## Problem Statement:

You work for XYZ Corporation. Your team is asked to deploy similar architecture multiple times for testing, development, and production purposes. Implement CloudFormation for the tasks assigned to you below.

## Tasks To Be Performed:

1. Create a template which can create an S3 bucket named "Intellipaat-<yourname >"
2. The template should be able to enable versioning for the bucket created.

aws | Services | Search [Option+S]

N. Virginia ▼ | rsujithsri16@gmail.com ▼

EC2

## CloudFormation ✕

CloudFormation > Stacks

**Stacks**

**StackSets**

**Exports**

Application Composer **New**

IaC generator

▼ **Registry**

Public extensions

Activated extensions

Publisher

Spotlight

**Feedback**

### Stacks (0)

🗘 | Delete | Update | Stack actions ▼ | **Create stack** ▼

Filter status

🔍 Filter by stack name

Active ▼ | 🔵 View nested

‹ **1** › ⚙

| Stack name | Status | Created time ▼ | Description |
|---|---|---|---|

**No stacks**

No stacks to display

**Create stack**

**View getting started guide**

☰ CloudFormation > Stacks > Create stack ◎

**Step 1**
**Create stack**

**Step 2**
Specify stack details

**Step 3**
Configure stack options

**Step 4**
Review and create

# Create stack

## Prerequisite – Prepare template

### 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.

⭘ **Use a sample template**
Choose from our sample template library.

🔘 **Build from Application Composer**
Create a template using a visual builder.

## Create a template in Application Composer

Use Application Composer to visually design your stacks on a simple, drag-and-drop interface. Application Composer automatically updates and validates the template.

**Create in Application Composer**

Cancel    **Next**

**aws** ::: Services | Q Search | [Option+S]

N. Virginia ▼ | rsujithsri16@gmail.com ▼

EC2

CloudFormation > Stacks > Create stack

Step 1
**Create stack**

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review and create

# Create stack

## Prerequisite – Prepare template

### 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.

- ○ **Use a sample template**
  Choose from our sample template library.

- ○ **Build from Application Composer**
  Create a template using a visual builder.

## Specify template
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 - *new***
  Sync a template from your Git repository.

### Upload a template file

◉ **Choose an existing template**
Upload or choose an existing template.

○ **Use a sample template**
Choose from our sample template library.

○ **Build from Application Composer**
Create a template using a visual builder.

Step 4
Review and create

## Specify template

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 - *new***
Sync a template from your Git repository.

### Upload a template file

⬆ Choose file

| my new.template | ✕ |

JSON or YAML formatted file

S3 URL:  https://s3.us-east-1.amazonaws.com/cf-templates-15tthmkgl61g-us-east-1/2024-04-11T033704.805Z47n-mynew.template

View in Application Composer

Cancel    **Next**

**aws** ::: Services  Q *Search*  [Option+S]  N. Virginia ▾  rsujithsri16@gmail.com ▾

EC2

CloudFormation > Stacks > Create stack

Step 1
**Create stack**

Step 2
**Specify stack details**

Step 3
Configure stack options

Step 4
Review and create

# Specify stack details

## Provide a stack name

Stack name

```
mystack
```

Stack name can include letters (A–Z and a–z), numbers (0–9) and dashes (-).

## Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

### No parameters

There are no parameters defined in your template

Cancel      Previous      **Next**

aws | ▦ Services | Q Search | [Option+S] | ⌕ | ⌂ | ⑦ | ⚙ | N. Virginia ▾ | rsujithsri16@gmail.com ▾

⬚ EC2

**Step 2**

**Specify stack details**

**Step 3**

**Configure stack options**

**Step 4**

Review and create

## Tags

You can specify tags (key-value pairs) to apply to resources in your stack. You can add up to 50 unique tags for each stack.

No tags associated with the stack.

[ Add new tag ]

You can add 50 more tag(s)

## Permissions

IAM role – optional

Choose the IAM role for CloudFormation to use for all operations performed on the stack.

| IAM role name                ▼ | Sample-role-name                                ▼ |   [ Remove ]

[ ↻ ]

## Stack failure options

Behaviour on provisioning failure

Specify the roll-back behaviour for a stack failure. Learn more ↗

🔘 Roll back all stack resources

## Stack failure options

### Behaviour on provisioning failure

Specify the roll-back behaviour for a stack failure. **Learn more** 🔗

- ● **Roll back all stack resources**
  Roll back the stack to the last known stable state.

- ○ **Preserve successfully provisioned resources**
  Preserves the state of successfully provisioned resources, while rolling back failed resources to the last known stable state.
  Resources without a last known stable state will be deleted upon the next stack operation.

### Delete newly created resources during a rollback

Specify whether resources that were created during a failed operation should be deleted regardless of their deletion policy. **Learn more** 🔗

- ● **Use deletion policy**
  Retains or deletes created resources according to their attached deletion policy.

- ○ **Delete all newly created resources**
  Deletes created resources during a rollback regardless of their attached deletion policy.

## Advanced options

You can set additional options for your stack, like notification options and a stack policy. **Learn more** 🔗

aws

Services | Q Search | [Option+S]

N. Virginia ▾ | rsujithsri16@gmail.com ▾

EC2

Retains or deletes created resources according to their attached deletion policy.

○ **Delete all newly created resources**
Deletes created resources during a rollback regardless of their attached deletion policy.

## Advanced options

You can set additional options for your stack, like notification options and a stack policy. Learn more ⧉

▶ **Stack policy**

Defines the resources that you want to protect from unintentional updates during a stack update.

▶ **Rollback configuration**

Specify alarms for CloudFormation to monitor when creating and updating the stack. If the operation breaches an alarm threshold, CloudFormation rolls it back.

▶ **Notification options**

▶ **Stack creation options**

Cancel | Previous

aws Services Q Search [Option+S]

EC2

CloudFormation > Stacks > Create stack

**Step 1**
Create stack

**Step 2**
Specify stack details

**Step 3**
Configure stack options

**Step 4**
Review and create

# Review and create

## Step 1: Specify template

Edit

### Prerequisite – Prepare template

Template
Template is ready

### Template

Template URL
https://s3-external-1.amazonaws.com/cf-templates-15tthmkgl61g-us-east-1/2024102WoQ-new.templatepz3mjbacnk

Stack description
Simple S3 Bucket Template

## Step 2: Specify stack details

Edit

aws          Services        Q Search                                        [Option+S]                          ⚙ Δ ⓘ ⚙    N. Virginia ▼    rsujithsri16@gmail.com ▼

EC2

☰              Edit          ⟳

# Step 2: Specify stack details                                                    Edit

## Provide a stack name

Stack name
mystack

## Parameters

Q Search                                                                                    ‹  1  ›    ⚙

| Key | ▲ | Value | ▽ |
|---|---|---|---|

**No parameters**

There are no parameters defined in your template

# Step 3: Configure stack options                                                  Edit

## Tags

aws  Services  Q Search  [Option+S]  N. Virginia ▼  rsujithsri16@gmail.com ▼

EC2

## Tags

Q Search tags

< 1 >

| Key ▲ | | Value ▽ |
|---|---|---|

**No tags**

There are no tags defined for this stack

## Permissions

**No permissions**

There is no IAM role associated with this stack

## Stack failure options

Rollback on failure
Activated

Delete newly created resources during a rollback

-

## Notification options

**SNS topic ARN**

**No notification options**

There are no notification options defined

## Stack creation options

Timeout

-

Termination protection

Deactivated

▶ **Quick-create link**

Create changeset                          Cancel    Previous    **Submit**

aws | Services | Q Search | [Option+S] | N. Virginia ▼ | rsujithsri16@gmail.com ▼

EC2

CloudFormation > Stacks > mystack

**⊟ Stacks (1)**

**Filter status**

🔍 Filter by stack name | Active ▼ | 🔵 View nested

‹ 1 ›

**Stacks**

🔵 **mystack**
2024-04-11 15:20:48 UTC+0530
ⓘ CREATE_IN_PROGRESS

**mystack**                                          ⚙ ✕

Delete | Update | Stack actions ▼ | Create stack ▼

‹ | Stack info | **Events** | Resources | Outputs | Parameters | Template | Chang › ›

**Events (3)**                          Detect root cause | ↻

🔍 Search events | ⚙

| Timestamp ▼ | Logical ID | Status | Detailed status |
|---|---|---|---|
| 2024-04-11 15:20:52 UTC+0530 | MyS3Bucket | ⓘ CREATE_IN_PROGRESS | - |
| 2024-04-11 15:20:50 UTC+0530 | MyS3Bucket | ⓘ CREATE_IN_PROGRESS | - |
| 2024-04-11 15:20:48 UTC+0530 | mystack | ⓘ CREATE_IN_PROGRESS | - |

aws | ⠿ Services | 🔍 Search | [Option+S] | ⊡ ⌂ ⑦ ⚙ | N. Virginia ▼ | rsujithsri16@gmail.com ▼

🗗 EC2

☰

CloudFormation > Stacks > mystack

Delete | Update | Stack actions ▼ | Create stack ▼

ⓘ
⊘

⊟ **Stacks (1)** ⟲

| < Stack info | **Events** | Resources | Outputs | Parameters | Template | Ch: > |

**Filter status**

🔍 Filter by stack name | Active ▼ | 🔵 View nested

< 1 >

**Stacks**

**mystack**
⬤ 2024-04-11 15:20:48 UTC+0530
⊘ CREATE_COMPLETE

‖

**Events (5)**

Detect root cause | ⟲

🔍 Search events

⚙

| Timestamp ▼ | Logical ID | Status | Detailed status |
|---|---|---|---|
| 2024-04-11 15:21:15 UTC+0530 | mystack | ⊘ CREATE_COMPLETE | - |
| 2024-04-11 15:21:15 UTC+0530 | MyS3Bucket | ⊘ CREATE_COMPLETE | - |
| 2024-04-11 15:20:52 UTC+0530 | MyS3Bucket | ⓘ CREATE_IN_PROGRESS | - |
| 2024-04-11 15:20:50 UTC+0530 | MyS3Bucket | ⓘ CREATE_IN_PROGRESS | - |
| 2024-04-11 15:20:48 UTC+0530 | mystack | ⓘ CREATE_IN_PROGRESS | - |

aws ⠿ Services Q Search [Option+S] ⤢ ⟳ ⑦ ⚙ N. Virginia ▼ rsujithsri16@gmail.com ▼

EC2

CloudFormation > Stacks > mystack

mystack ⚙ ✕

### Stacks (1) ⟳

Filter status

🔍 Filter by stack name | Active ▼ | 🔵 View nested

< 1 >

**Stacks**

mystack
2024-04-11 15:20:48 UTC+0530
⊘ CREATE_COMPLETE

**mystack**

Delete | Update | Stack actions ▼ | Create stack ▼

< Stack info | Events | **Resources** | Outputs | Parameters | Template | Chan > >

### Resources (1) ⟳

🔍 Search resources

< 1 > ⚙

| Logical ID ▲ | Physical ID ▽ | Type ▽ | Status |
|---|---|---|---|
| MyS3Bucket | my-unique-bucket-name-2024 ↗ | AWS::S3::Bucket | ⊘ CREATE_COMPL |