

Activity 1

In this activity, the process involves using AWS CloudFormation to create and manage resources based on a provided template. First, locate the template file in the GitHub repository, save it locally, and edit it in VS Code as needed. Navigate to the AWS CloudFormation Console, upload the template, and create a new stack with a preferred name. After the stack creation completes, verify the successful creation of an Amazon DynamoDB table and an Amazon RDS MariaDB instance in their respective service consoles. Finally, delete the stack, wait for it to reach the 'DELETE_COMPLETE' state, and confirm the resources are removed.

What we have done in this activity 1:

1. **DynamoDB Table:**
 - **TableName:** section1-activity-table
 - **KeySchema:** id (HASH)
 - **ProvisionedThroughput:** 1 read and 1 write capacity unit.
2. **RDS Database Instance :**
 - **Engine:** MariaDB
 - **Instance Class:** db.t4g.micro
 - **DBInstanceIdentifier:** section1-activity-db
 - **Storage:** 8 GB (General Purpose SSD)
 - **DeletionPolicy:** Delete on stack deletion

Activity

1. Find the template files in our GitHub repository under the same name as the heading for easy access and edits. Find and Save the attached template locally, open it in VS Code for edits.
2. Go to the AWS CloudFormation Console and create a new stack by uploading the template. Name the stack as you like and wait for the stack creation to complete successfully.

CloudFormation > Stacks > Create stack

CloudFormation

- Stacks
- StackSets
- Exports
- Infrastructure Composer
- laC generator
- Hooks overview [New](#)
- Hooks [New](#)
- Registry
 - Public extensions
 - Activated extensions
 - Publisher
- Spotlight

Step 1: Create stack

Step 2: Specify stack details

Step 3: Configure stack options

Step 4: Review and create

Create stack

Prerequisite - Prepare template

You can also create a template by scanning your existing resources in the [laC generator](#).

Prepare template

Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

☒ Choose an existing template
Upload or choose an existing template.
☐ Build from Infrastructure Composer
Create a template using a visual builder.

Specify template [Info](#)

A template is a JSON or YAML file that describes your stack's resources and properties.

Template source

Selecting a template generates an Amazon S3 URL where it will be stored.

☐ Amazon S3 URL
Provide an Amazon S3 URL to your template.
☒ Upload a template file
Upload your template directly to the console.
☐ Sync from Git
Sync a template from your Git repository.

Upload a template file

[Choose file](#)

section1-activity-template.yaml

JSON or YAML formatted file

- Check the service consoles to verify the creation of an Amazon DynamoDB table and an Amazon RDS MariaDB instance.

stack1

[Delete](#)
[Update](#)
[Stack actions](#)
[Create stack](#)

[Stack info](#)
[Events - updated](#)
[Resources](#)
[Outputs](#)
[Parameters](#)
[Template](#)

[Table view](#)
[Timeline view - new](#)

Events (6) [Detect root cause](#)

Timestamp	Logical ID	Status	Detailed status
2024-12-02 11:01:55 UTC+0530	SampleTable	CREATE_COMPLETE	-

- Delete the stack after verifying the resources and wait until it reaches the 'DELETE_COMPLETE' state. Confirm the resources are removed from their service consoles.

stack1

[Delete](#)
[Update](#)
[Stack actions](#)
[Create stack](#)

[Stack info](#)
[Events - updated](#)
[Resources](#)
[Outputs](#)
[Parameters](#)
[Template](#)

[Table view](#)
[Timeline view - new](#)

Events (11) [Detect root cause](#)

Timestamp	Logical ID	Status	Detailed status
2024-12-02 11:03:11 UTC+0530	SampleTable	DELETE_COMPLETE	-