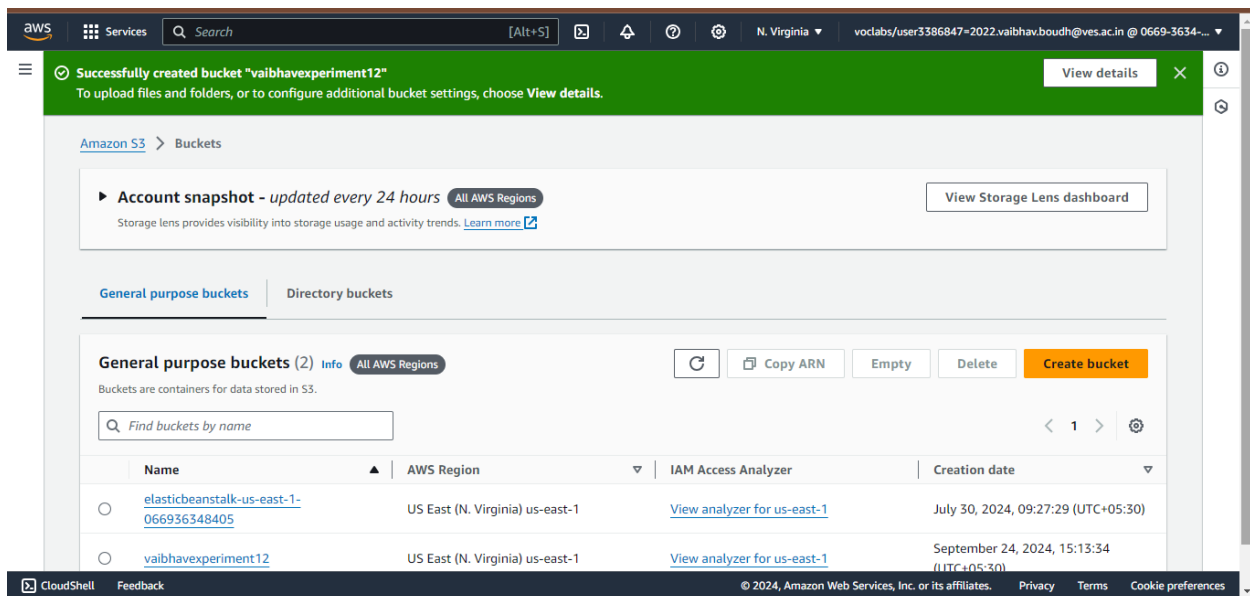


# Experiment 12

Here are the steps to create a Lambda function that logs “An Image has been added” once an object is added to a specific S3 bucket in AWS Learner Lab:

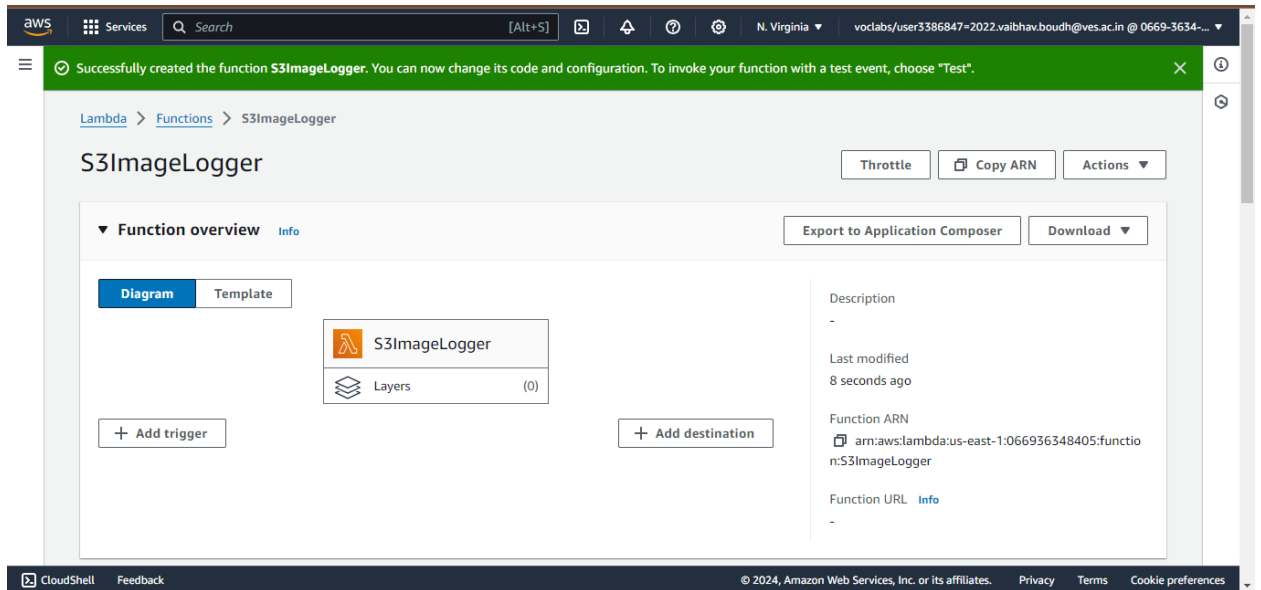
## 1. Create an S3 Bucket

- Go to the AWS Management Console.
- Navigate to the S3 service.
- Click on "Create bucket."
- Enter a unique bucket name and choose a region.
- Configure other settings as needed and click "Create bucket."



## 2. Create a Lambda Function

- Go to the AWS Management Console.
- Navigate to the Lambda service.
- Click on "Create function."
- Choose "Author from scratch."
- Enter a name for your function, e.g., **S3ImageLogger**.
- Select a runtime (e.g., Python 3.x or Node.js).
- Click "Create function."



### 3. Write the Lambda Function Code

- In the Lambda function console, scroll down to the code editor.

Replace the default code with the following code snippet (assuming you're using Python):

python

Copy code

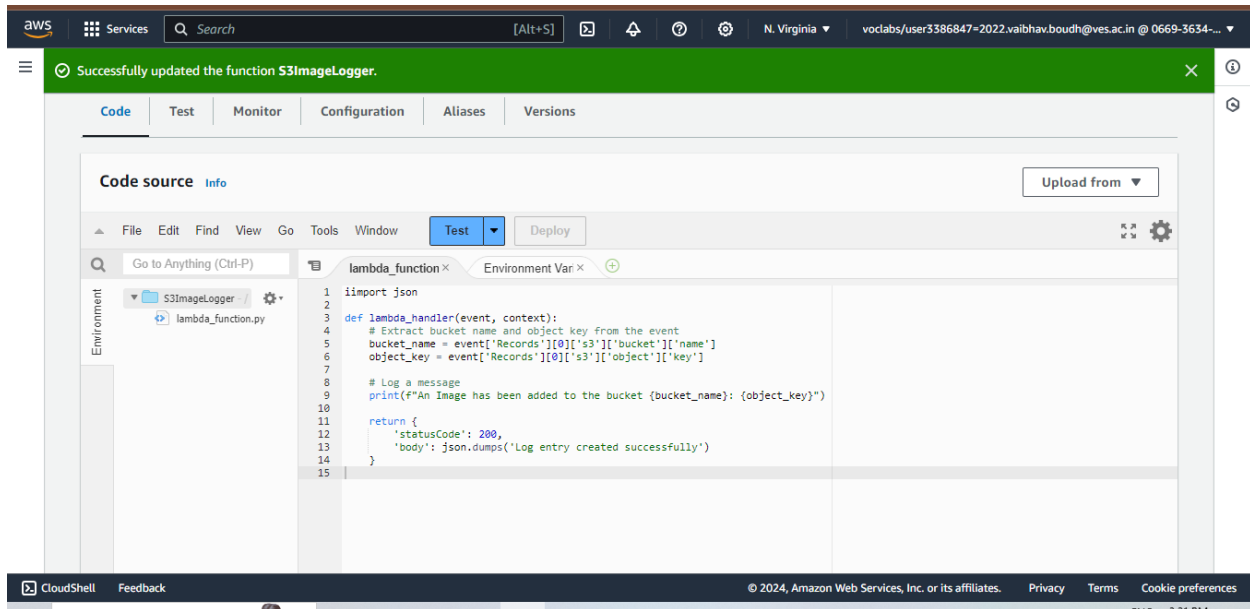
```
import json
```

```
def lambda_handler(event, context):
    # Extract bucket name and object key from the event
    bucket_name = event['Records'][0]['s3']['bucket']['name']
    object_key = event['Records'][0]['s3']['object']['key']

    # Log a message
    print(f"An Image has been added to the bucket {bucket_name}: {object_key}")

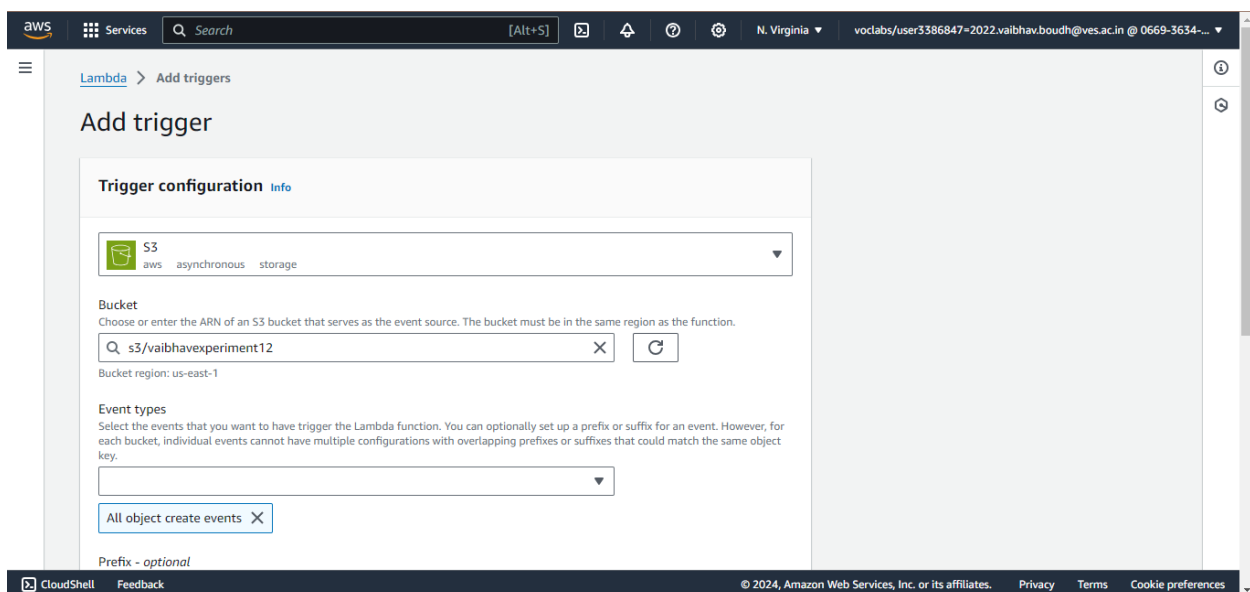
    return {
        'statusCode': 200,
        'body': json.dumps('Log entry created successfully')
    }
```

- 
- Click "Deploy" to save your changes.



## 4. Set Up S3 Trigger for the Lambda Function

- Scroll down to the "Function overview" section in the Lambda console.
- Click on "Add trigger."
- Select "S3" from the list of triggers.
- Choose the S3 bucket you created earlier.
- In the "Event type" dropdown, select "All object create events."
- Optionally, specify a prefix or suffix to filter the events (e.g., for images only, you can use suffix `.jpg`, `.png`).
- Click "Add."

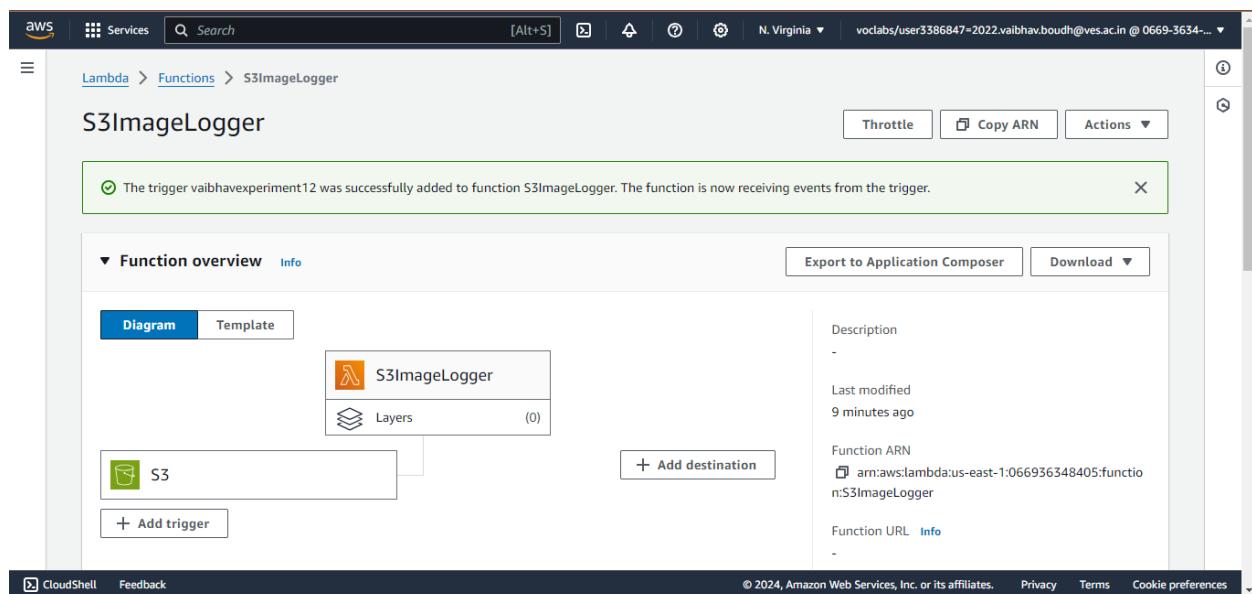


## 5. Grant Permissions to Lambda

- Navigate to the "Permissions" tab of your Lambda function.
- Ensure the Lambda function's execution role has the necessary permissions to access the S3 bucket.
- If needed, attach the [AmazonS3ReadOnlyAccess](#) policy or create a custom policy with the necessary permissions.

## 6. Test the Setup

- Upload an image file to your S3 bucket.
- Go to the "Monitoring" tab in your Lambda function to check the logs.
- Alternatively, use CloudWatch Logs to view the output and confirm that the message "An Image has been added" has been logged.



This setup should ensure that each time an image is uploaded to the specified S3 bucket, the Lambda function will log the appropriate message.

```
import json
import boto3
import urllib
```

```
def lambda_handler(event, context):
    s3_client=boto3.client('s3')
    bucket_name=event["Records"][0]['s3']['bucket']['name']
    key=event["Records"][0]['s3']['object']['key']
    key=urllib.parse.unquote_plus(key,encoding='utf-8')
```

```
message='ping! file was uploaded with key' + key +'to bucket ' + bucket_name  
print(message)
```

```
response=s3_client.getobject(Bucket=bucket_name,Key=key)
```

```
contents=response["Body"].read().decode()
```

```
contents=json.loads(co
```