



AWS CDK Operations: A Guide

This presentation provides a comprehensive guide to AWS CDK operations, empowering developers to efficiently manage infrastructure through code.

 **by The XYZ Company**

Initializing Your Project with `cdk init`

Purpose

Creates a new CDK project with a pre-defined structure and necessary files.

Command

```
cdk init app --language=typescript
```

Setting Up AWS Resources with cdk bootstrap

Purpose

Prepares your AWS environment by creating resources for CDK deployments.

Command

```
cdk bootstrap
```

Generating CloudFormation Templates with cdk synth

Purpose

Converts CDK code into an AWS CloudFormation template.

Command

```
cdk synth
```

Comparing Changes with cdk diff

Purpose

Shows the difference between local changes and the deployed stack.

Command

```
cdk diff
```

Deploying Changes to AWS with cdk deploy

Purpose

Deploys your stack to AWS, creating or updating resources.

Command

```
cdk deploy
```

Deleting the Stack with cdk destroy

Purpose

Deletes the stack and associated resources from AWS.

Command

```
cdk destroy
```

CDK Doctor

CDK Doctor is a command-line tool that helps you diagnose potential issues in your CDK projects. It checks for inconsistencies and problems before deployment, preventing common mistakes and improving reliability.

It checks for inconsistencies and problems before deployment,
preventing common mistakes and improving reliability.

`cdk doctor`



Key Takeaways

1

Streamlined Infrastructure

CDK simplifies infrastructure management with code, allowing developers to define, deploy, and manage resources.

2

Enhanced Efficiency

CDK operations automate tasks, reducing manual effort and improving operational efficiency.

3

Improved Consistency

CDK ensures consistency and repeatability across deployments, minimizing errors and deviations.