Creating an S3 bucket using Terraform is a common task for automating infrastructure management on AWS. Below is a step-by-step tutorial to guide you through the process. Since I can't directly create images, I'll describe the steps with corresponding image suggestions. You can create the images based on these descriptions.

Prerequisites

- \*\*Terraform Installed:\*\* Ensure that you have Terraform installed on your machine. [Installation guide](https://learn.hashicorp.com/tutorials/terraform/install-cli).

- \*\*AWS Account:\*\* You need access to an AWS account.

- \*\*AWS CLI Configured:\*\* Make sure your AWS CLI is configured with your credentials.

Step 1: Set Up Your Project Directory

1. \*\*Create a Directory for Your Terraform Project\*\*

- Open a terminal and create a directory for your Terraform project:

```bash

mkdir terraform-s3-bucket

cd terraform-s3-bucket

```

- \*Image Suggestion:\* Screenshot of a terminal with the above commands being executed.

2. \*\*Create a `main.tf` file\*\*

- In your project directory, create a `main.tf` file:

```bash

touch main.tf

```

- \*Image Suggestion:\* File explorer view showing an empty `main.tf` file.

Step 2: Configure the AWS Provider

1. \*\*Define the AWS Provider\*\*

- Open the `main.tf` file and add the following configuration:

```hcl

provider "aws" {

region = "us-west-2"

}

```

- This tells Terraform to use the AWS provider and specifies the region where resources will be created.

- \*Image Suggestion:\* Code editor view showing the provider configuration.

Step 3: Define the S3 Bucket Resource

1. \*\*Add S3 Bucket Resource\*\*

- In the `main.tf` file, add the following resource block to create an S3 bucket:

provider "aws" {

region = var.region

}

resource "aws\_s3\_bucket" "S3Demo" {

bucket = "my-aug17-test-bucket"

tags = {

Name = var.aws\_tags[1]

Environment = var.aws\_tags[0]

}

}

- Replace `"my-unique-bucket-name"` with a unique bucket name.

- \*Image Suggestion:\* Code editor view showing the S3 bucket resource configuration.

variables.tf

variable "region" {

description = "Region"

default = "us-west-2"

}

variable "aws\_tags" {

description = "tags"

type = list(string)

default = ["Dev", "My Test bucket", "S3Demo"]

}

Step 4: Initialize Terraform

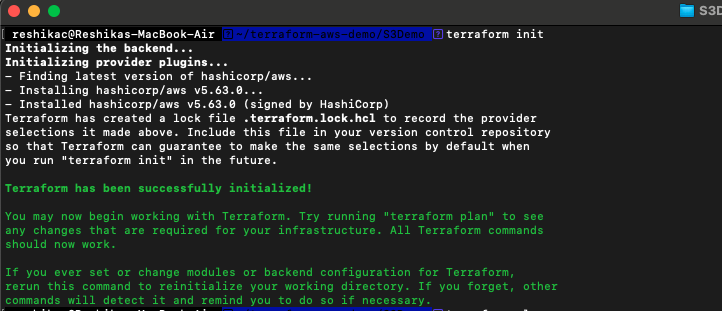
1. \*\*Initialize the Terraform Project\*\*

- In the terminal, run the following command to initialize Terraform:

```bash

terraform init

```



- This command downloads the necessary provider plugins.

- \*Image Suggestion:\* Terminal screenshot showing the output of `terraform init`.

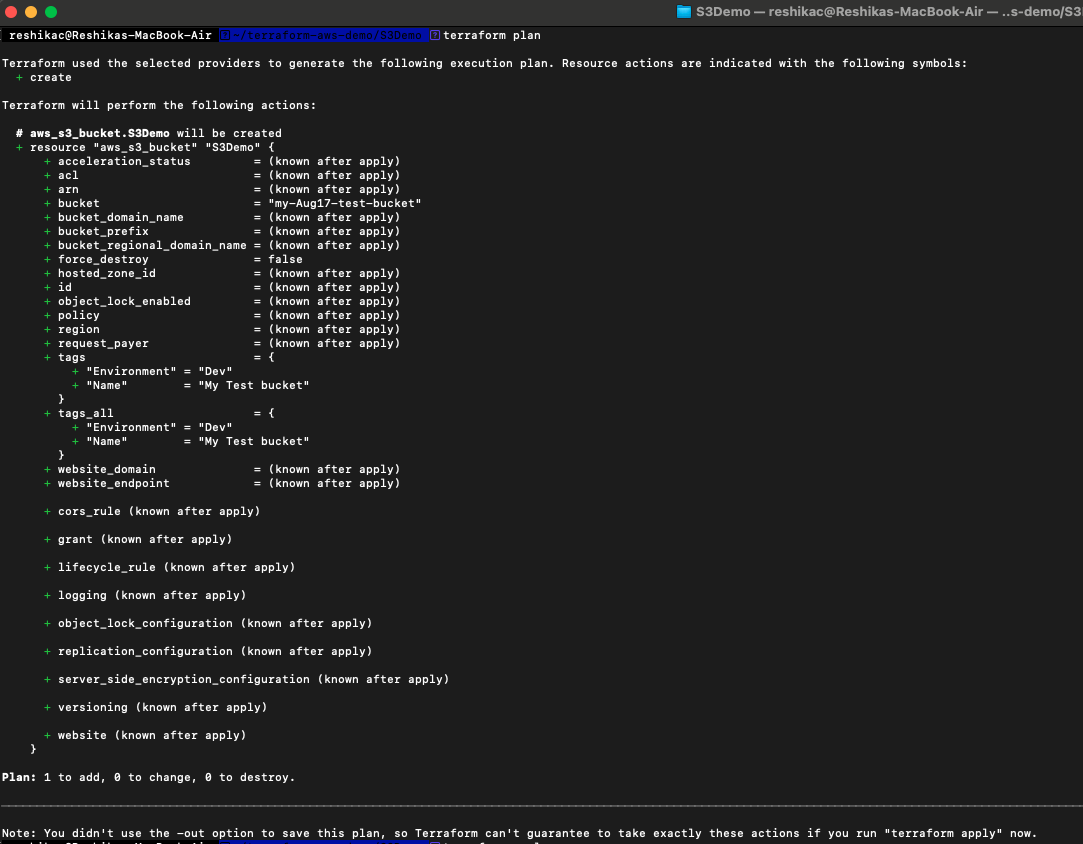
Step 5: Review and Apply the Configuration

1. \*\*Review the Execution Plan\*\*

- To review the changes Terraform will make, run:

```bash

terraform plan



```

- This will show you an execution plan, detailing the resources Terraform will create.

- \*Image Suggestion:\* Terminal screenshot showing the output of `terraform plan`, highlighting the new S3 bucket.

2. \*\*Apply the Configuration\*\*

- To create the S3 bucket, run:

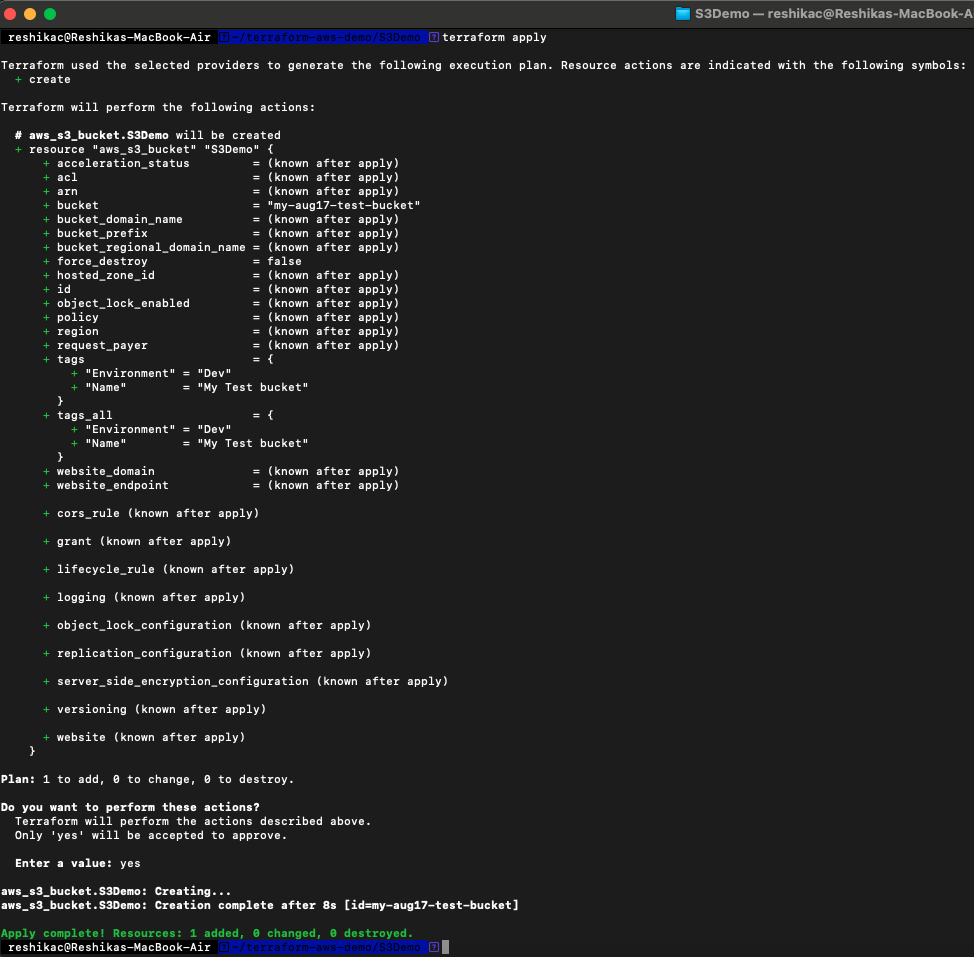
```bash

terraform apply

```

- Confirm the action by typing `yes` when prompted.

- \*Image Suggestion:\* Terminal screenshot showing the output of `terraform apply`, confirming the S3 bucket creation.

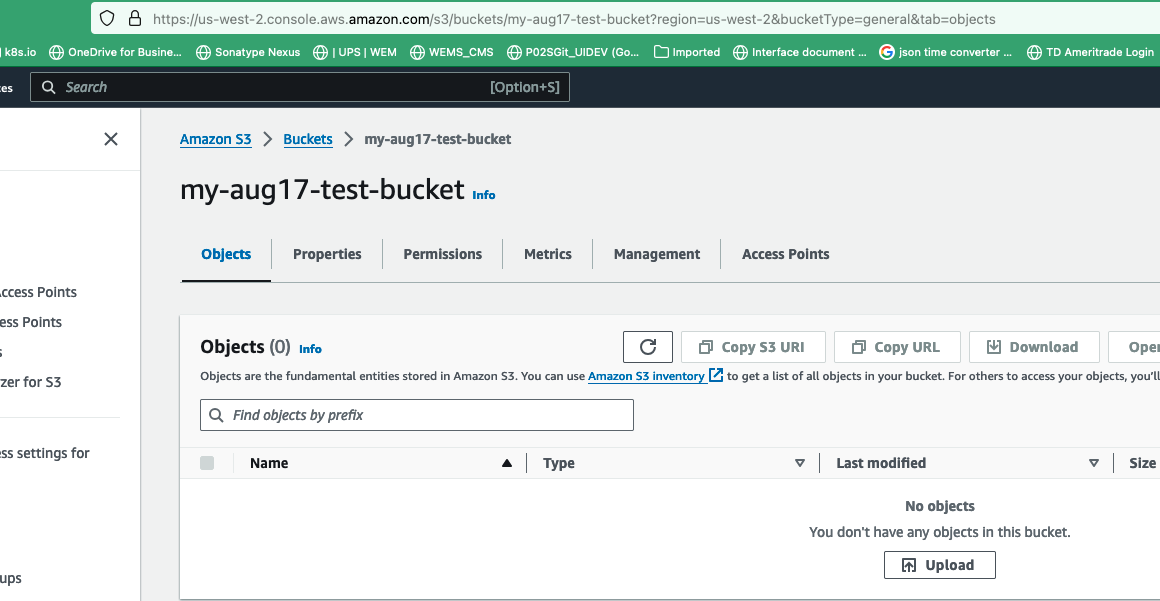


Step 6: Verify the S3 Bucket in AWS Console

1. \*\*Log in to AWS Management Console\*\*

- Navigate to the S3 service and verify that your new bucket has been created.

- \*Image Suggestion:\* Screenshot of the AWS S3 console showing the newly created bucket.



Step 7: Clean Up Resources (Optional)

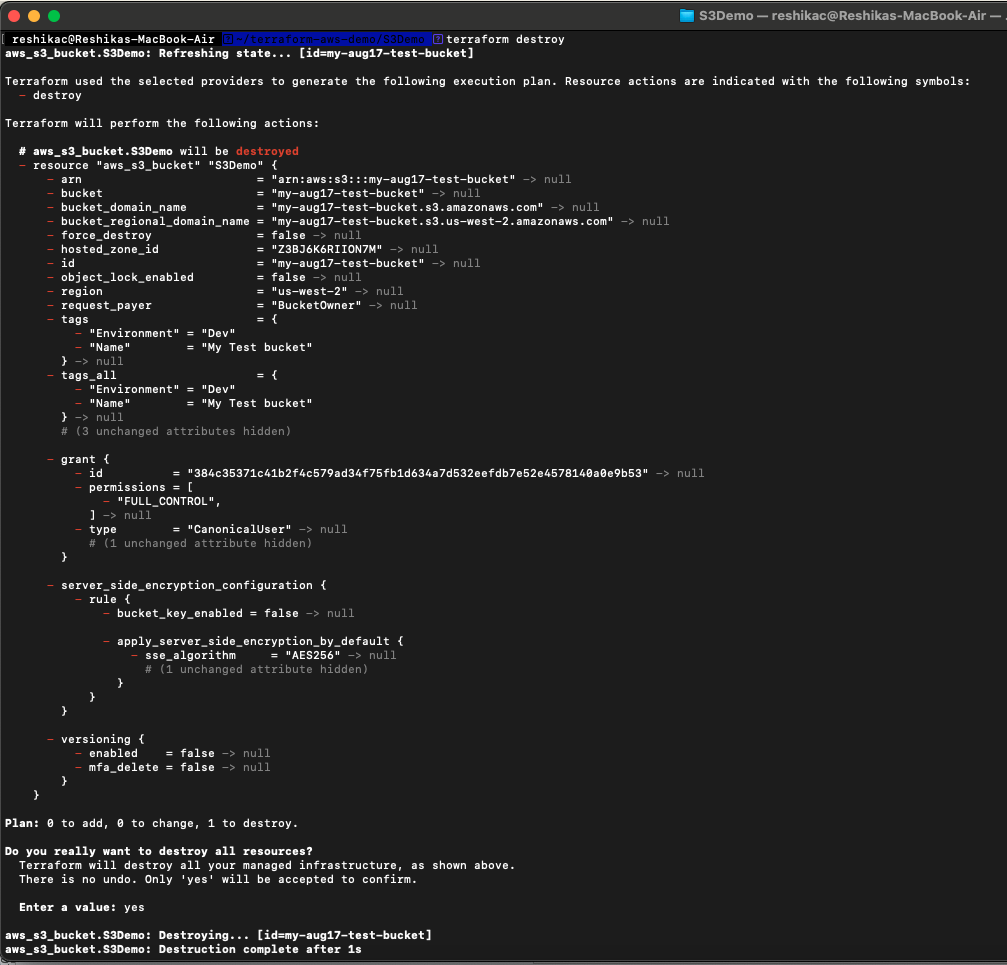
1. \*\*Destroy the S3 Bucket\*\*

- If you want to remove the S3 bucket and other resources, run:

```bash

terraform destroy

```



- Confirm the action by typing `yes`.

- \*Image Suggestion:\* Terminal screenshot showing the output of `terraform destroy`, indicating resource destruction.

Conclusion

This tutorial covers the basics of creating an S3 bucket using Terraform. For more advanced usage, you can explore Terraform modules, state management, and remote backends.

If you want to generate images for this tutorial, use a code editor like Visual Studio Code for screenshots, and the AWS Management Console for verifying your results.