

Setting up CDK

->first install node.js,then npm

->After installing Node.js, install the AWS CDK Toolkit(npm install -g aws-cdk)

->npm install -g typescript

Creating a project

->mkdir newcdk

->cd newcdk

->cdk init app --language typescript

Commands

<code>cdk list (ls)</code>	Lists the stacks in the app
<code>cdk synthesize (synth)</code>	Synthesizes and prints the CloudFormation template for one or more specified stacks
<code>cdk bootstrap</code>	Deploys the CDK Toolkit staging stack; see Bootstrapping
<code>cdk deploy</code>	Deploys one or more specified stacks
<code>cdk destroy</code>	Destroys one or more specified stacks
<code>cdk diff</code>	Compares the specified stack and its dependencies with the deployed stacks

Create a CDK code using TS to deploy an S3 Bucket with versioning, Encryption enabled (KMS)

-> created a new stack named -NewcdkStack on starting point of CDK, newcdk.ts.

```
const app = new cdk.App();
new NewcdkStack(app, 'NewcdkStack', {});
```

-> then cdk ls to check the stack names

```
PS C:\Users\Rraks\try1\newcdk> cdk ls
NewcdkStack
```

-> then on the stack-newcdk-stack.ts- created a bucket- with bucketname-yurnkiyu787,
With versioning enabled , with kms encryption enabled, with lifecycle rules-rules for
Infrequent Access transition after 30 days and Glacier after 90 Days.

```
const bucket = new s3.Bucket(this, 'MyEncryptedBucket', {
  encryption: s3.BucketEncryption.KMS,
  versioned: true,
  bucketName: "yurnkiyu787",
  lifecycleRules: [
    {
      transitions: [
        {
          storageClass: s3.StorageClass.INFREQUENT_ACCESS,
          transitionAfter: cdk.Duration.days(30),
        },
        {
          storageClass: s3.StorageClass.GLACIER,
          transitionAfter: cdk.Duration.days(90),
        },
      ],
    },
  ],
});
```

-> then cdk deploy NewcdkStack

Bucket Versioning

Enabled

Multi-factor authentication (MFA) delete

An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

Disabled

Tags (3)

You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

Key	Value
aws:cloudformation:stack-id	arn:aws:cloudformation:ap-northeast-1:335516814222:stack/NewcdkStack/2ae66620-c18f-11ed-a648-0effd498074f
aws:cloudformation:stack-name	NewcdkStack
aws:cloudformation:logical-id	MyEncryptedBucket9A8D2FE1

Versioning has been enabled on the bucket

Default encryption

Info

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption key type

Info

AWS Key Management Service key (SSE-KMS)

Encryption key ARN

arn:aws:kms:ap-northeast-1:335516814222:key/df27d290-2423-4721-9766-f98286477ffd

Bucket Key

When KMS encryption is used to encrypt new objects in this bucket, the bucket key reduces encryption costs by lowering calls to AWS KMS. [Learn more](#)

Disabled

Encryption has been enabled

Lifecycle rules (1)

Use lifecycle rules to define actions you want Amazon S3 to take during an object's lifetime such as transitioning objects to another storage class, archiving them, or deleting them after a specified period of time. [Learn more](#)

View details

Edit

Delete

Actions

Create lifecycle rule

	Lifecycle rule name	Status	Scope	Current version actions	Noncurrent versions actions	Expired object delete markers	Incomplete multipart uploads
	Mjc1ZGM2NWEtNDk1MjY0OYTcyLTk2NmZtMGMyMmVmMTVlZGUy	Enabled	Entire bucket	Transition to Standard-IA, then Glacier Flexible Retrieval (formerly Glacier)	-	-	-

Lifecycle rules has been enabled