

# AWS CICD - CFT Training

Demo Resources :

[https://github.com/Ajay-Chidambaram/manual\\_cft\\_example](https://github.com/Ajay-Chidambaram/manual_cft_example)

[https://github.com/Ajay-Chidambaram/aws\\_codepipeline](https://github.com/Ajay-Chidambaram/aws_codepipeline)

## CloudFormation Concepts – One-Liners

- **Stack:** A single unit of deployment in CloudFormation that manages a collection of AWS resources defined in a template.
- **Change Set:** A preview of changes that CloudFormation will apply to your stack before you execute the update.
- **Drift Detection:** A feature that detects if stack resources have changed outside of CloudFormation's control.
- **Nested Stack:** A stack created within another stack to reuse templates and modularize infrastructure.
- **Parameters:** Template inputs that allow you to customize stack behavior at runtime.
- **Mappings:** Static lookup tables used to define key–value pairs for conditional configuration (e.g., region-to-AMI mapping).
- **Outputs:** Used to return values from a stack, such as resource IDs or names, for use in another stack or user display.
- **Conditions:** Control resource creation based on parameter values or environment (e.g., only create if in prod).
- **Resources:** The actual AWS infrastructure components to be created (e.g., EC2, S3, VPC).
- **Metadata:** Additional data about resources in the template; often used by tools or the AWS console.
- **Transforms:** Enable macro processing in templates, such as including AWS::Include or AWS::Serverless-2016-10-31 for SAM.

## AWS CI/CD Services – One-Liners

- **CodeCommit:** A fully managed, Git-based source control service for hosting private repositories.
- **CodeBuild:** A scalable, managed build service that compiles code, runs tests, and produces artifacts.
- **CodePipeline:** An automated CI/CD service that orchestrates the build, test, and deploy

stages of your software release process.

- **CodeDeploy:** A deployment service that automates code delivery to EC2, Lambda, or on-premise instances with various deployment strategies.
- **CodeArtifact:** A managed artifact repository for storing and sharing software packages across teams.
- **CodeStar:** An integrated service for setting up entire CI/CD projects quickly with pre-configured pipelines.
- **buildspec.yml:** A YAML file used by CodeBuild to define build commands and environment settings.
- **Deployment Group:** A set of target instances or resources used by CodeDeploy during application deployment.
- **Blue/Green Deployment:** A strategy in CodeDeploy where new versions are deployed alongside old ones, enabling safe cutover after testing.
- **Approval Action:** A manual checkpoint in CodePipeline that pauses progress until an approver reviews and approves the change.
- **Integration with CloudWatch:** All services can emit logs and metrics to CloudWatch for monitoring and debugging.

Google Canvas / Forms :

<https://g.co/gemini/share/fe0e80951a54>

[https://docs.google.com/forms/d/e/1FAIpQLSerlvYQHxtgF24IAhjsy0b2SUK\\_nAStNU8eKbh-tkdv\\_-OhKA/viewform?usp=sharing&oid=116537534101164195240](https://docs.google.com/forms/d/e/1FAIpQLSerlvYQHxtgF24IAhjsy0b2SUK_nAStNU8eKbh-tkdv_-OhKA/viewform?usp=sharing&oid=116537534101164195240)

Reference Docs :

<https://dev.to/aws-builders/deploying-your-app-with-aws-cloudformation-and-codepipeline-eea>

<https://docs.aws.amazon.com/AWSCloudFormation/latest/TemplateReference/aws-template-resource-type-ref.html>

<https://docs.aws.amazon.com/AWSCloudFormation/latest/TemplateReference/intrinsic-function-reference-foreach-example-resource.html>

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/using-cfn-nested-stacks.html>

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/conditions-section-structure.html>

Workshop Hands-On Ref :

<https://catalog.us-east-1.prod.workshops.aws/workshops/df7f72cf-4f10-4664-acb6-b30dc8d4bcf0/en-US>

<https://workshops.aws/>