Write Phase.....	65
<i>Example</i>	65
Plan Phase	66
<i>Example</i>	66
Create/ Apply Phase	66
<i>Example</i>	66
Initializing Terraform Directory	67
<i>Create Terraform Configuration Files</i>	67
<i>Open Terminal and Navigate to Directory</i>	68
<i>Initialize the Directory</i>	68
<i>Expected Output</i>	68
<i>Troubleshooting Tips</i>	68
Validating Terraform Configuration	69
What is Configuration Validation?	69
<i>Syntax Check</i>	69
<i>Attribute Check</i>	69
How to Validate Configuration.....	69
<i>Open Your Terminal</i>	69
<i>Run the Validate Command</i>	70
<i>Expected Output</i>	70
Tips for Validation	70
Advanced Validation.....	70
Execution Plan	70
What is an Execution Plan?	71
Importance of Execution Plan.....	71
Step-By-Step Execution Plan	71
<i>Navigate to Terraform Directory</i>	71
<i>Run terraform plan</i>	71

<i>Review the Plan</i>	71
<i>Check Dependencies</i>	72
Additional Flags and Options	72
What to Look For?	72
Applying Changes with Terraform	73
Importance of Applying Changes	73
Step-By-Step To Apply Changes	73
<i>Initiate Apply Command</i>	73
<i>Review and Approve</i>	73
<i>Monitor the Output</i>	73
<i>Validate in CloudOptic</i>	74
<i>State File Update</i>	74
Additional Commands and Flags	74
Handling Unintended Changes	75
How Unintended Changes Occur?	75
<i>Configuration Drift</i>	75
<i>Human Error</i>	75
<i>Versioning Issues</i>	75
<i>Interdependencies</i>	75
Consequences of Unintended Changes	75
How to Minimize the Risk of Unintended Changes?	75
<i>Thoroughly Review Execution Plan</i>	76
<i>Use terraform destroy Wisely</i>	76
<i>Test Environments</i>	76
<i>Version Control</i>	76
<i>Backups and Snapshots</i>	76
<i>Policy as Code</i>	76
<i>Limit Permissions</i>	76
<i>Locking State Files</i>	76
Formatting and Style in Terraform	77
Terraform Formatting and Style Guidelines	77
<i>Indentation</i>	77
<i>Attribute Naming</i>	77
<i>Resource Naming</i>	77
<i>Comments and Documentation</i>	78
<i>File Structure</i>	78
Terraform fmt Command	78

<i>What Does terraform fmt Do?</i>	78
<i>Usage Example</i>	79
Verbose Logging	79
How to Enable Verbose Logging in Terraform?	79
<i>Enabling Verbose Logging</i>	79
<i>Output Log File</i>	80
Verbose Logging in CloudOptic with Terraform	80
Summary	81
Test Exercise	82
Chapter 4: State Management and Backend	85
Introduction to Terraform Backend	86
Local Backend.....	86
Storage and Security Concerns.....	86
How it Works?	86
Transitioning to Other Backends	86
State Locking and Unlocking	87
How State Locking Works?	87
Locking in Local Backend.....	87
Sample Program: Getting Local Backend Locked.....	87
Unlocking State File	88
Backend and Cloud Integration	89
Relationship between Backend and Cloud.....	89
Sample Program: Integrate AWS S3 Backend.....	89
Remote State Backend and Options	91
Terraform Cloud as a Remote Backend	91
<i>Key Features</i>	91
<i>Sample Configuration</i>	91
Consul as a Remote Backend	92
<i>Key Features</i>	92
<i>Sample Configuration</i>	92
Terraform Cloud vs. Consul.....	93
Backend Configuration	93
Using Terraform Cloud with CloudOptic.....	93
Using AWS S3 with CloudOptic:	94
Whom to Choose?.....	95

Managing Resource Drift.....	95
Tools and Techniques for Detecting Drift.....	96
<i>Terraform Refresh.....</i>	<i>96</i>
<i>Terraform Plan.....</i>	<i>96</i>
Sample Program: Detect and Manage Drift.....	96
<i>Detecting Drift.....</i>	<i>96</i>
<i>Managing Drift.....</i>	<i>96</i>
Integrate Drift Detection into CI/CD	98
Terraform Initialization.....	98
Drift Detection Script.....	99
Conditional Apply	100
Notification	100
Secrets Management	101
Why It Matters?	101
Basic Secrets Handling Methods	102
Advanced Secrets Management	102
Sensitive State Files	103
Encrypt State Files	104
Setting Up Encrypted Storage.....	104
<i>Create an Encrypted S3 Bucket</i>	<i>104</i>
<i>Terraform Backend Configuration.....</i>	<i>104</i>
Utilizing AWS KMS.....	104
Configuring Resource with Encrypted State Files.....	105
Backing Up State Files.....	106
Local Backend Backups	106
<i>Manual Copy</i>	<i>106</i>
<i>Version Control Systems</i>	<i>106</i>
<i>Scripted Backups</i>	<i>106</i>
Terraform Cloud Backups	106
Consul Backend Backups.....	107
<i>Snapshot Mechanism</i>	<i>107</i>
<i>Manual Copy from KV Store</i>	<i>107</i>
Summary	107
Test Exercise.....	108
Chapter 5: Advanced Configuration Techniques	110

Advanced Terraform Overview	111
Secure Secret Injection.....	111
Significance of Secret Injection	111
Function of the HashiCorp Vault.....	112
Setting up Vault.....	112
<i>Injecting Secrets</i>	112
<i>Retrieving Secrets</i>	112
<i>Using Secrets</i>	113
Collection and Structural Types.....	113
Lists	114
Sets	114
Maps.....	114
Objects.....	115
Tuples	115
Sample Program: Using Collections and Structural Types.....	116
Resource Addressing and Usage	117
Syntax.....	118
Qualified Resource Address	118
Sample Program: Establishing Dependency	118
For Targeted Actions	119
For Outputs	119
For Data Sources	120
Functions in Terraform.....	120
Categories of Functions.....	121
<i>Numeric Functions</i>	121
<i>String Functions</i>	122
<i>Collection Functions</i>	122
<i>Logical Functions</i>	122
<i>Encoding Functions</i>	123
<i>Time Functions</i>	123
Function Nesting.....	123
Built-in Dependency Management.....	124
Explicit Dependencies	124
Implicit Dependencies	125
Meta-Arguments for Controlling Dependencies	126

Dependency Blocks and Modules	126
Order of Execution.....	127
Conditional Attributes.....	127
Conditional Operators.....	128
Count.....	128
Conditional Expressions	129
Local Values	129
Conditional Creation and Modules	130
Dynamic Blocks and Conditional Logic	130
Multiple Conditions with Logical Operators.....	131
Conditional Blocks with Dynamic Blocks	132
Nested Conditional Statements	132
Null Values for Optional Attributes	133
Summary	133
Test Exercise.....	134
Chapter 6: Terraform Modules	136
Introduction to Terraform Modules.....	137
Deep Dive into Terraform Modules.....	137
Encapsulation and Abstraction.....	137
Reusability.....	138
Version Control.....	138
Standardization and Best Practices.....	138
Resource Organization and Management.....	138
Dependency Resolution	138
Parameterization.....	138
Types of Modules.....	139
Root Modules.....	139
Child Modules	139
Published Modules	139
Additional Module Types	140
<i>Environment Modules</i>	<i>140</i>
<i>Meta Modules.....</i>	<i>140</i>
<i>Data Modules.....</i>	<i>140</i>
Module Source Options	140
Local Source Paths	140

GitHub and Git Source	141
HTTP URLs	141
S3 Buckets	141
Terraform Registry	142
Module Source Best Practices	142
Module Inputs and Outputs	143
Module Inputs	143
<i>Key Points about Module Inputs</i>	143
Module Outputs	144
<i>Key Points about Module Outputs</i>	144
Terraform Variables	145
Variables Overview	145
<i>Data Types and Complex Objects</i>	145
<i>Variable Validation</i>	146
<i>Environmental Variables</i>	147
Module-Specific Role of Variables	147
Variables for Reusability and Versioning	147
Setting Module Version in Terraform	148
Why Module Versions?	148
Version Constraints	148
Using Local Modules with Versioning	149
Dynamic Version Pinning	149
Immutability and Semantic Versioning	150
Exploring Terraform Module Registry	150
Overview	150
Basic Navigation and Features	151
Using Module from Registry	151
Using Module Outputs	151
Forking and Customizing Modules	151
Private Module Registry	152
Versioning and the Registry	152
Auditing and Security Concerns	152
Creating Custom Modules	152
Need for Custom Modules	152
Advantages of Custom Modules	152

Groundwork for Custom Module.....	153
<i>main.tf</i>	153
<i>variables.tf</i>	154
<i>outputs.tf</i>	155
Using Custom Module	155
Managing Module Dependencies.....	156
Dependency Blocks within Modules	156
Data Blocks and Dependency.....	157
Conditional Dependencies.....	157
Nested Module Dependencies	158
Dependency Locking with Version Constraints	158
Summary	159
Test Exercise.....	160
Chapter 7: Terraform Import and Maintenance.....	162
Overview.....	163
Terraform Import and Its Utilization.....	163
Importance of Terraform Import	164
The Mechanics of Import.....	164
Challenges and Caveats	165
Sample Program: Terraform Import for Complex Resources.....	165
Pre-requisites.....	165
Setup Configuration Blocks.....	165
Importing the Resources.....	166
Configuration Importer Tools	167
Beware of Drift.....	168
Managing Configuration Drift	168
Detecting Configuration Drift	168
Managing Drift	168
Reconcile Drift.....	169
Manual Reconciliation	169
Drift Detection Tools.....	170
Terraform Resource Targeting	170
Syntax for Resource Targeting.....	170
Use Cases for Resource Targeting	170
<i>Selective Updates</i>	170

<i>Dependent Resource Modification</i>	171
<i>Debugging</i>	171
<i>Targeting with Resource Addressing</i>	171
<i>Targeting Multiple Resources</i>	171
Automating Resource Targeting in CI/CD	171
Terraform State Manipulation Commands	172
<i>terraform state list</i>	173
<i>terraform state show</i>	173
<i>terraform state mv</i>	173
<i>terraform state rm</i>	173
<i>terraform state pull</i>	173
<i>terraform state push</i>	174
<i>terraform state replace-provider</i>	174
State File Backup in CI/CD	174
Limitations and Precautions	175
Updating Terraform Codebase	175
Conceptual Importance	175
Sample Program: Keeping Codebase Updated	176
<i>Version Pinning</i>	176
<i>Regular Checks and Audits</i>	176
<i>Manual Review</i>	176
<i>Test Environment</i>	177
<i>Gradual Rollout</i>	177
<i>Monitoring and Alerting</i>	178
Summary	178
Test Exercise	178
Chapter 8: Best Practices and Design Patterns	181
Introduction to Advanced Terraform Practices	182
Terraform Style Guide and Best Practices	182
Formatting Conventions	182
<i>Indentation</i>	182
<i>Naming Conventions</i>	183
<i>Comments and Documentation</i>	183
<i>Resource Blocks</i>	183
<i>Variables and Outputs</i>	184
<i>Modules</i>	184

Sample Program: Using Style Principles	184
Strategic Purpose of Variables and Locals.....	185
Leveraging Dynamic Blocks	186
Effective Use of Conditionals.....	187
File Organization for Modularity	187
Repository Structure	187
Sample Program: Adopting Style Guidelines	188
<i>File: vpc.tf</i>	188
Utilizing Reusable Modules.....	189
Why Reusable Modules?	189
Anatomy of a Module.....	190
Sample Program: Reusable Module	190
<i>modules/vpc/main.tf</i>	190
<i>modules/vpc/variables.tf</i>	190
<i>dev/main.tf</i>	191
<i>prod/main.tf</i>	191
Extending VPC Module to Include EC2.....	192
Creating Module Instances for Different Environments.....	194
Versioning Modules	195
Count and For_each in Modules	195
Isolated Environments	196
Understanding Concept	196
Sample Program: Establishing Isolated Environment.....	196
<i>development.tfvars</i>	197
Separate Directories vs. Isolated Workspaces.....	199
Understanding Separate Directories	199
Sample Program: Separate Directories vs. Workspaces	199
<i>Directory Structure</i>	199
<i>Risk Mitigation</i>	200
<i>Synchronization Challenge</i>	201
<i>Version Control</i>	201
Performance and Scalability Indicators	201
Scalability Indicators	201
<i>Resource Utilization</i>	201
<i>Infrastructure Elasticity</i>	201
<i>Queue Depth</i>	201

Performance Indicators	202
<i>Latency</i>	202
<i>Throughput</i>	202
<i>Error Rates</i>	202
Practical Measures for CloudOptic	202
<i>Auto-Scaling Groups</i>	202
<i>State Backends</i>	202
<i>Resource Tagging</i>	202
<i>Code Reviews</i>	202
<i>Parallelization and Workspaces</i>	203
<i>Caching and Content Delivery</i>	203
<i>Monitoring and Alerts</i>	203
Integrating Terraform with CDKTF	203
<i>Install CDKTF CLI</i>	203
<i>Initialize New CDKTF Project</i>	203
<i>Generate Constructs</i>	204
<i>Reference Existing Resources</i>	204
<i>Add New Resources</i>	204
<i>Synthesize</i>	204
<i>CDKTF Deploy</i>	205
<i>Make Iterative Changes</i>	205
Precautions	205
Summary	205
Test Exercise	206
Index	208
Epilogue	210

Preface

This all-inclusive book will take you from zero to hero in Terraform, a revolutionary tool from HashiCorp for managing and deploying datacenter infrastructure in the cloud or on-premises. Terraform is the key to unlocking the world of IaC. With the knowledge you gain from this book, you will be prepared to confidently take HashiCorp's Terraform Certification exam.

An extensive introduction to Terraform's architecture, covering its providers, resources, and state management features, sets the stage in the first chapter of the book. After you finish the first few chapters, you'll be behind the wheel, practicing for real-world situations with the help of several exercises. Using CloudOptic, a made-up but believable tech corporation, as an example, you will learn the ins and outs of module development and management. Starting with the basics and working its way up to advanced topics, this book walks you through every step of a Terraform project's lifetime.

Still, it's only the start. State manipulation commands, resource targeting, and handling configuration drifts are just a few of the tough but vital topics covered thoroughly in this book. With examples and practical suggestions provided, you will learn how to work with variables, output values, dependencies, and state files. This book will teach you all you need to know about reusable modules and more, so you can create and integrate your own.

The book's inclusion of exercises and real-world scenarios further increases its usefulness by providing ample opportunity to demonstrate mastery of the material. Every level of Terraform knowledge, from the most fundamental to the most complex, is covered thoroughly, preparing you to face any problem that comes your way.

In this book you will learn how to:

- Understand the fundamentals, including providers, resources, and state files.
- An in-depth look at making reusable modules, dealing with dependencies, and using versioning.
- Applicable, real-world context learning through the made-up corporation CloudOptic.
- Acquire knowledge of commands for manipulating states, handling drifts, and version control.
- Gain a grasp of the most important metrics and best practices for scaling Terraform systems.
- Proficiency in maintaining the Terraform codebase for future versions and ensuring compatibility.
- Instructions for setting up a connection between Terraform and the CDKTF.

- Code in accordance with industry standards to make Terraform scripts more dependable and cleaner.
- Prep for HashiCorp's Terraform Certification with tough quizzes and exercises.

GitforGits

Prerequisites

Anyone with a basic understanding of HashiCorp Terraform and a background in cloud, IT operations, devops, or software development can apply for this certification. Prior knowledge of on-premise and cloud architectural is preferred although not mandatory.

Codes Usage

Are you in need of some helpful code examples to assist you in your programming and documentation? Look no further! Our book offers a wealth of supplemental material, including code examples and exercises.

Not only is this book here to aid you in getting your job done, but you have our permission to use the example code in your programs and documentation. However, please note that if you are reproducing a significant portion of the code, we do require you to contact us for permission.

But don't worry, using several chunks of code from this book in your program or answering a question by citing our book and quoting example code does not require permission. But if you do choose to give credit, an attribution typically includes the title, author, publisher, and ISBN. For example, "Hashicorp Terraform Associate Certification (Exam 003) by Kimiko Lee".

If you are unsure whether your intended use of the code examples falls under fair use or the permissions outlined above, please do not hesitate to reach out to us at support@gitforgits.com.

We are happy to assist and clarify any concerns.

Acknowledgement

I owe a tremendous debt of gratitude to GitforGits, for their unflagging enthusiasm and wise counsel throughout the entire process of writing this book. Their knowledge and careful editing helped make sure the piece was useful for people of all reading levels and comprehension skills. In addition, I'd like to thank everyone involved in the publishing process for their efforts in making this book a reality. Their efforts, from copyediting to advertising, made the project what it is today.

Finally, I'd like to express my gratitude to everyone who has shown me unconditional love and encouragement throughout my life. Their support was crucial to the completion of this book. I appreciate your help with this endeavour and your continued interest in my career.

CHAPTER 1:

INFRASTRUCTURE-AS-

CODE AND TERRAFORM