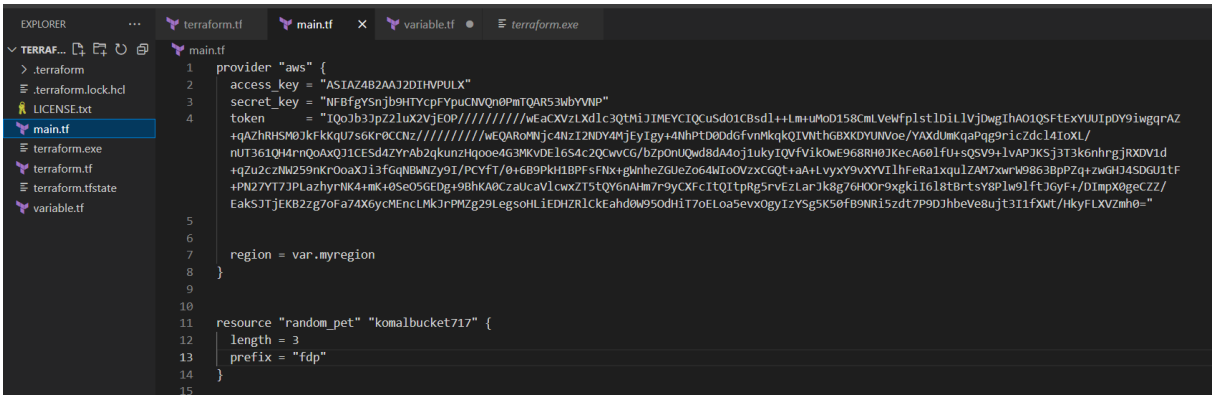
Set Up Your Terraform Environment

- **Install Terraform**: If you haven't already installed Terraform, download it from Terraform's official site and follow the installation steps.
- **Configure AWS CLI**: Make sure the AWS CLI is installed and configured with the necessary permissions.

Create a Terraform Project

- Create a directory for your project and navigate to it



Define the Provider

- In `main.tf`, start by defining the provider (AWS)

terraform.tf main.tf variable.tf terraform.exe

terraform.tf

```
terraform {
  required_providers {
    aws = {
      source = "hashicorp/aws"
    }
    random = {
      source = "hashicorp/random"
    }
    archive = {
      source = "hashicorp/archive"
    }
  }
}
```

terraform.tf main.tf variable.tf terraform.exe

variable.tf

```
variable "myregion" {
  type = string
  default = "us-east-1"
}
```

Create an S3 Bucket

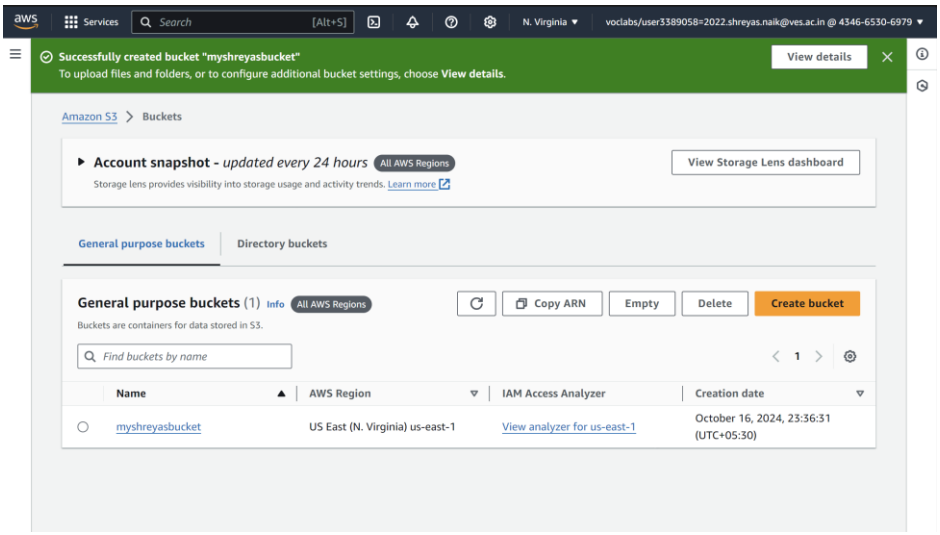
- Next, define an S3 bucket where Lambda can receive file uploads.

```
C:\Terraform>set AWS_ACCESS_KEY_ID=ASIAZ4B2AAJ2DIHVPULX

C:\Terraform>set AWS_SECRET_ACCESS_KEY=NFBfgYSnjb9HTYcpFYpuCNVQn0PmTQAR53WbYVNP

C:\Terraform>terraform -v
Terraform v1.9.4
on windows_amd64

C:\Terraform>terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/archive...
- Finding latest version of hashicorp/aws...
- Finding latest version of hashicorp/random...
- Installing hashicorp/archive v2.5.0...
- Installed hashicorp/archive v2.5.0 (signed by HashiCorp)
- Installing hashicorp/aws v5.62.0...
- Installed hashicorp/aws v5.62.0 (signed by HashiCorp)
- Installing hashicorp/random v3.6.2...
- Installed hashicorp/random v3.6.2 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

```
C:\Terraform>terraform plan

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # random_pet.komalbucket717 will be created
  + resource "random_pet" "komalbucket717" {
      + id        = (known after apply)
      + length    = 3
      + prefix    = "fdp"
      + separator = "-"
    }

Plan: 1 to add, 0 to change, 0 to destroy.

---

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.

C:\Terraform>terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # random_pet.komalbucket717 will be created
  + resource "random_pet" "komalbucket717" {
      + id        = (known after apply)
      + length    = 3
      + prefix    = "fdp"
      + separator = "-"
    }

Plan: 1 to add, 0 to change, 0 to destroy.
Terraform will perform the following actions:

  # random_pet.komalbucket717 will be created
```

```
C:\Terraform>terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # random_pet.komalbucket717 will be created
  + resource "random_pet" "komalbucket717" {
      + id        = (known after apply)
      + length    = 3
      + prefix    = "fdp"
      + separator = "-"
    }

Plan: 1 to add, 0 to change, 0 to destroy.
Terraform will perform the following actions:

  # random_pet.komalbucket717 will be created
  + resource "random_pet" "komalbucket717" {
      + id        = (known after apply)
      + length    = 3
      + prefix    = "fdp"
      + separator = "-"
    }

Plan: 1 to add, 0 to change, 0 to destroy.
  + resource "random_pet" "komalbucket717" {
      + id        = (known after apply)
      + length    = 3
      + prefix    = "fdp"
      + separator = "-"
    }
```

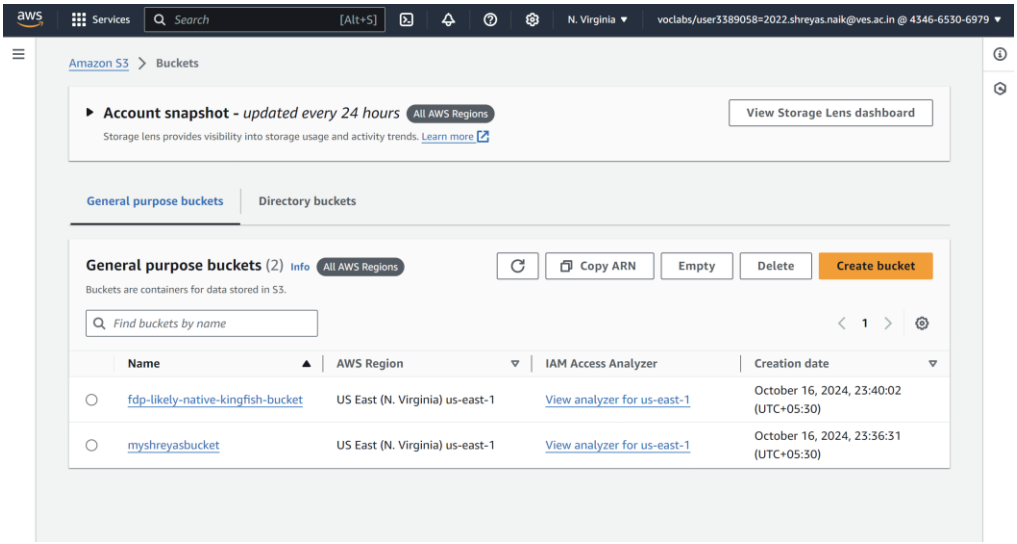
```
    # random_pet.komalbucket717 will be created
    + resource "random_pet" "komalbucket717" {
        + id        = (known after apply)
        + length    = 3
        + prefix    = "fdp"
        + separator = "-"
      }

  Plan: 1 to add, 0 to change, 0 to destroy.
    + resource "random_pet" "komalbucket717" {
        + id        = (known after apply)
        + length    = 3
        + prefix    = "fdp"
        + separator = "-"
      }

  Plan: 1 to add, 0 to change, 0 to destroy.
        + prefix    = "fdp"
        + separator = "-"
      }

  Plan: 1 to add, 0 to change, 0 to destroy.
      }

  Plan: 1 to add, 0 to change, 0 to destroy.
  Plan: 1 to add, 0 to change, 0 to destroy.

  Do you want to perform these actions?
    Terraform will perform the actions described above.
    Terraform will perform the actions described above.
    Only 'yes' will be accepted to approve.

    Enter a value: yes

  random_pet.komalbucket717: Creating...
  random_pet.komalbucket717: Creation complete after 0s [id=fdp-likely-native-kingfish]

  Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
```




```
resource "aws_s3_bucket" "shreyasbucket" {
  bucket = "${random_pet.shreyasbucket.id}-bucket"
  acl    = "private"
}

output "s3_arn" {
  value       = aws_s3_bucket.shreyasbucket.arn
  description = "The ARN of the S3 bucket"
}

variable "myregion" {
  description = "The AWS region to deploy resources in"
  default     = "us-east-1"  # Change this to your desired region if necessary
}
```

```
C:\Terraform>terraform apply
random_pet.komalbucket717: Refreshing state... [id=fdp-likely-native-kingfish]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbol
  + create

Terraform will perform the following actions:

  # aws_s3_bucket.komalbucket717 will be created
  + resource "aws_s3_bucket" "komalbucket717" {
      + acceleration_status         = (known after apply)
      + acl                         = "private"
      + arn                         = (known after apply)
      + bucket                      = "fdp-likely-native-kingfish-bucket"
      + bucket_domain_name          = (known after apply)
      + bucket_prefix               = (known after apply)
      + bucket_regional_domain_name = (known after apply)
      + force_destroy               = false
      + hosted_zone_id              = (known after apply)
      + id                          = (known after apply)
      + object_lock_enabled         = (known after apply)
      + policy                      = (known after apply)
      + region                      = (known after apply)
      + request_payer               = (known after apply)
      + tags_all                    = (known after apply)
      + website_domain              = (known after apply)
      + website_endpoint            = (known after apply)

      + cors_rule (known after apply)

      + grant (known after apply)

      + lifecycle_rule (known after apply)
```

```
aws_s3_bucket.komalbucket717: Creating...
aws_s3_bucket.komalbucket717: Creation complete after 5s [id=fdp-likely-native-kingfish-bucket]

Warning: Argument is deprecated

  Enter a value: yes

aws_s3_bucket.komalbucket717: Creating...
aws_s3_bucket.komalbucket717: Creation complete after 5s [id=fdp-likely-native-kingfish-bucket]

Warning: Argument is deprecated


Warning: Argument is deprecated

  with aws_s3_bucket.komalbucket717,
  on main.tf line 16, in resource "aws_s3_bucket" "komalbucket717":
  16:   acl    = "private"

  with aws_s3_bucket.komalbucket717,
  on main.tf line 16, in resource "aws_s3_bucket" "komalbucket717":
  16:   acl    = "private"


Use the aws_s3_bucket_acl resource instead
Use the aws_s3_bucket_acl resource instead


Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

Outputs:

s3_arn = "arn:aws:s3:::fdp-likely-native-kingfish-bucket"

s3_arn = "arn:aws:s3:::fdp-likely-native-kingfish-bucket"
s3_arn = "arn:aws:s3:::fdp-likely-native-kingfish-bucket"
```



Create an SQS Queue

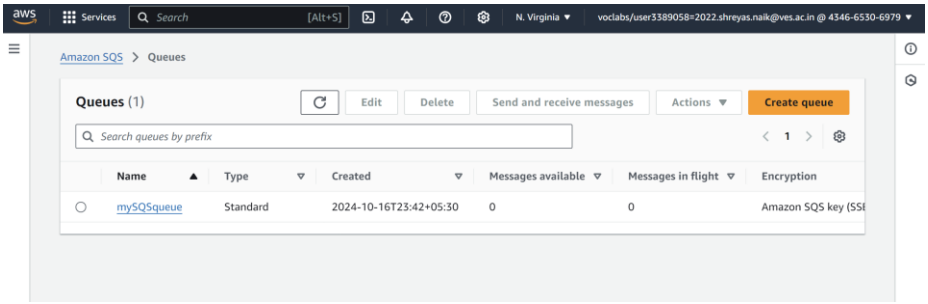
- Add an SQS queue that will receive events from the S3 bucket when a new object is uploaded.

```
resource "aws_sqs_queue" "myqueue" {
  name = "mySQSqueue"
}
```

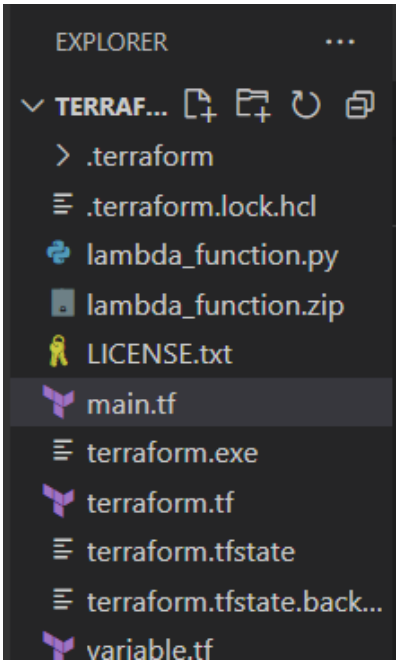
```
C:\Terraform>terraform plan
random_pet.komalbucket717: Refreshing state... [id=fdp-likely-native-kingfish]
aws_s3_bucket.komalbucket717: Refreshing state... [id=fdp-likely-native-kingfish-bucket]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # aws_sqs_queue.myqueue will be created
  + resource "aws_sqs_queue" "myqueue" {
      + arn                               = (known after apply)
      + content_based_deduplication       = false
      + deduplication_scope               = (known after apply)
      + delay_seconds                     = 0
      + fifo_queue                        = false
      + fifo_throughput_limit             = (known after apply)
      + id                                = (known after apply)
      + kms_data_key_reuse_period_seconds = (known after apply)
      + max_message_size                  = 262144
      + message_retention_seconds         = 345600
      + name                              = "mySQSqueue"
      + name_prefix                       = (known after apply)
      + policy                            = (known after apply)
      + receive_wait_time_seconds         = 0
      + redrive_allow_policy              = (known after apply)
      + redrive_policy                    = (known after apply)
      + sqs_managed_sse_enabled           = (known after apply)
      + tags_all                          = (known after apply)
      + url                               = (known after apply)
      + visibility_timeout_seconds        = 30
    }

Plan: 1 to add, 0 to change, 0 to destroy.
```

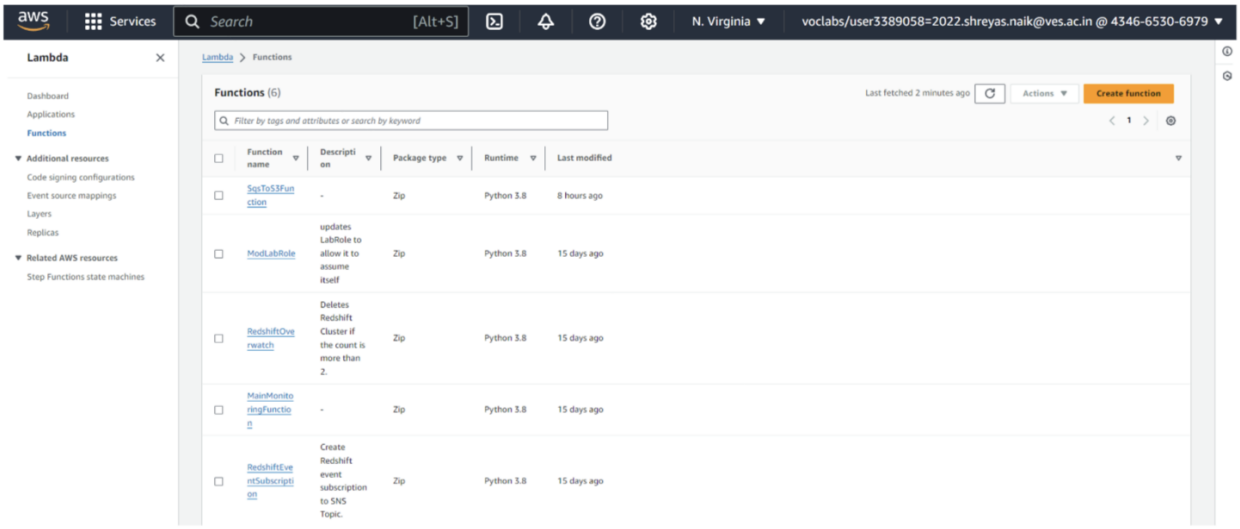
```
C:\Terraform>terraform apply
random_pet.komalbucket717: Refreshing state... [id=fdp-likely-native-kingfish]
aws_s3_bucket.komalbucket717: Refreshing state... [id=fdp-likely-native-kingfish-bucket]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # aws_sqs_queue.myqueue will be created
  + resource "aws_sqs_queue" "myqueue" {
      + arn                               = (known after apply)
      + content_based_deduplication       = false
      + deduplication_scope               = (known after apply)
      + delay_seconds                     = 0
      + fifo_queue                        = false
      + fifo_throughput_limit             = (known after apply)
      + id                                = (known after apply)
      + kms_data_key_reuse_period_seconds = (known after apply)
      + max_message_size                  = 262144
      + message_retention_seconds         = 345600
      + name                              = "mySQSqueue"
      + name_prefix                       = (known after apply)
      + policy                            = (known after apply)
      + receive_wait_time_seconds         = 0
      + redrive_allow_policy              = (known after apply)
      + redrive_policy                    = (known after apply)
      + sqs_managed_sse_enabled           = (known after apply)
      + tags_all                          = (known after apply)
      + url                               = (known after apply)
      + visibility_timeout_seconds        = 30
    }

Plan: 1 to add, 0 to change, 0 to destroy.

  Warning: Argument is deprecated

  with aws_s3_bucket.komalbucket717
```

```
Do you want to perform these actions?
  Terraform will perform the actions described above.

Do you want to perform these actions?
  Terraform will perform the actions described above.
Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_sqs_queue.myqueue: Creating...

  Enter a value: yes

aws_sqs_queue.myqueue: Creating...

aws_sqs_queue.myqueue: Creating...
aws_sqs_queue.myqueue: Creating...
aws_sqs_queue.myqueue: Still creating... [10s elapsed]
aws_sqs_queue.myqueue: Still creating... [20s elapsed]
aws_sqs_queue.myqueue: Creation complete after 28s [id=https://sqs.us-east-1.amazonaws.com/678726468212/mySQSqueue]

aws_sqs_queue.myqueue: Still creating... [20s elapsed]
aws_sqs_queue.myqueue: Creation complete after 28s [id=https://sqs.us-east-1.amazonaws.com/678726468212/mySQSqueue]

aws_sqs_queue.myqueue: Creation complete after 28s [id=https://sqs.us-east-1.amazonaws.com/678726468212/mySQSqueue]


Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

Outputs:

s3_arn = "arn:aws:s3:::fdp-likely-native-kingfish-bucket"
```




```
C:\Terraform>terraform apply
random_pet.komalbucket717: Refreshing state... [id=fdp-likely-native-kingfish]
data.archive_file.zip: Reading...
data.archive_file.zip: Read complete after 0s [id=93c92209eafac774599673c33c7e7636e68e60e8]
aws_sqs_queue.myqueue: Refreshing state... [id=https://sqs.us-east-1.amazonaws.com/678726468212/mySQSqueue]
aws_s3_bucket.komalbucket717: Refreshing state... [id=fdp-likely-native-kingfish-bucket]

No changes. Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no differences, so no changes are needed.

  Warning: Argument is deprecated

    with aws_s3_bucket.komalbucket717,
    on main.tf line 16, in resource "aws_s3_bucket" "komalbucket717":
    16:   acl    = "private"

  Use the aws_s3_bucket_acl resource instead

  (and one more similar warning elsewhere)


Apply complete! Resources: 0 added, 0 changed, 0 destroyed.
```



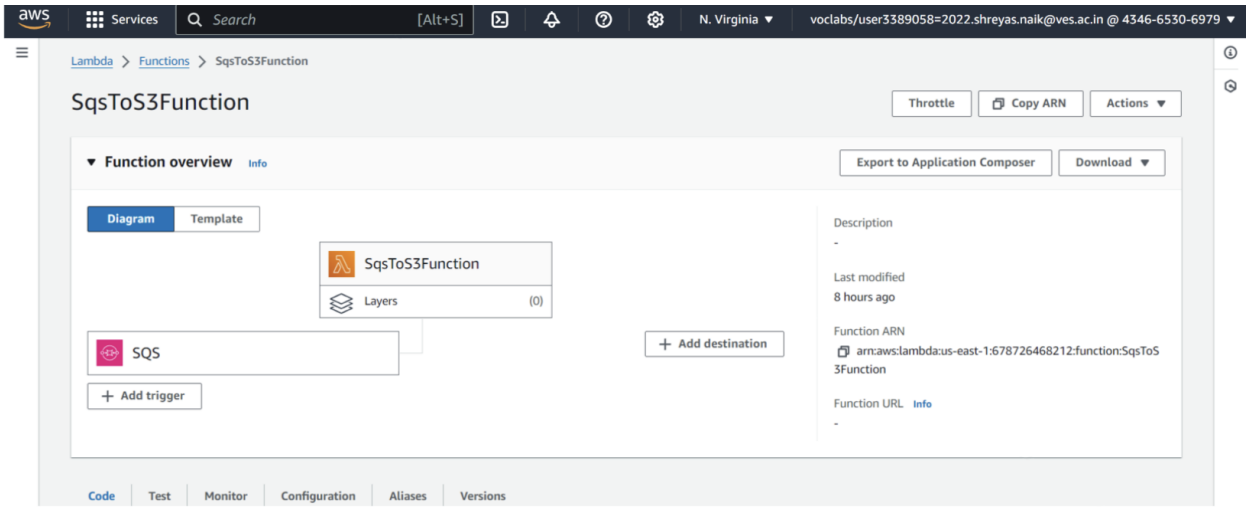


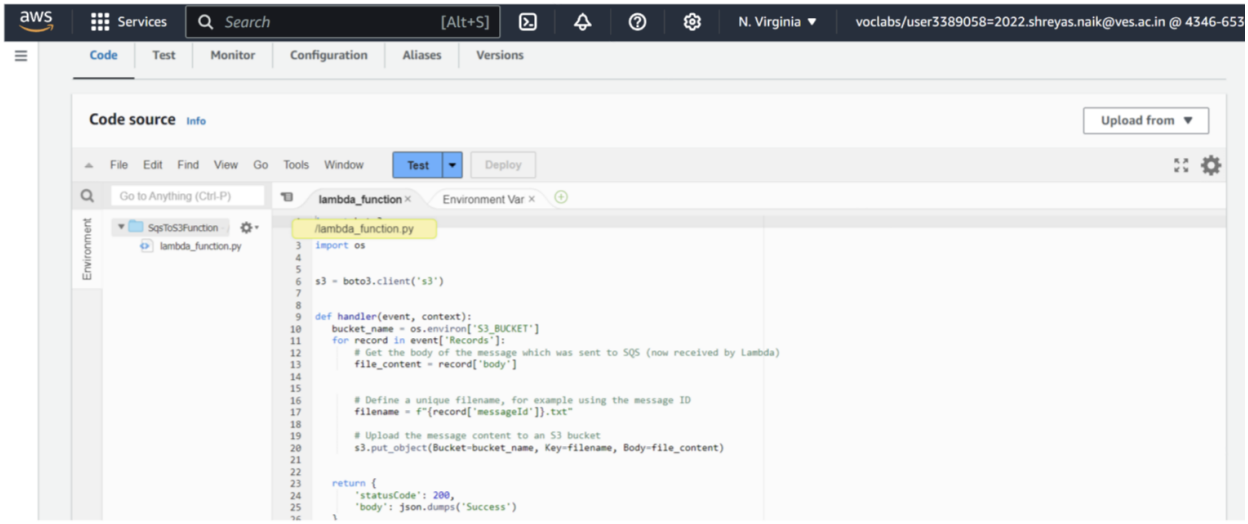
Create a Lambda Function

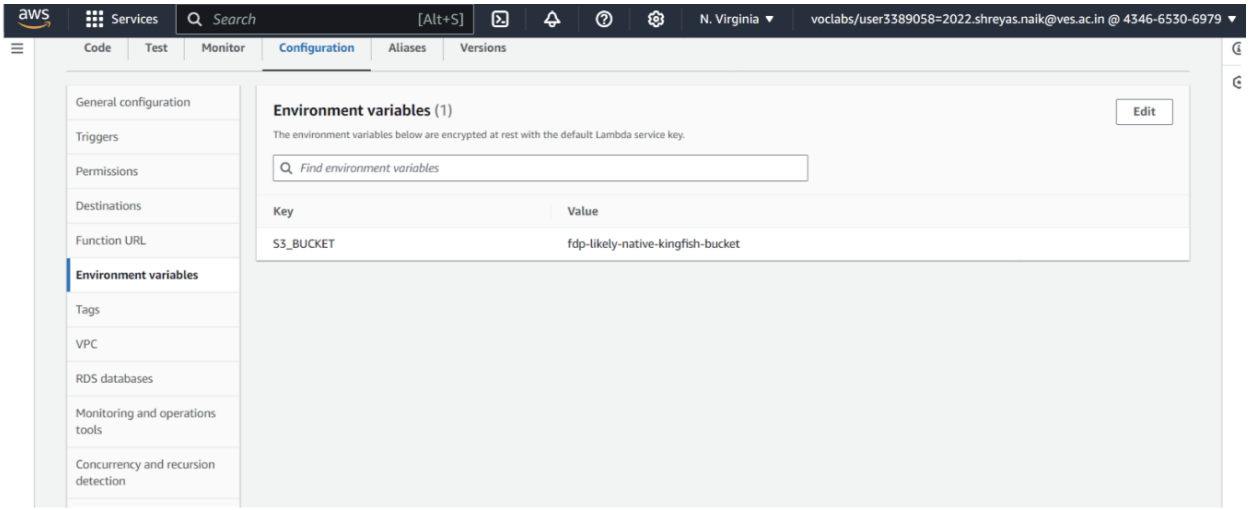
- Define the Lambda function that will be triggered by S3 events and send messages to SQS. You need a ZIP file containing your Lambda function code, which you can upload to the S3 bucket.

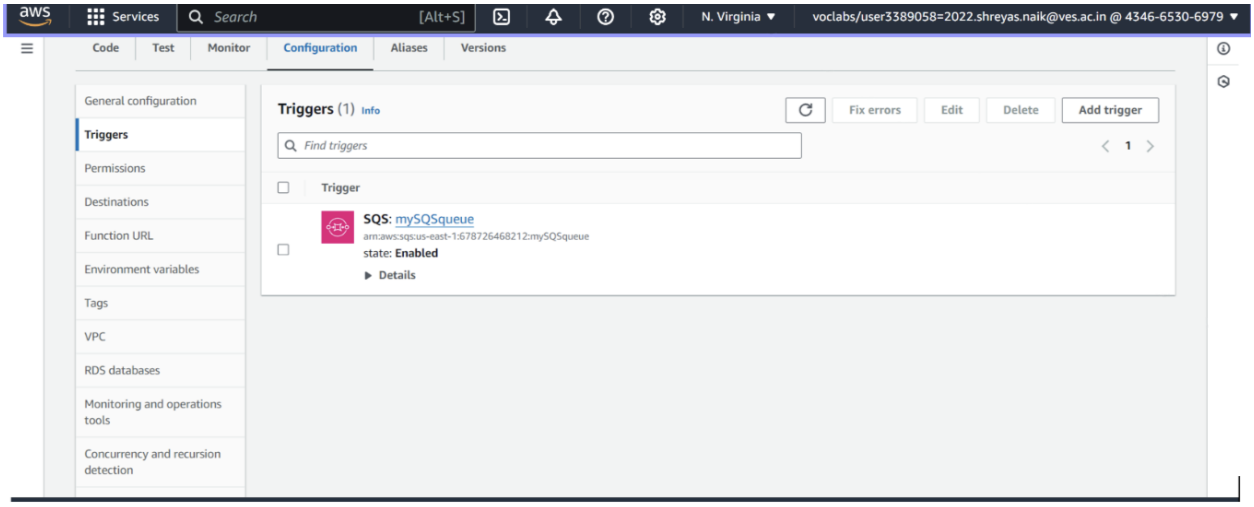
Create IAM Role for Lambda

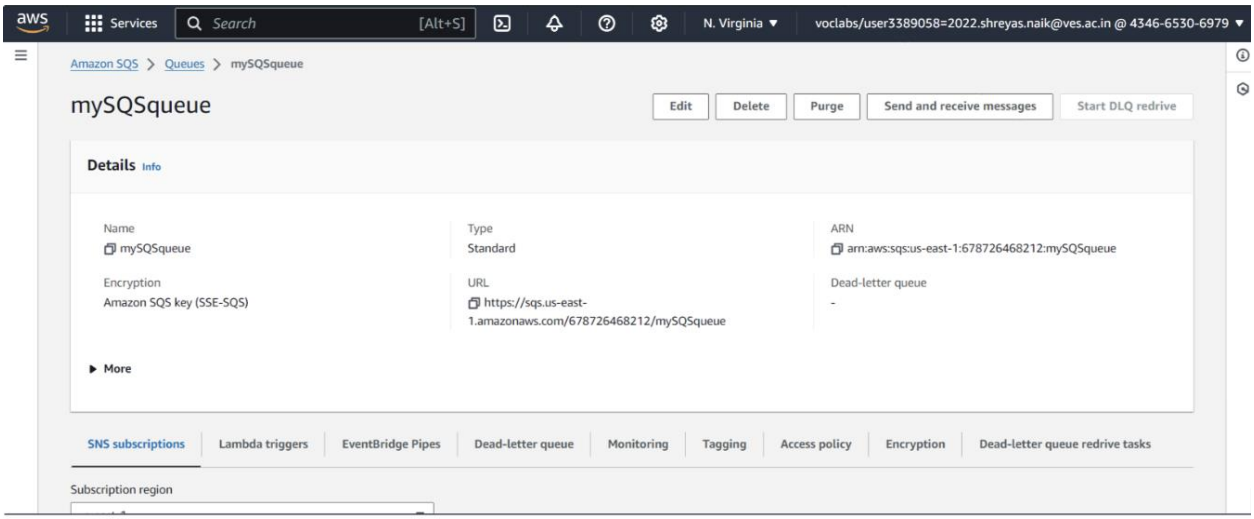
- Lambda needs permissions to read from S3 and send messages to SQS. Define an IAM role and policy for the Lambda function.

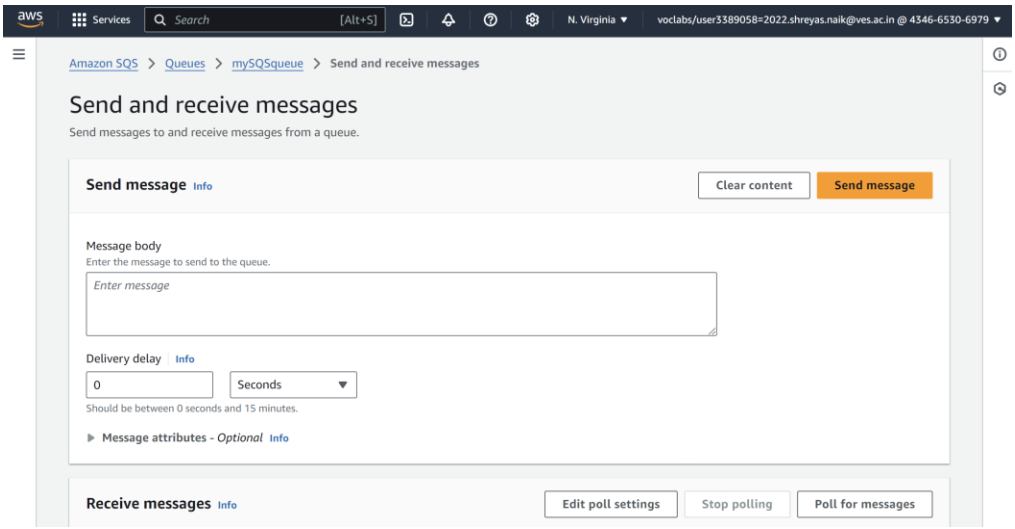

Services
Search
[Alt+S]
N. Virginia
voclabs/user3389058=2022.shreyas.naik@ves.ac.in @ 4346-6530-6979
Lambda
Functions
SqsToS3Function
SqsToS3Function
Throttle
Copy ARN
Actions
Function overview
Info
Export to Application Composer
Download
Diagram
Template
SqsToS3Function
Layers
(0)
SQS
Add trigger
Add destination
Description
-
Last modified
8 hours ago
Function ARN
arn:aws:lambda:us-east-1:678726468212:function:SqsToS3Function
Function URL
Info
-
Code
Test
Monitor
Configuration
Aliases
Versions

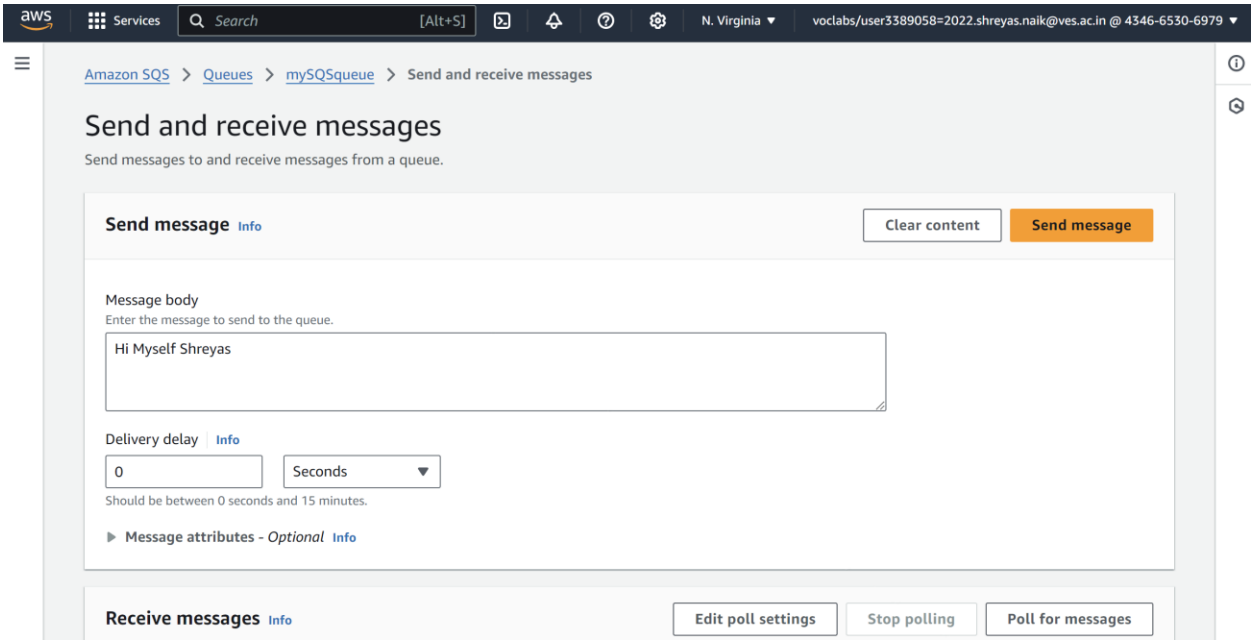

Services
Search
[Alt+S]
N. Virginia
voclabs/user3389058=2022.shreyas.naik@ves.ac.in @ 4346-653
Code
Test
Monitor
Configuration
Aliases
Versions
Code source
Info
Upload from
File
Edit
Find
View
Go
Tools
Window
Test
Deploy
Go to Anything (Ctrl-P)
lambda_function
Environment Var
Environment
SqsToS3Function
lambda_function.py
/lambda_function.py
import os
s3 = boto3.client('s3')
def handler(event, context):
bucket_name = os.environ['S3_BUCKET']
for record in event['Records']:
# Get the body of the message which was sent to SQS (now received by Lambda)
file_content = record['body']
# Define a unique filename, for example using the message ID
filename = f"{record['messageId']}.txt"
# Upload the message content to an S3 bucket
s3.put_object(Bucket=bucket_name, Key=filename, Body=file_content)
return {
'statusCode': 200,
'body': json.dumps('Success')

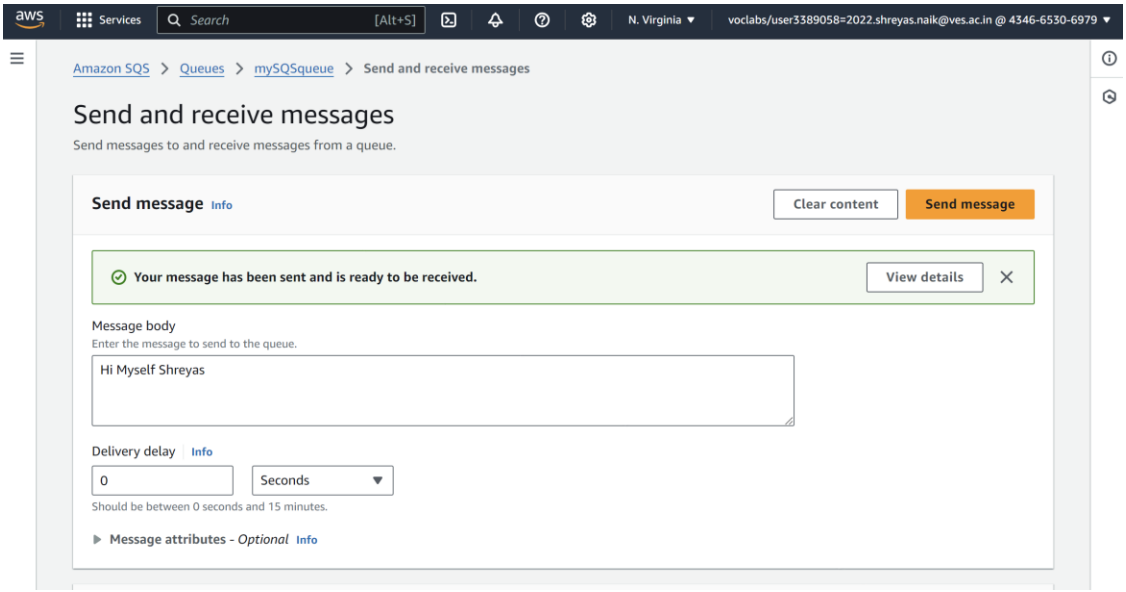

aws
Services
Search
[Alt+S]
N. Virginia
voclabs/user3389058=2022.shreyas.naik@ves.ac.in @ 4346-6530-6979
Code
Test
Monitor
Configuration
Aliases
Versions
General configuration
Triggers
Permissions
Destinations
Function URL
Environment variables
Tags
VPC
RDS databases
Monitoring and operations tools
Concurrency and recursion detection
Environment variables (1)
Edit
The environment variables below are encrypted at rest with the default Lambda service key.
Find environment variables
Key
Value
S3_BUCKET
fdp-likely-native-kingfish-bucket


aws
Services
Search
[Alt+S]
N. Virginia
voclabs/user3389058=2022.shreyas.naik@ves.ac.in @ 4346-6530-6979
Code
Test
Monitor
Configuration
Aliases
Versions
General configuration
Triggers
Permissions
Destinations
Function URL
Environment variables
Tags
VPC
RDS databases
Monitoring and operations tools
Concurrency and recursion detection
Triggers (1) Info
Fix errors
Edit
Delete
Add trigger
Find triggers
1
Trigger
SQS: mySQSqueue
arn:aws:sqs:us-east-1:678726468212:mySQSqueue
state: Enabled
Details


aws
Services
Search
[Alt+S]
N. Virginia
voclabs/user3389058=2022.shreyas.naik@ves.ac.in @ 4346-6530-6979
Amazon SQS
Queues
mySQSqueue
mySQSqueue
Edit
Delete
Purge
Send and receive messages
Start DLQ redrive
Details Info
Name
mySQSqueue
Type
Standard
ARN
arn:aws:sqs:us-east-1:678726468212:mySQSqueue
Encryption
Amazon SQS key (SSE-SQS)
URL
https://sqs.us-east-1.amazonaws.com/678726468212/mySQSqueue
Dead-letter queue
-
More
SNS subscriptions
Lambda triggers
EventBridge Pipes
Dead-letter queue
Monitoring
Tagging
Access policy
Encryption
Dead-letter queue redrive tasks
Subscription region

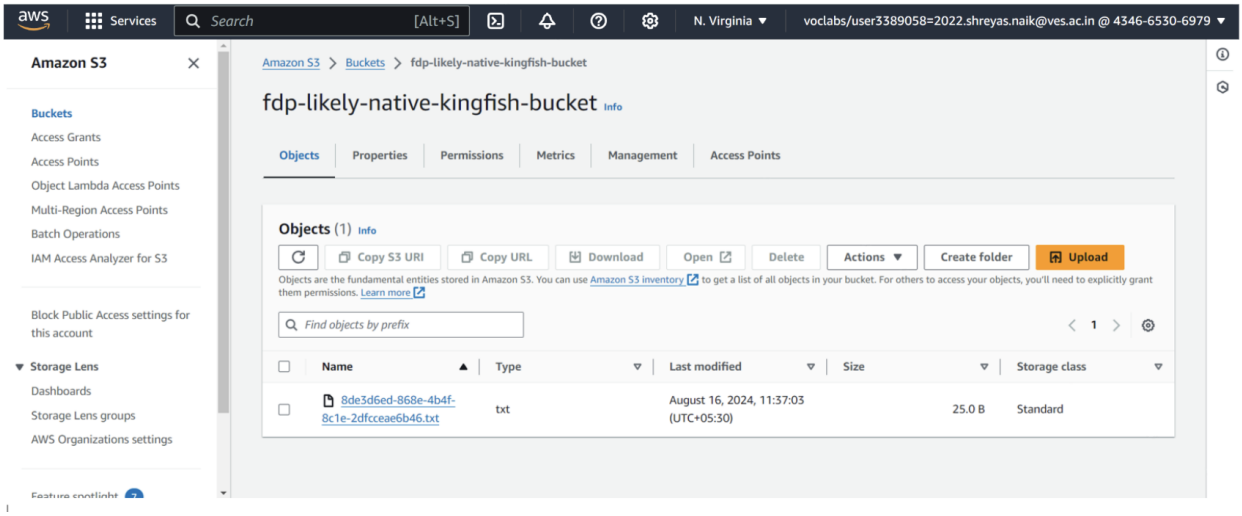

Services
Search
[Alt+S]
N. Virginia
voclabs/user3389058=2022.shreyas.naik@ves.ac.in @ 4346-6530-6979
Amazon SQS > Queues > mySQSqueue > Send and receive messages
Send and receive messages
Send messages to and receive messages from a queue.
Send message Info
Clear content
Send message
Message body
Enter the message to send to the queue.
Enter message
Delivery delay Info
0
Seconds
Should be between 0 seconds and 15 minutes.
Message attributes - Optional Info
Receive messages Info
Edit poll settings
Stop polling
Poll for messages

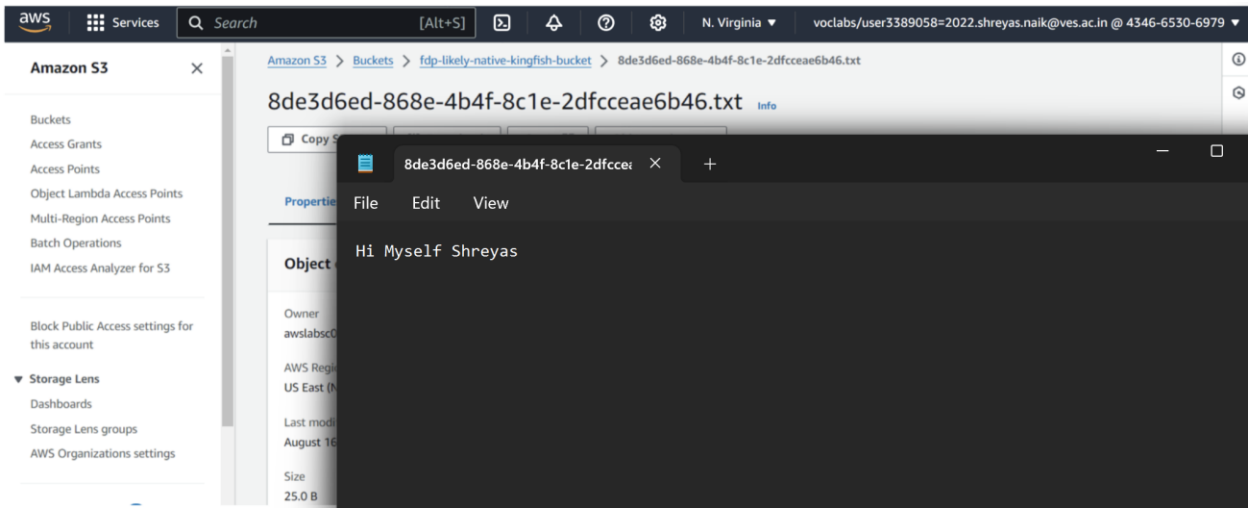

Services
Search
[Alt+S]
N. Virginia
voclabs/user3389058=2022.shreyas.naik@ves.ac.in @ 4346-6530-6979
Amazon SQS > Queues > mySQSqueue > Send and receive messages
Send and receive messages
Send messages to and receive messages from a queue.
Send message Info
Clear content
Send message
Message body
Enter the message to send to the queue.
Hi Myself Shreyas
Delivery delay Info
0
Seconds
Should be between 0 seconds and 15 minutes.
Message attributes - Optional Info
Receive messages Info
Edit poll settings
Stop polling
Poll for messages



Add S3 Event Notification for Lambda Trigger

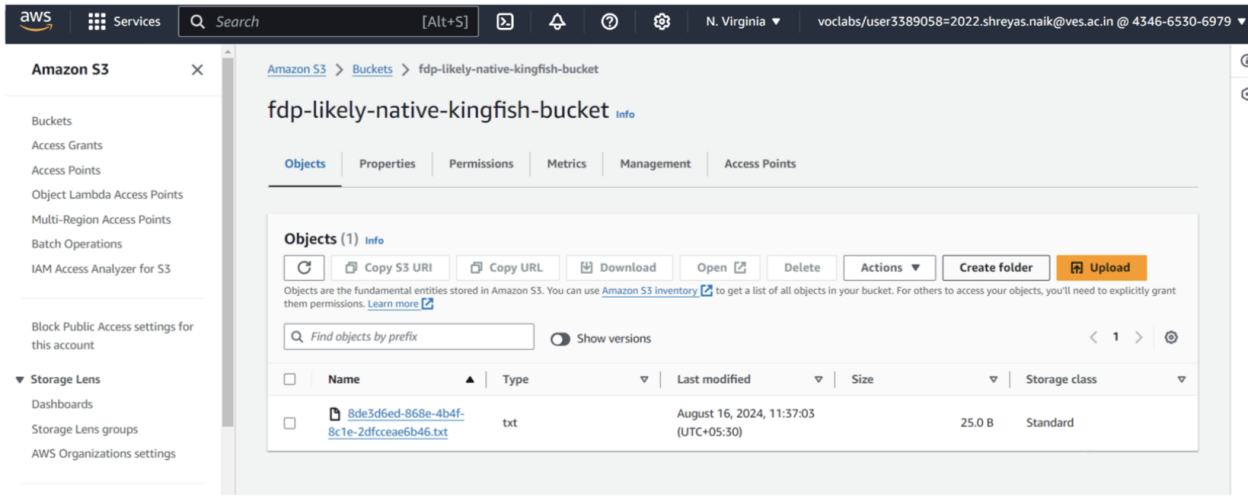
- Define a bucket notification to trigger the Lambda function when objects are uploaded to the S3 bucket.

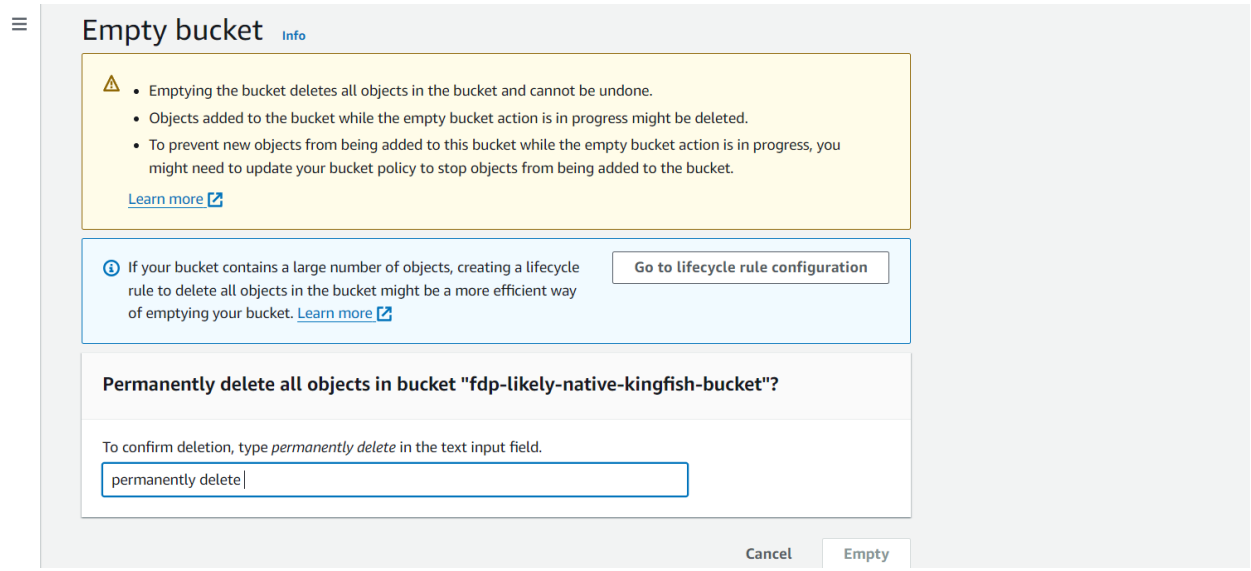




Initialize and Apply Terraform

- Initialize the Terraform configuration.





PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR

```
Microsoft Windows [Version 10.0.22000.2057]
(c) Microsoft Corporation. All rights reserved.

C:\Terraform>terraform destroy
random_pet.komalbucket717: Refreshing state... [id=fdp-likely-native-kingfish]
data.archive_file.zip: Reading...
data.archive_file.zip: Read complete after 0s [id=93c92209eafac774599673c33c7e7636e68e60e8]
aws_sqs_queue.myqueue: Refreshing state... [id=https://sqs.us-east-1.amazonaws.com/678726468212/mySQSqueue]
aws_s3_bucket.komalbucket717: Refreshing state... [id=fdp-likely-native-kingfish-bucket]
aws_lambda_event_source_mapping.SqsToLambda: Refreshing state... [id=4582b6aa-1865-4866-86eb-1062635c21a7]
aws_lambda_function.mykomlambda: Refreshing state... [id=SqsToS3Function]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  - destroy

Terraform will perform the following actions:

  # aws_lambda_event_source_mapping.SqsToLambda will be destroyed
  - resource "aws_lambda_event_source_mapping" "SqsToLambda" {
      - batch_size                       = 1 -> null
      - bisect_batch_on_function_error   = false -> null
      - enabled                          = true -> null
      - event_source_arn                 = "arn:aws:sqs:us-east-1:678726468212:mySQSqueue" -> null
      - function_arn                     = "arn:aws:lambda:us-east-1:678726468212:function:SqsToS3Function" -> null
      - function_name                    = "arn:aws:lambda:us-east-1:678726468212:function:SqsToS3Function" -> null
      - function_response_types          = [] -> null
      - id                               = "4582b6aa-1865-4866-86eb-1062635c21a7" -> null
      - last_modified                    = "2024-08-16T06:50:00Z" -> null
      - maximum_batching_window_in_seconds = 0 -> null
      - maximum_record_age_in_seconds    = 0 -> null
      - maximum_retry_attempts           = 0 -> null
      - parallelization_factor           = 0 -> null
      - queues                           = [] -> null
      - state                            = "Enabled" -> null
      - state_transition_reason          = "USER_INITIATED" -> null
      - topics                           = [] -> null
      - tumbling_window_in_seconds       = 0 -> null
      - uuid                             = "4582b6aa-1865-4866-86eb-1062635c21a7" -> null
```