AWS CloudFormation Mastery: From Beginner to Advanced

A Complete Course on Infrastructure as Code with AWS CloudFormation

---

Course Overview

This course teaches you **AWS CloudFormation** from the ground up, covering **YAML/JSON templates, stacks, best practices, and advanced automation techniques**. By the end, you'll be able to **deploy complex architectures reliably** using Infrastructure as Code (IaC).

Module 1: CloudFormation Fundamentals

1.1 Introduction to IaC & CloudFormation

✅ What is AWS CloudFormation?

✅ Benefits over manual AWS console deployments

✅ How CloudFormation compares to Terraform, CDK, and Pulumi

1.2 CloudFormation Core Concepts

📌 **Stacks** – Deployable units of AWS resources

📌 **Templates** – JSON/YAML files defining infrastructure

📌 **Change Sets** – Preview changes before applying

📌 **Stack Policies** – Protect critical resources

1.3 Writing Your First Template

📝 Basic YAML/JSON structure

📝 `AWSTemplateFormatVersion` & `Description`

📝 `Resources` section (S3, EC2, IAM examples)

Hands-on Lab:

- 🔹 Deploy a simple S3 bucket using CloudFormation

Module 2: Intermediate CloudFormation

2.1 Parameters, Mappings, and Conditions

📌 **Parameters** – User inputs (e.g., `InstanceType`)

📌 **Mappings** – Region/AMI lookups

📌 **Conditions** – Deploy resources conditionally (e.g., `Prod` vs `Dev`)

2.2 Outputs & Cross-Stack References

📌 Exporting values (`Outputs`)

📌 Importing in other stacks (`Fn::ImportValue`)

2.3 Intrinsic Functions

📌 `Fn::Ref`, `Fn::GetAtt`, `Fn::Join`, `Fn::Sub`

📌 `Fn::If`, `Fn::Equals` for conditional logic

Hands-on Lab:

- 🔹 Deploy a **VPC with Public & Private Subnets** using parameters and mappings

---

Module 3: Advanced CloudFormation Techniques

3.1 Nested Stacks & StackSets

📌 **Nested Stacks** – Modularize large templates

📌 **StackSets** – Deploy across multiple AWS accounts/regions

3.2 Custom Resources & Lambda Backed Templates

📌 Extending CloudFormation with **AWS Lambda**

📌 Use cases: Dynamic config, external API calls

3.3 Drift Detection & Troubleshooting

📌 What is **drift** and how to detect it?

📌 Debugging **CREATE_FAILED** errors

📌 Using **CloudFormation Rollbacks**

Hands-on Lab:

- 🔹 Create a **custom resource** to fetch data from an external API

Module 4: CloudFormation Best Practices

4.1 Security & IAM Integration

🔐 Least privilege IAM roles for CloudFormation

🔐 Stack Policies to prevent accidental deletions

4.2 CI/CD with CloudFormation

🚀 **AWS CodePipeline + CloudFormation**

🚀 **GitHub Actions / Jenkins Integration**

4.3 Cost Optimization

💰 Spot Fleets & Auto Scaling in templates

💰 Tagging strategies for cost allocation

Hands-on Lab:

- 🔹 Set up a **CI/CD pipeline** for CloudFormation

Module 5: Real-World CloudFormation Projects

5.1 Project 1: Three-Tier Web App

🌐 **ALB + EC2 (Auto Scaling) + RDS**

🌐 Multi-AZ deployment

5.2 Project 2: Serverless API

⚡ **API Gateway + Lambda + DynamoDB**

⚡ Environment-based configurations

5.3 Project 3: Enterprise Landing Zone

🏢 **AWS Organizations + SCPs + Multi-Account VPCs**

Bonus: CloudFormation vs. Terraform vs. CDK

✅ When to use **CloudFormation** vs. alternatives

✅ Migrating from Terraform to CloudFormation

✅ Using **CDK to generate CloudFormation templates**

Course Format

🎥 **Video Lectures** (10-20 mins each)

🛠️ **Hands-On Labs** (Step-by-step AWS deployments)

📚 **Cheat Sheets & Template Examples**

🎯 **Quizzes & Certification Exam**

Who Should Take This Course?

☑ **Cloud Engineers** moving to IaC

☑ **DevOps Professionals** automating AWS

☑ **Solutions Architects** designing repeatable deployments

Prerequisites

✔ Basic AWS knowledge (EC2, S3, IAM)

✔ Familiarity with YAML/JSON

Would you like me to expand any module?

I can provide:

- 🔹 **Detailed YAML/JSON examples**
- 🔹 **Troubleshooting guides**
- 🔹 **Real-world case studies**

Let me know which part you'd like to explore further! 🚀