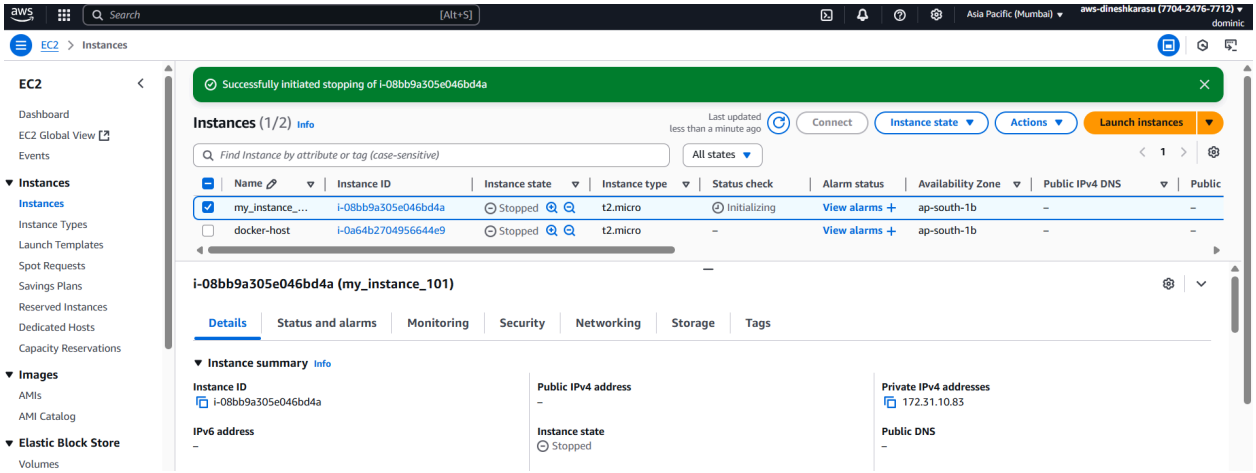
| Use case | Use case Description |
| --- | --- |
| **Serverless Image Processing** | Upload images to S3, process with Lambda (resize), store in another bucket. |

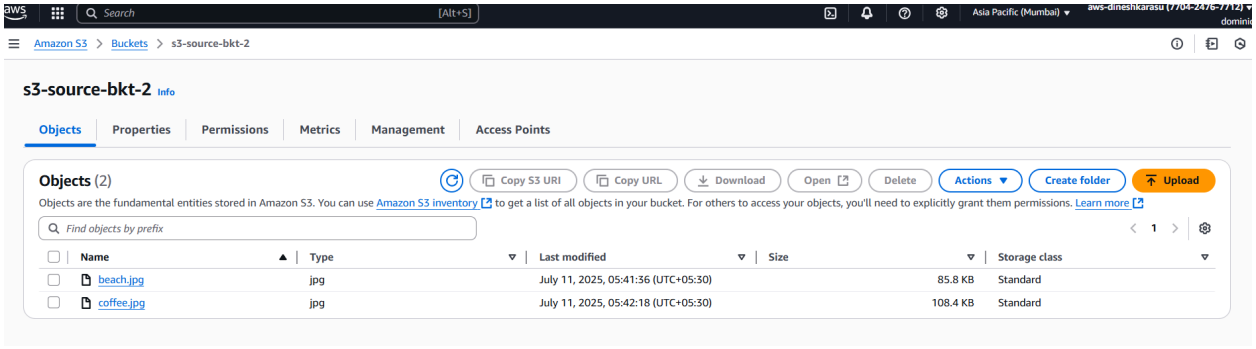
## Approach :

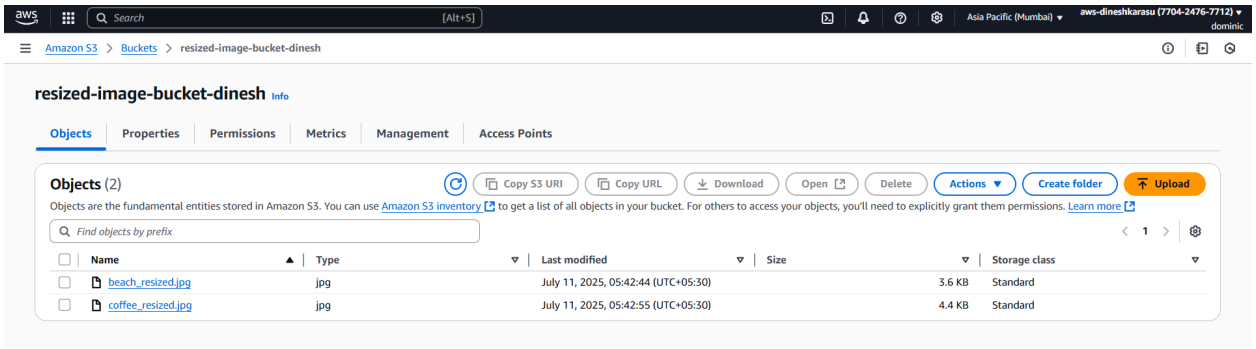
1. Create two S3 Buckets one for original file uploads and the other for storing resized images
2. Create a Lambda function and assign permissions
3. Code Lambda in a way that it reads the file uploaded to first bucket and uploads the compressed image to second image
4. Add a new layer to import Pillow package where we use built in function to resize the image
5. Create an EC2 instance and install python 3.12, necessary dependencies for pillow
6. ZIp the files and SCP into your local machine
7. Create and attach the layer to Lambda function
8. Run and test

Create an EC2 instance

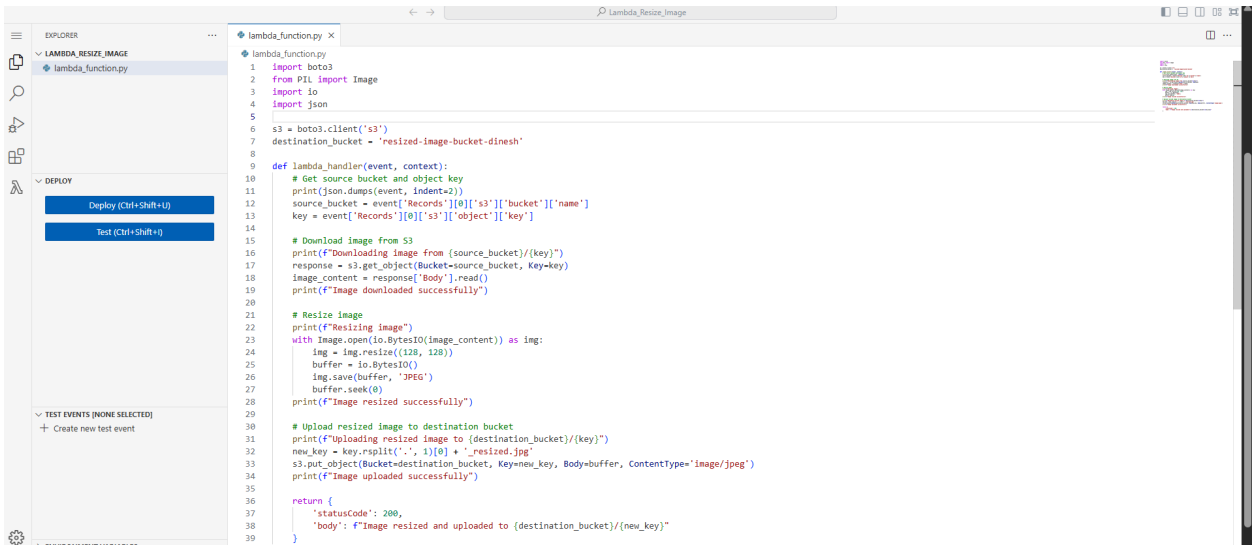



Create two S3 buckets - source ( to store original file ) and destination bucket( to store resized files )




## Create Lambda

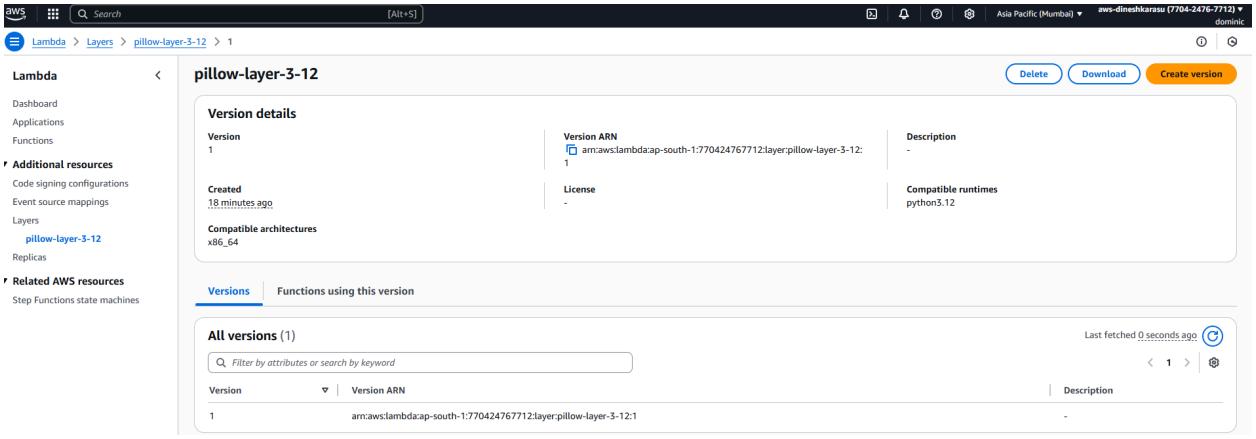


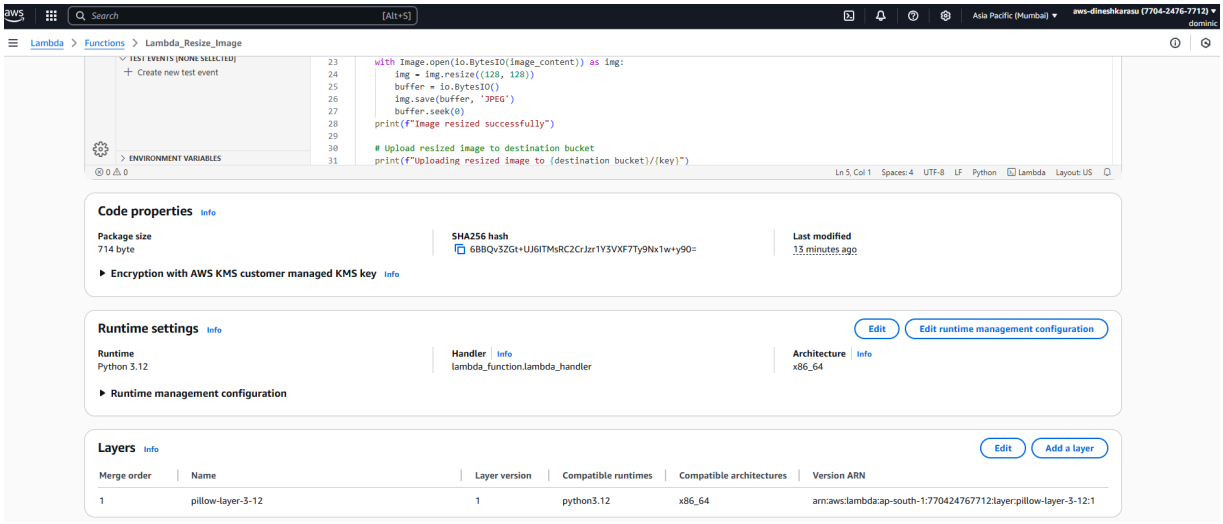

## SSH into that instance

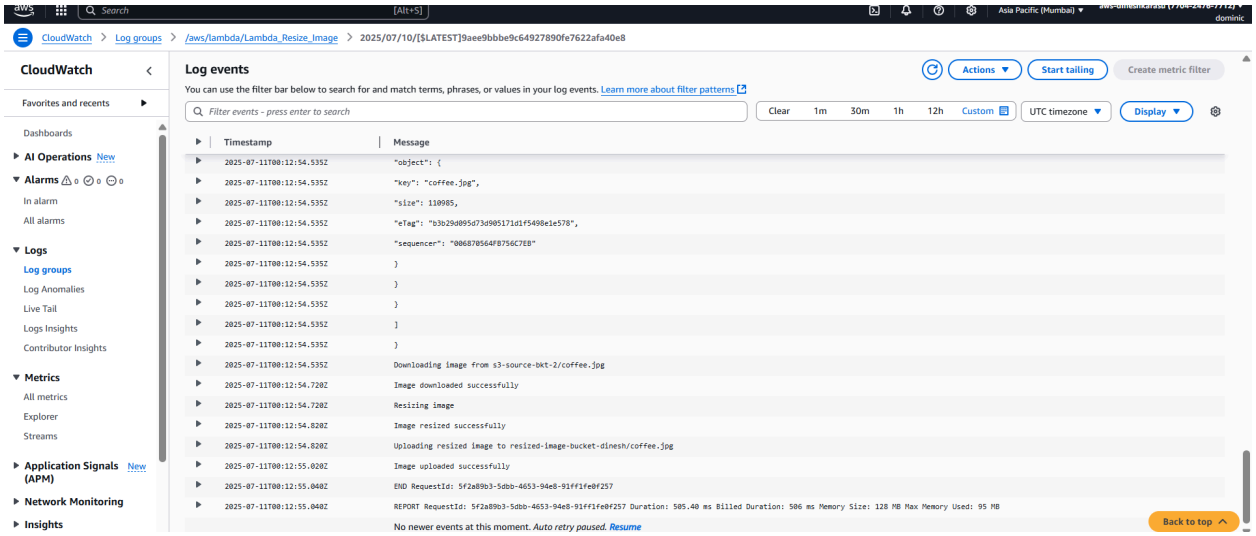
```
PS C:\Users\dkarasu\desktop\AWSCloudPractitioner\KeyPairs> ssh -i .\MumbaiRegion.pem ec2-user@65.2.153.4
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-172-31-10-83 ~]$ sudo dnf install python3.12 python3.12-pip -y
Amazon Linux 2023 Kernel Livepatch repository                                       161 kB/s |  17 kB     00:00
Dependencies resolved.
================================================================================================================
 Package                         Architecture       Version                          Repository              Size
================================================================================================================
Installing:
 python3.12                      x86_64             3.12.10-2.amzn2023.0.2           amazonlinux             28 k
 python3.12-pip                  noarch             23.2.1-4.amzn2023.0.2            amazonlinux            2.8 M
Installing dependencies:
```

```
Complete!
[ec2-user@ip-172-31-10-83 ~]$ python3.12 --version
Python 3.12.10
[ec2-user@ip-172-31-10-83 ~]$ mkdir -p pillow-layer/python
[ec2-user@ip-172-31-10-83 ~]$ cd pillow-layer/python
[ec2-user@ip-172-31-10-83 python]$ python3.12 -m pip install Pillow -t .
Collecting Pillow
  Obtaining dependency information for Pillow from https://files.pythonhosted.org/packages/e4/c9/06dd4a389
8ea3c546ce3f8c995d3f0985f8e5ba48bba19/pillow-11.3.0-cp312-cp312-manylinux_2_27_x86_64.manylinux_2_28_x86_6
  Downloading pillow-11.3.0-cp312-cp312-manylinux_2_27_x86_64.manylinux_2_28_x86_64.whl.metadata (9.0 kB)
Downloading pillow-11.3.0-cp312-cp312-manylinux_2_27_x86_64.manylinux_2_28_x86_64.whl (6.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6.6/6.6 MB 89.8 MB/s eta 0:00:00
Installing collected packages: Pillow
Successfully installed Pillow-11.3.0
[ec2-user@ip-172-31-10-83 python]$ cd ..
[ec2-user@ip-172-31-10-83 pillow-layer]$ zip -r pillow-layer.zip python
```

```
[ec2-user@ip-172-31-10-83 pillow-layer]$ exit
logout
Connection to 65.2.153.4 closed.
PS C:\Users\dkarasu\desktop\AWSCloudPractitioner\KeyPairs> scp -i .\MumbaiRegion.pem ec2-user@65.2.153.4:/home/ec2-user/
pillow-layer/pillow-layer.zip .
pillow-layer.zip                                                                    100% 6947KB   2.1MB/s   00:03
PS C:\Users\dkarasu\desktop\AWSCloudPractitioner\KeyPairs>
```




Lambda > Functions > Lambda_Resize_Image
Create new test event
ENVIRONMENT VARIABLES
with Image.open(io.BytesIO(image_content)) as img:
img = img.resize((128, 128))
buffer = io.BytesIO()
img.save(buffer, 'JPEG')
buffer.seek(0)
print(f"Image resized successfully")
# Upload resized image to destination bucket
print(f"Uploading resized image to {destination_bucket}/{key}")
Code properties Info
Package size
714 byte
SHA256 hash
6BBQv3ZGt+UJ6ITMsRC2CrJzr1Y3VXF7Ty9Nx1w+y90=
Last modified
13 minutes ago
Encryption with AWS KMS customer managed KMS key Info
Runtime settings Info
Edit
Edit runtime management configuration
Runtime
Python 3.12
Handler Info
lambda_function.lambda_handler
Architecture Info
x86_64
Runtime management configuration
Layers Info
Edit
Add a layer
Merge order
Name
Layer version
Compatible runtimes
Compatible architectures
Version ARN
1
pillow-layer-3-12
1
python3.12
x86_64
arn:aws:lambda:ap-south-1:770424767712:layer:pillow-layer-3-12:1


CloudWatch > Log groups > /aws/lambda/Lambda_Resize_Image > 2025/07/10/[$LATEST]9aee9bbbe9c64927890fe7622afa40e8
CloudWatch
Favorites and recents
Dashboards
AI Operations New
Alarms
In alarm
All alarms
Logs
Log groups
Log Anomalies
Live Tail
Logs Insights
Contributor Insights
Metrics
All metrics
Explorer
Streams
Application Signals (APM) New
Network Monitoring
Insights
Log events
You can use the filter bar below to search for and match terms, phrases, or values in your log events. Learn more about filter patterns
Actions
Start tailing
Create metric filter
Filter events - press enter to search
Clear
1m
30m
1h
12h
Custom
UTC timezone
Display
Timestamp
Message
2025-07-11T00:12:54.535Z "object": {
2025-07-11T00:12:54.535Z "key": "coffee.jpg",
2025-07-11T00:12:54.535Z "size": 110985,
2025-07-11T00:12:54.535Z "eTag": "b3b29d095d73d905171d1f5498e1e578",
2025-07-11T00:12:54.535Z "sequencer": "006870564F8756C7EB"
2025-07-11T00:12:54.535Z }
2025-07-11T00:12:54.535Z }
2025-07-11T00:12:54.535Z }
2025-07-11T00:12:54.535Z ]
2025-07-11T00:12:54.535Z }
2025-07-11T00:12:54.535Z Downloading image from s3-source-bkt-2/coffee.jpg
2025-07-11T00:12:54.720Z Image downloaded successfully
2025-07-11T00:12:54.720Z Resizing image
2025-07-11T00:12:54.820Z Image resized successfully
2025-07-11T00:12:54.820Z Uploading resized image to resized-image-bucket-dinesh/coffee.jpg
2025-07-11T00:12:55.020Z Image uploaded successfully
2025-07-11T00:12:55.040Z END RequestId: 5f2a89b3-5dbb-4653-94e8-91ff1fe0f257
2025-07-11T00:12:55.040Z REPORT RequestId: 5f2a89b3-5dbb-4653-94e8-91ff1fe0f257 Duration: 505.40 ms Billed Duration: 506 ms Memory Size: 128 MB Max Memory Used: 95 MB
No newer events at this moment. Auto retry paused. Resume
Back to top