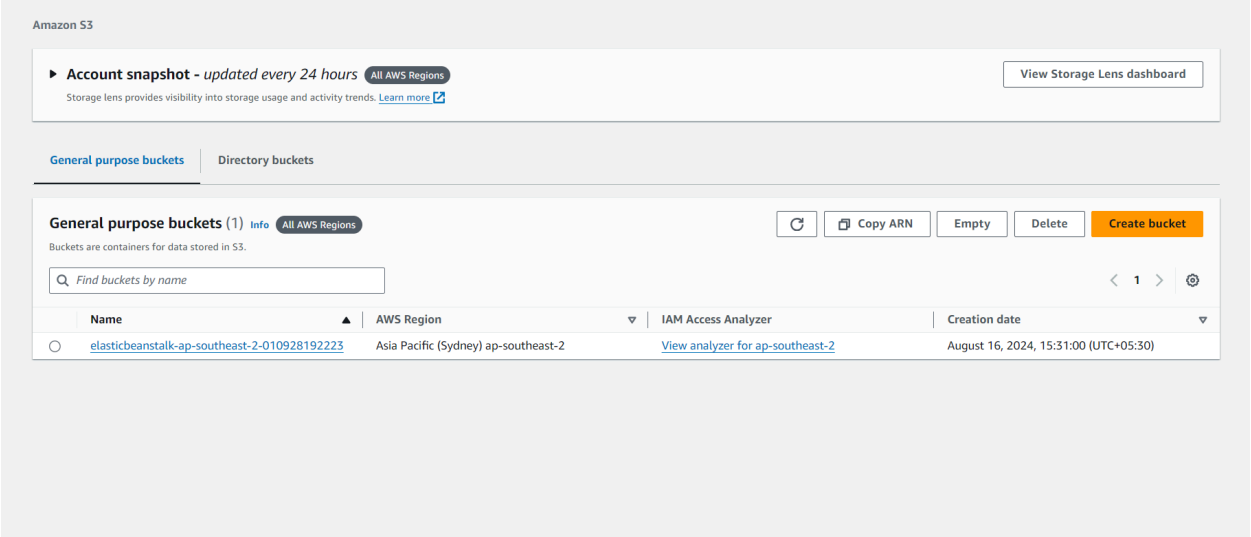
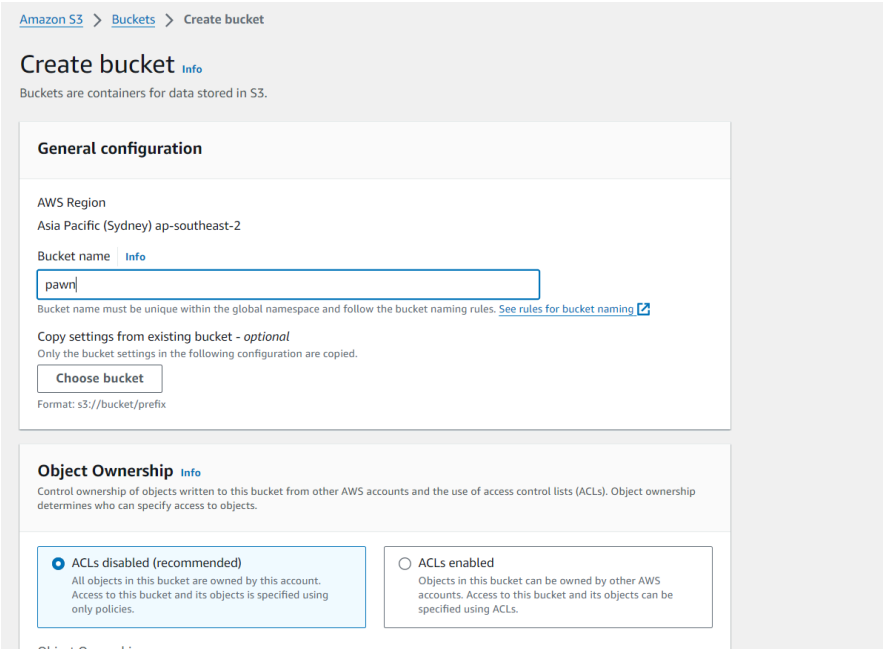
EXPERIMENT NO. 12

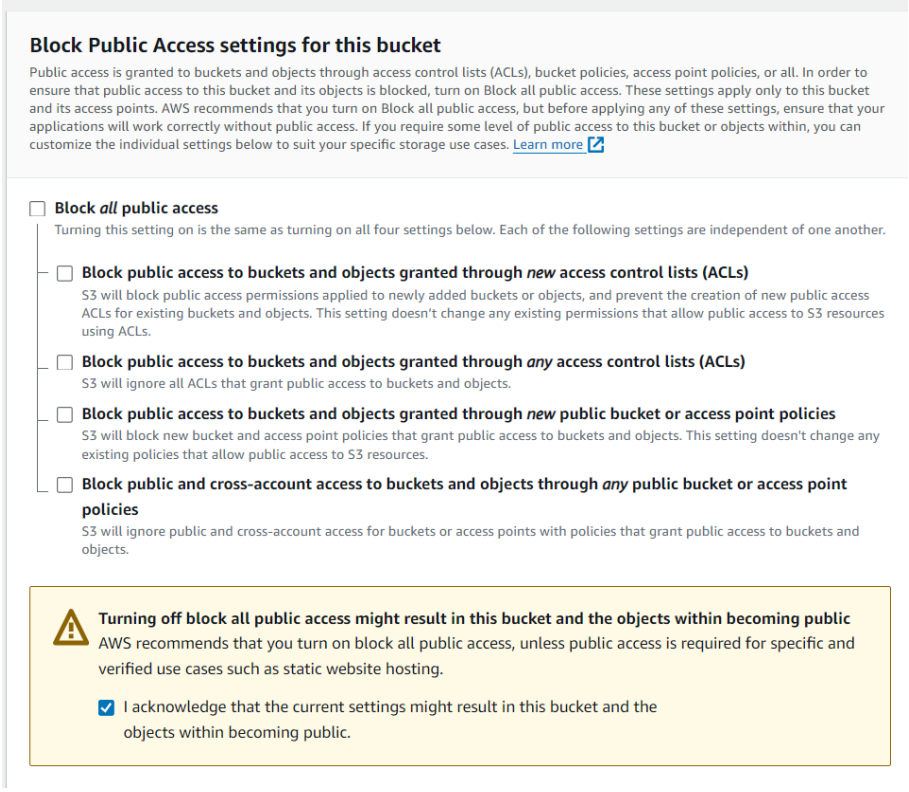
Aim: To create a Lambda function which will log “An Image has been added” once you add an object to a specific bucket in S3.

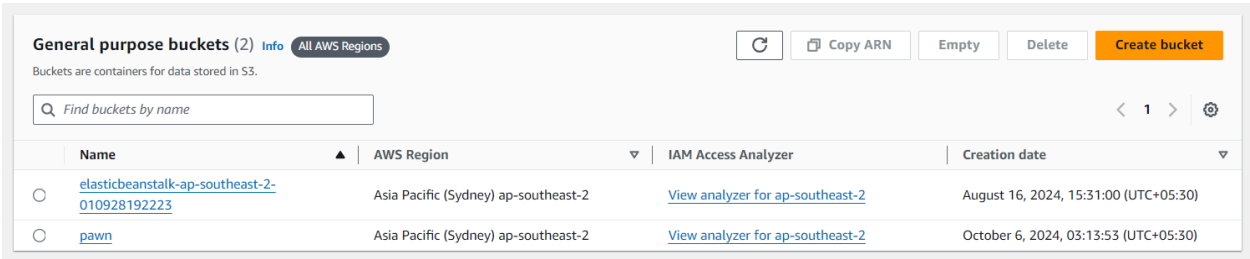
1. Login to your AWS Personal account. Now open S3 from services and click on create S3 bucket.



2. Now Give a name to the Bucket, select general purpose project and deselect the Block public access and keep other things to default.

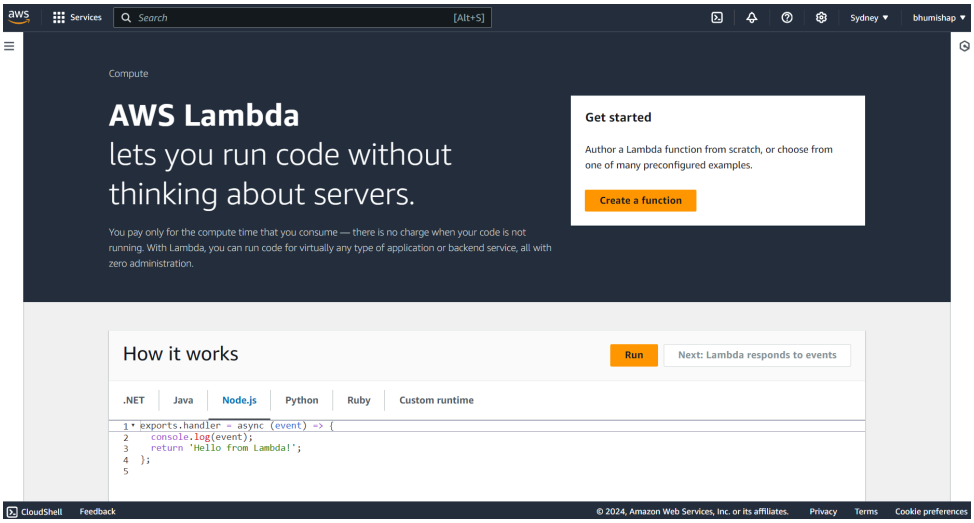




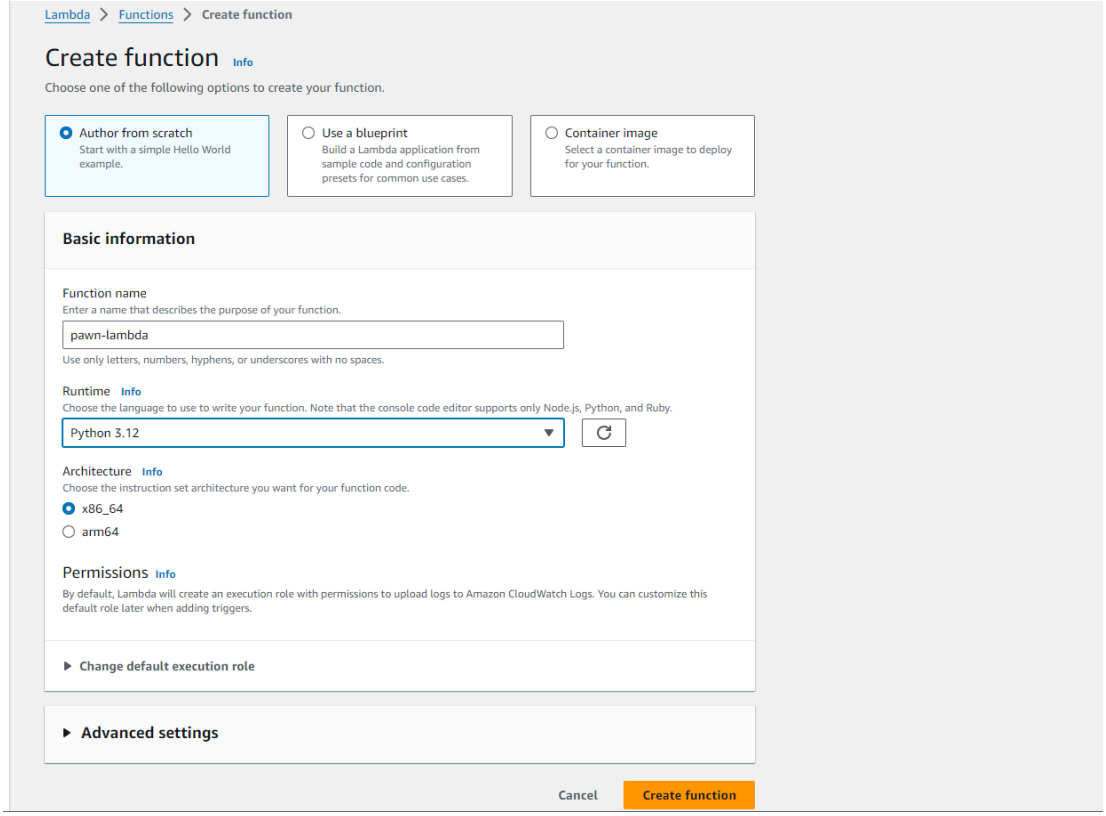


Thus, we have created a bucket named *pawn*.

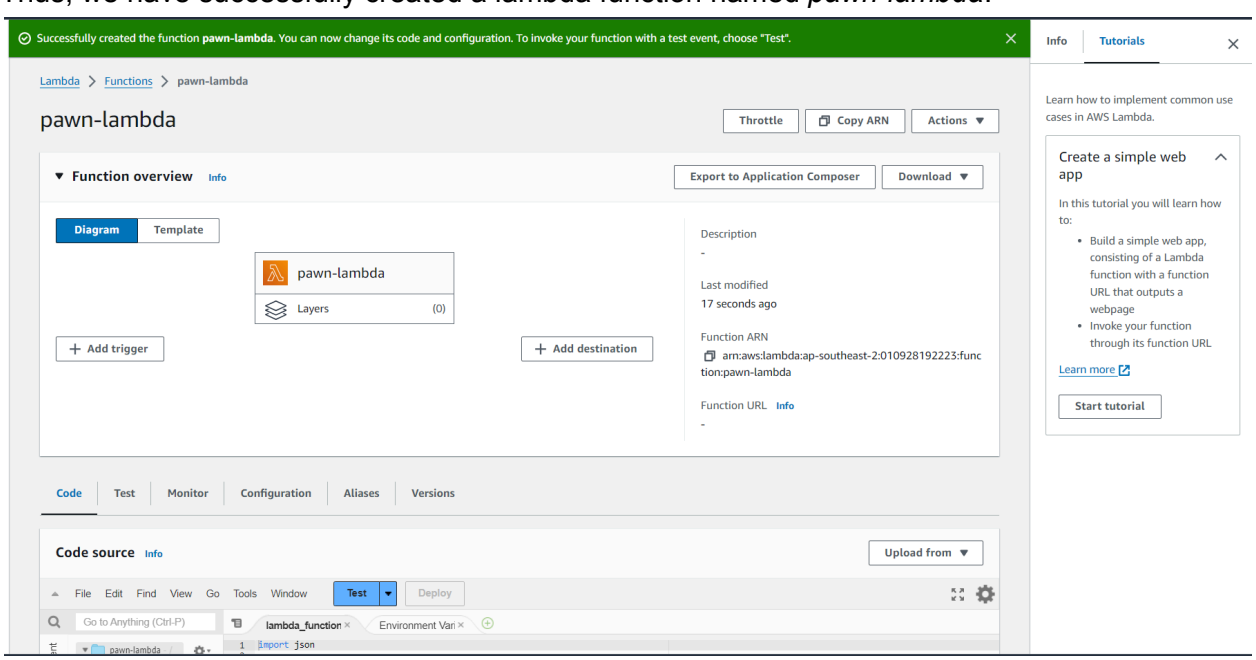
3. Open lambda console and click on the create function button.



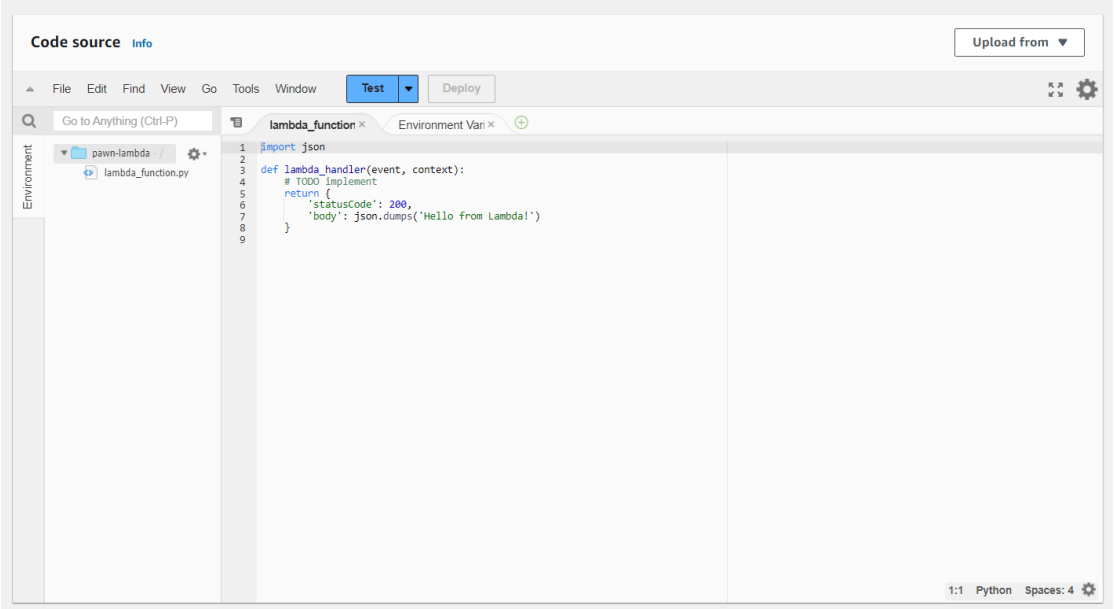
4. Now Give a name to your Lambda function, Select the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby. So will select Python 3.12, Architecture as x86 and Exceution role to Create a new role with basic Lambda permissions



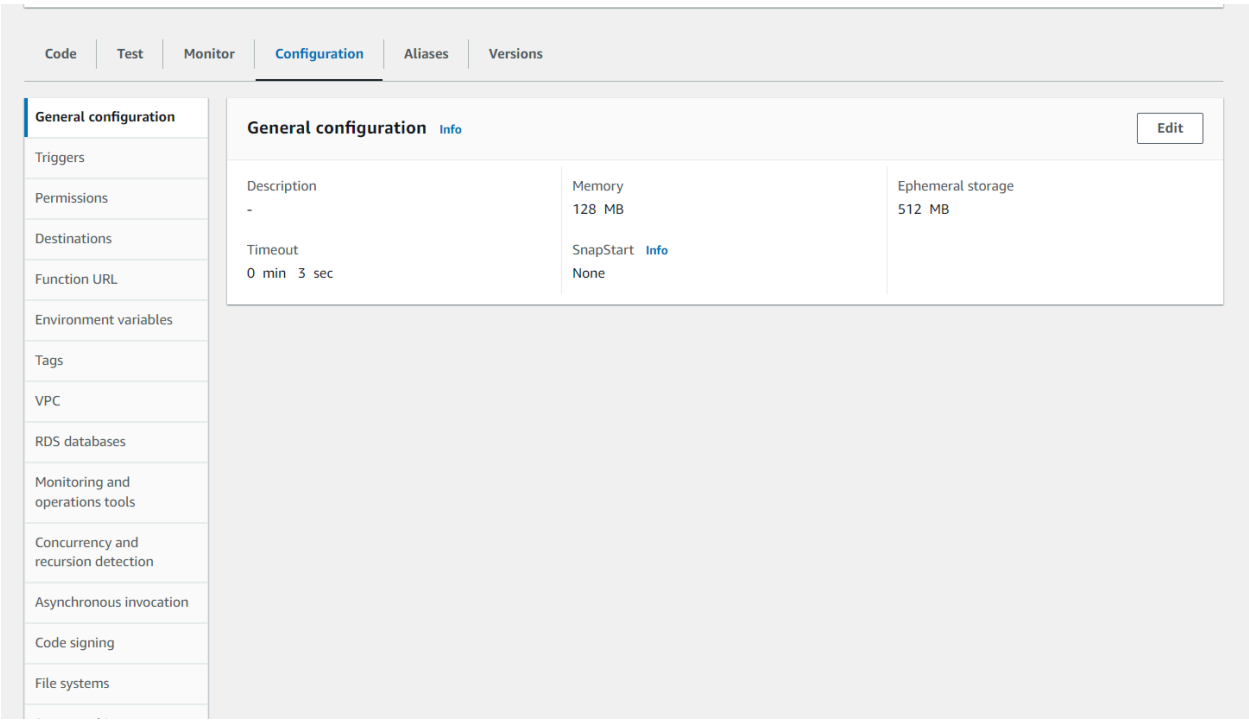
Thus, we have successfully created a lambda function named *pawn-lambda*.



This is how the function is displayed.

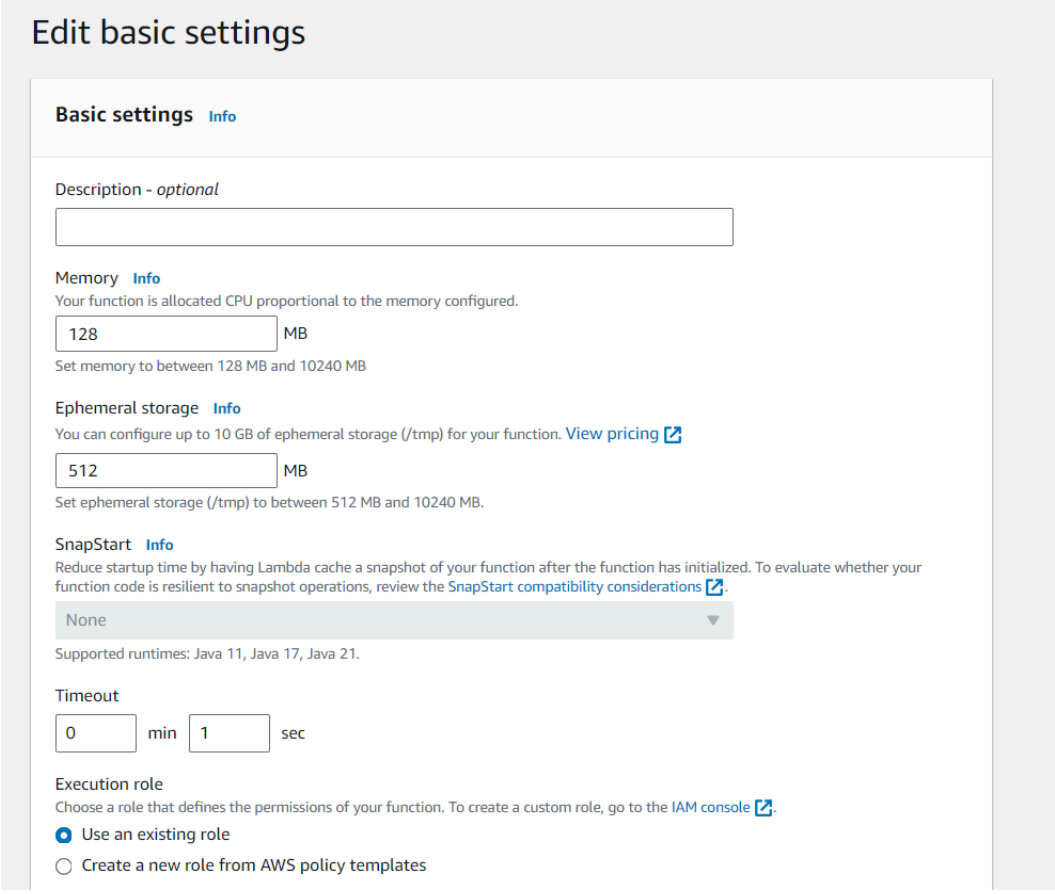


Go to the Configuration tab to see the general configuration of our function.

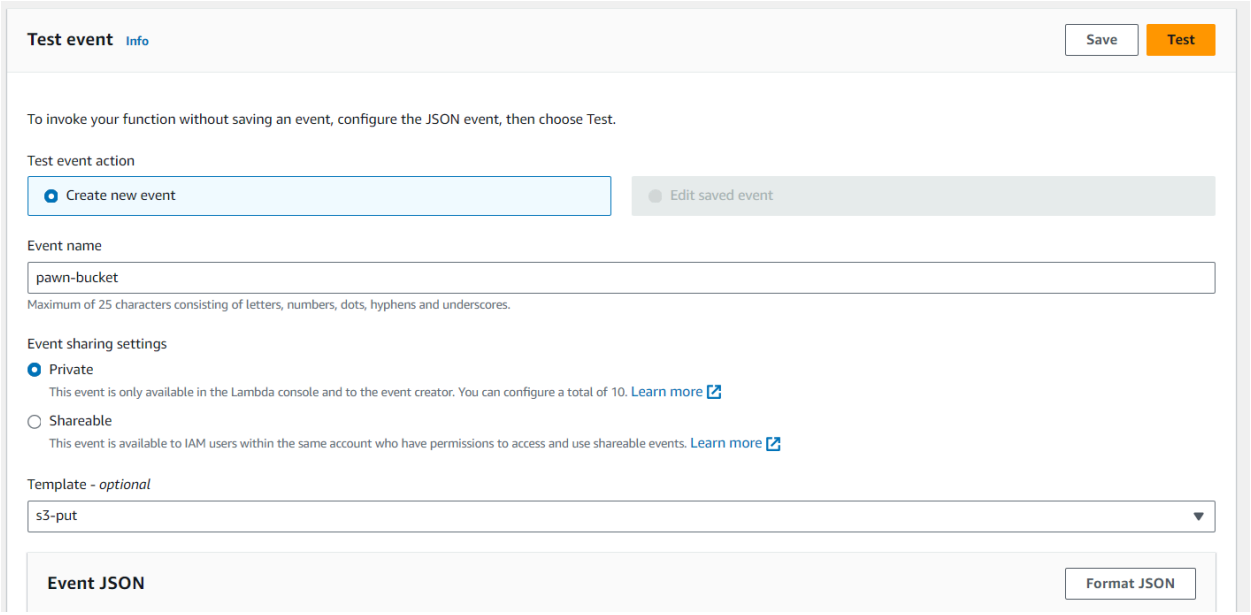


We want to edit the timeout time and the rest can be kept the default.
Here, we can enter a description and change Memory and Timeout. I've changed the Timeout period to 1 sec since that is sufficient for now.

To edit, we go to the Edit button seen in the above screenshot.

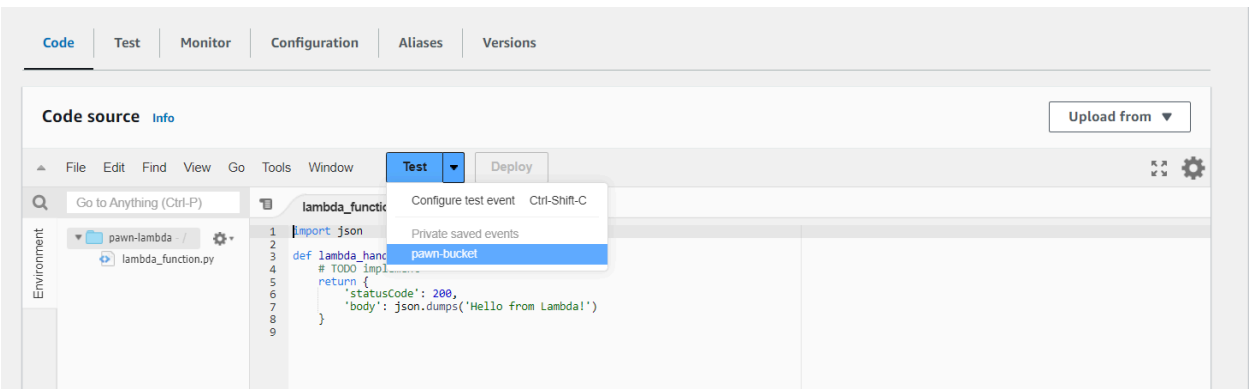


5. Now Click on the Test tab then select Create a new event, give a name to the event and select Event Sharing to private, and select s3 put template.

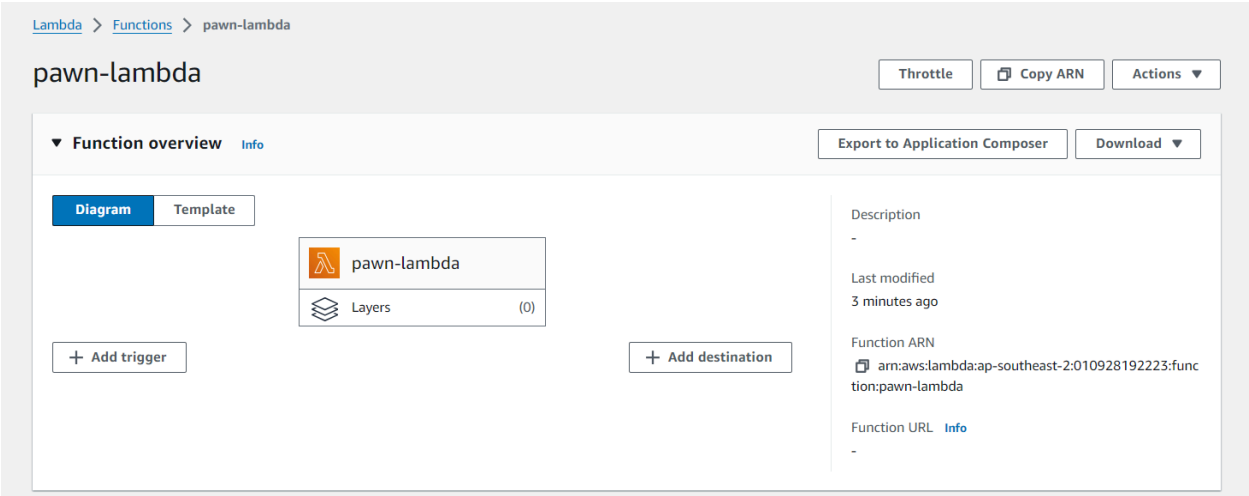



```
{
  "Records": [
    {
      "eventVersion": "2.0",
      "eventSource": "aws:s3",
      "awsRegion": "us-east-1",
      "eventTime": "1970-01-01T00:00:00.000Z",
      "eventName": "ObjectCreated:Put",
      "userIdentity": {
        "principalId": "EXAMPLE"
      },
      "requestParameters": {
        "sourceIPAddress": "127.0.0.1"
      },
      "responseElements": {
        "x-amz-request-id": "EXAMPLE123456789",
        "x-amz-id-2": "EXAMPLE123/5678abcdefghijklambdaisawesome/mnopqrstuvwxyzABCDEFGH"
      },
      "s3": {
        "s3SchemaVersion": "1.0",
        "configurationId": "testConfigRule",
        "bucket": {
          "name": "example-bucket",
          "ownerIdentity": {
            "principalId": "EXAMPLE"
          },
          "arn": "arn:aws:s3:::example-bucket"
        },
        "object": {
          "key": "test%2Fkey",
```

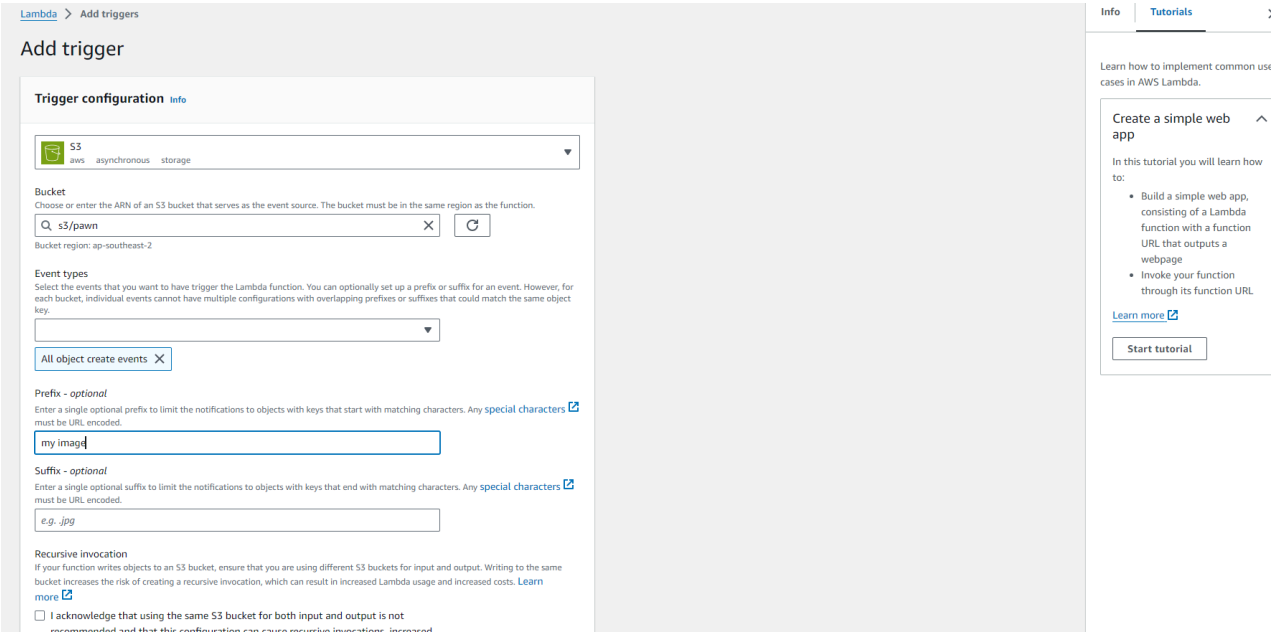

6. Now in the Code section select the created event from the dropdown.

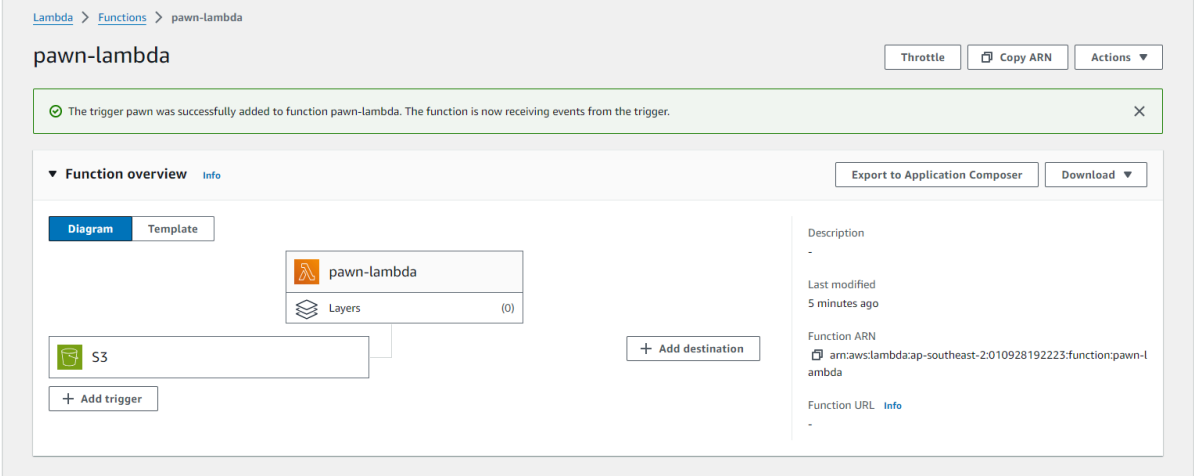


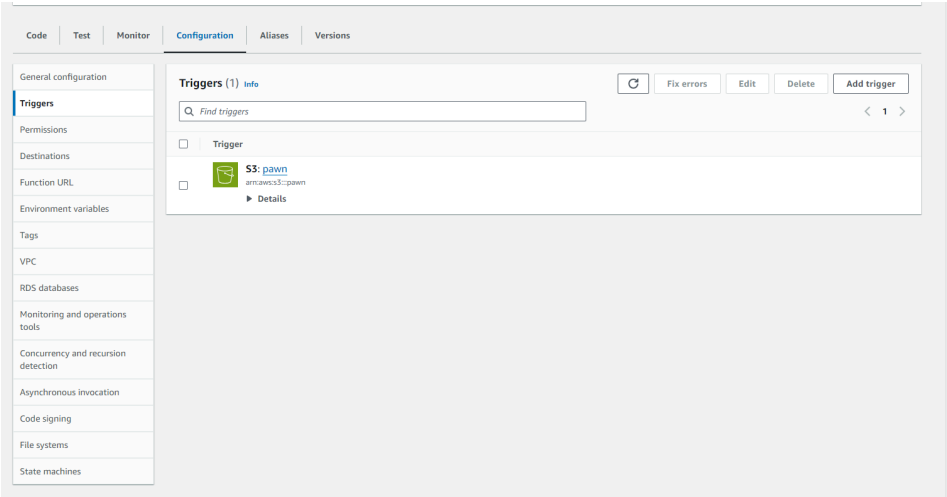
7. Now In the Lambda function click on add trigger.



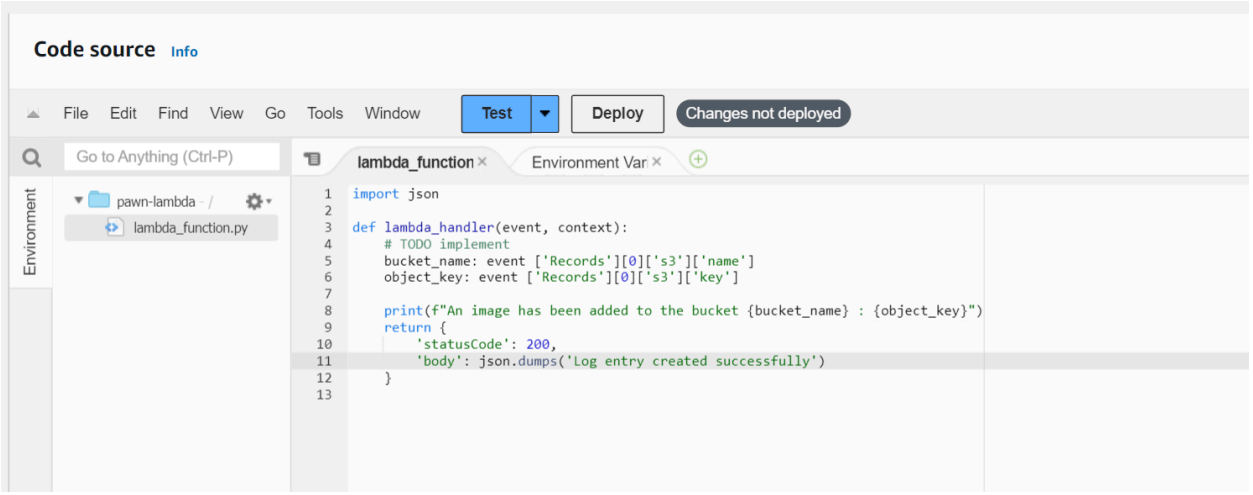
Now select the source as S3 then select the bucket name from the dropdown, keep other things to default and also you can add prefix to image

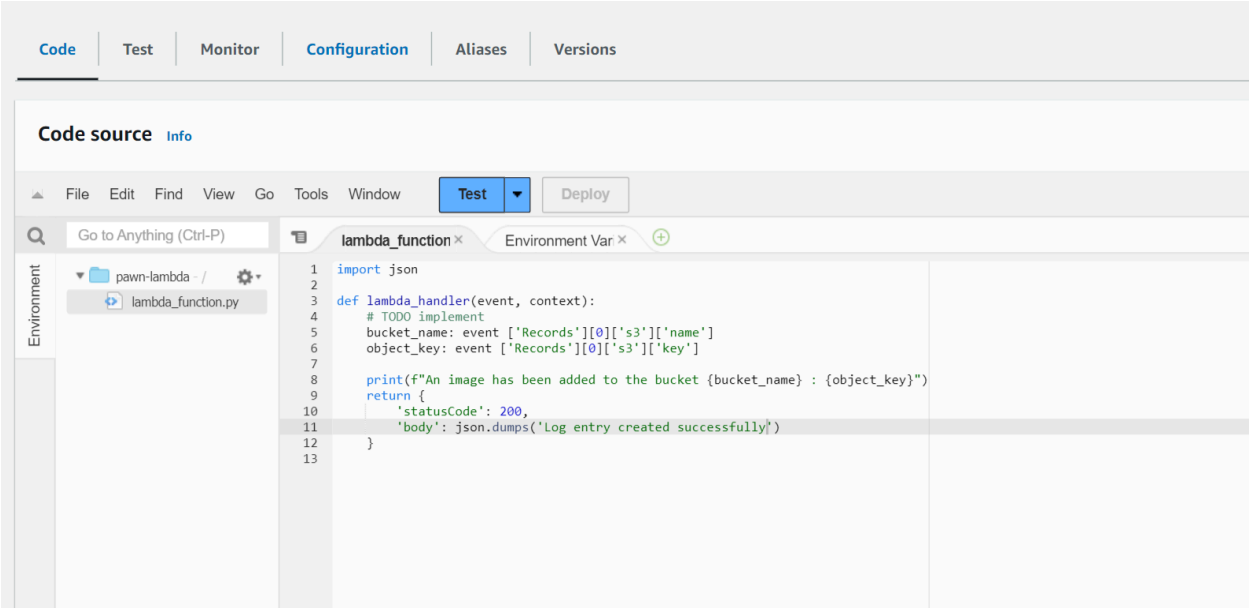




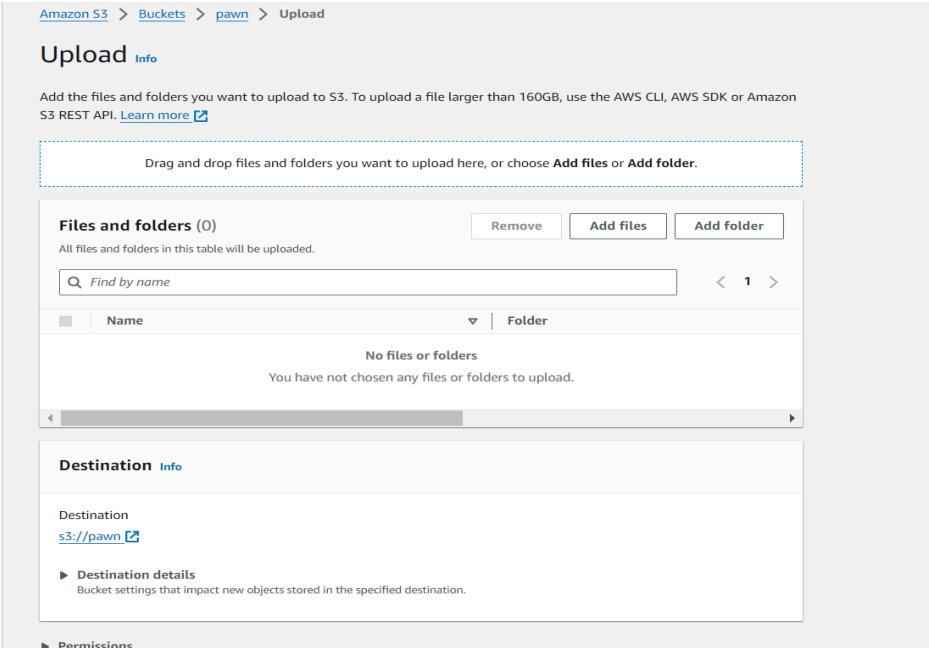


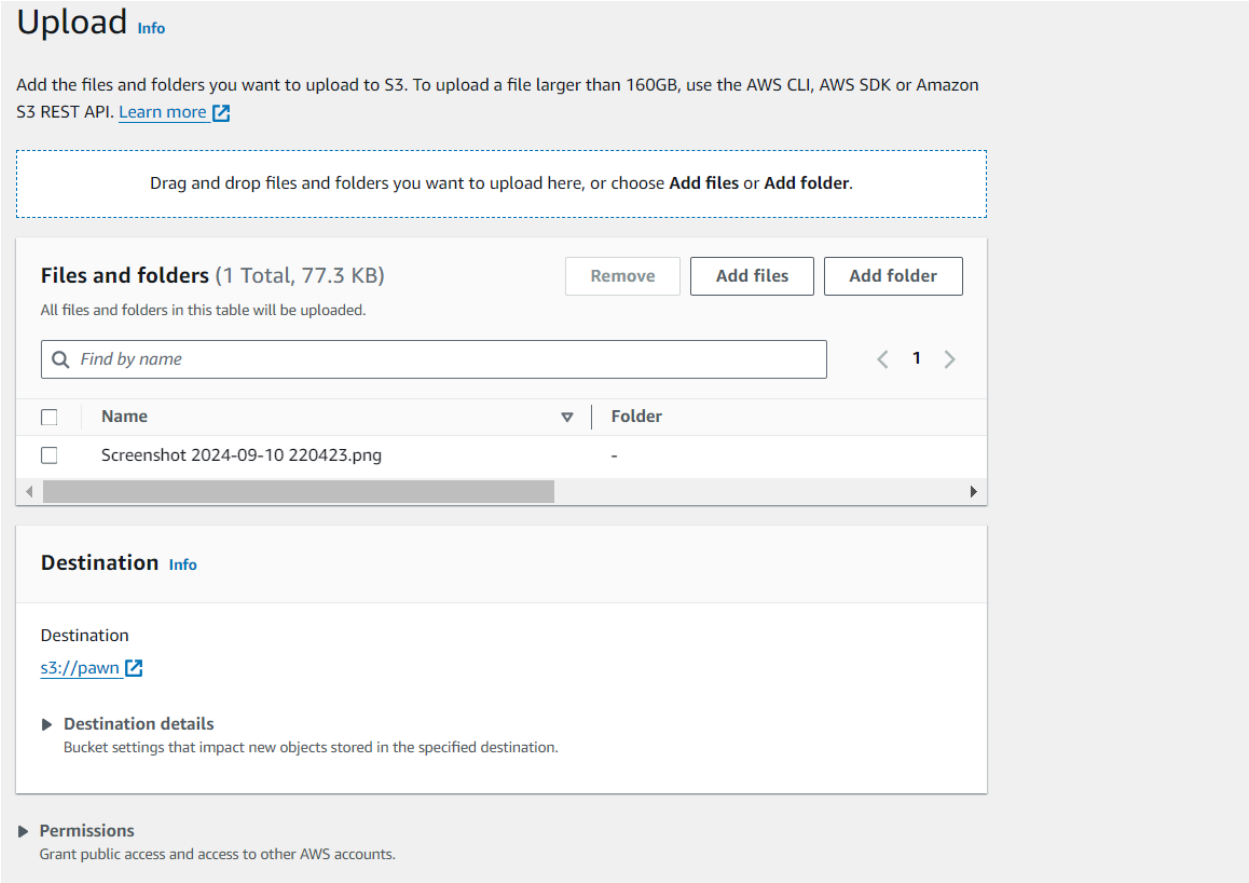
8. Now Write code that logs a message like “An Image has been added” when triggered. Save the file and click on deploy.

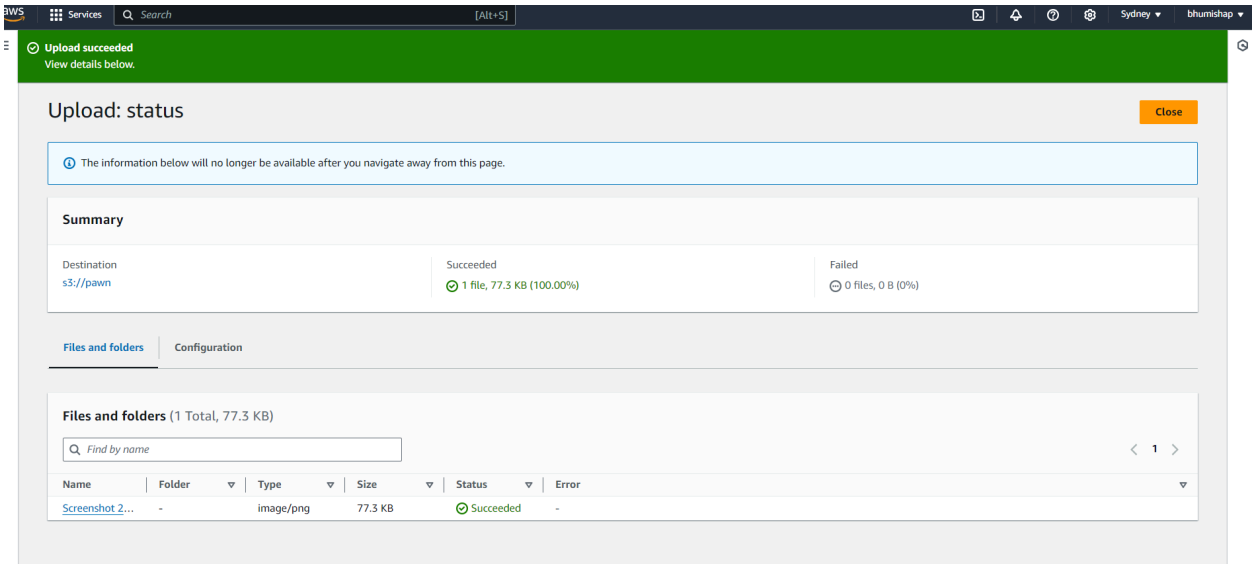




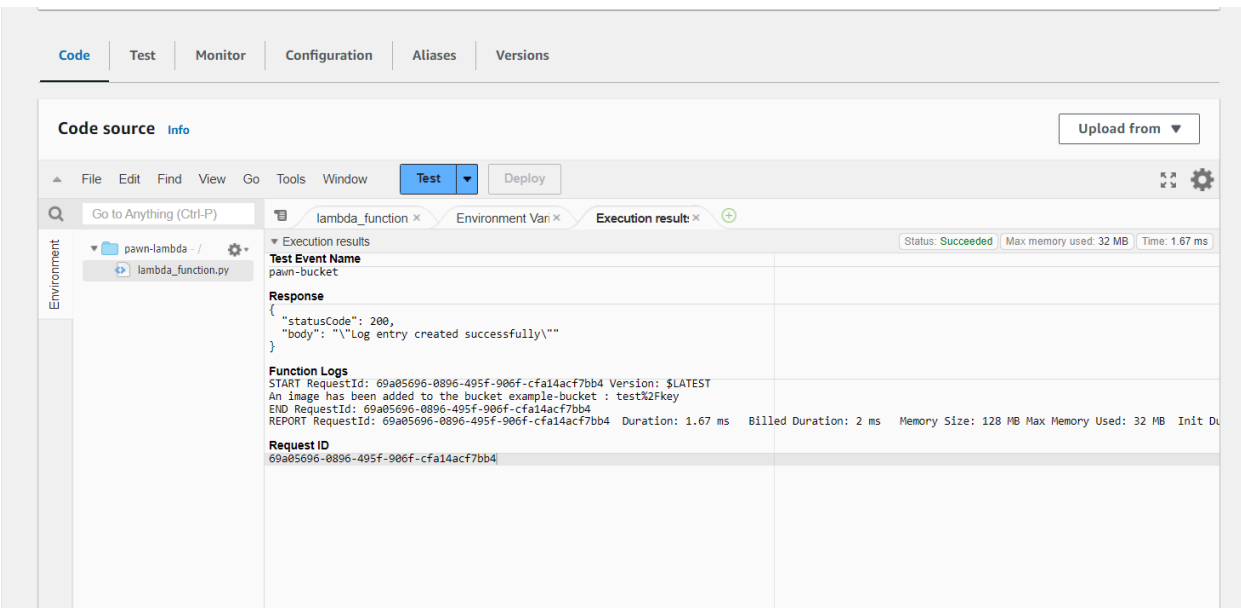
9. Now upload any image to the bucket.


Amazon S3 > Buckets > pawn > Upload
Upload Info
Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more
Drag and drop files and folders you want to upload here, or choose Add files or Add folder.
Files and folders (0)
All files and folders in this table will be uploaded.
Remove
Add files
Add folder
Find by name
1
Name
Folder
No files or folders
You have not chosen any files or folders to upload.
Destination Info
Destination
s3://pawn
Destination details
Bucket settings that impact new objects stored in the specified destination.

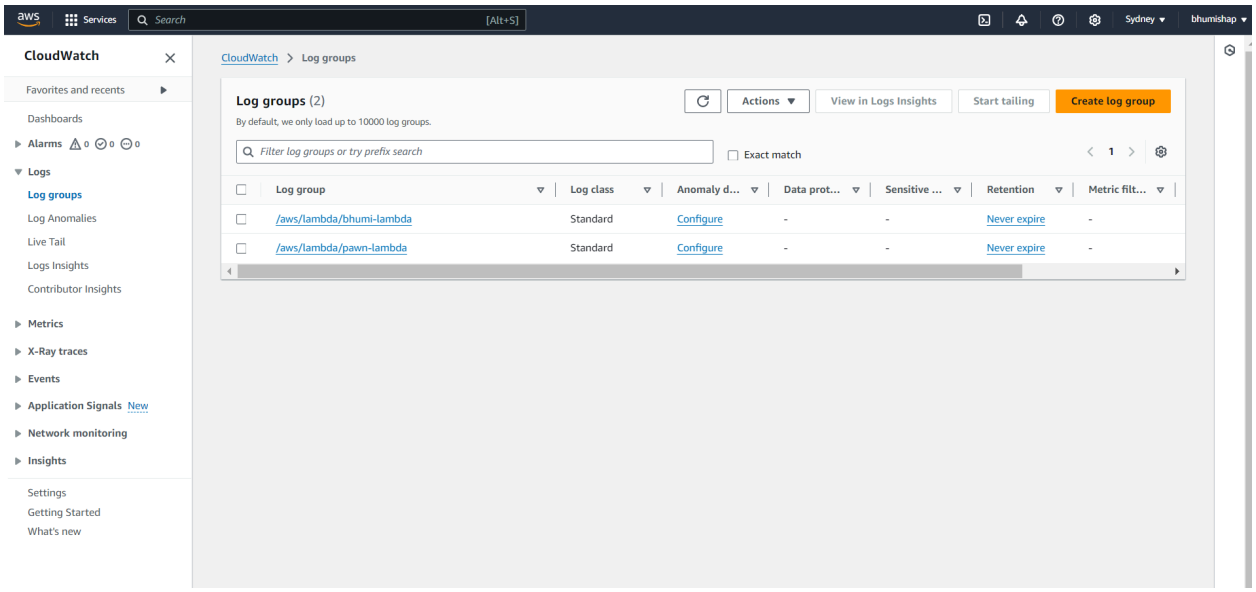

Upload Info
Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more
Drag and drop files and folders you want to upload here, or choose Add files or Add folder.
Files and folders (1 Total, 77.3 KB)
All files and folders in this table will be uploaded.
Remove
Add files
Add folder
Find by name
1
Name
Folder
Screenshot 2024-09-10 220423.png
-
Destination Info
Destination
s3://pawn
Destination details
Bucket settings that impact new objects stored in the specified destination.
Permissions
Grant public access and access to other AWS accounts.

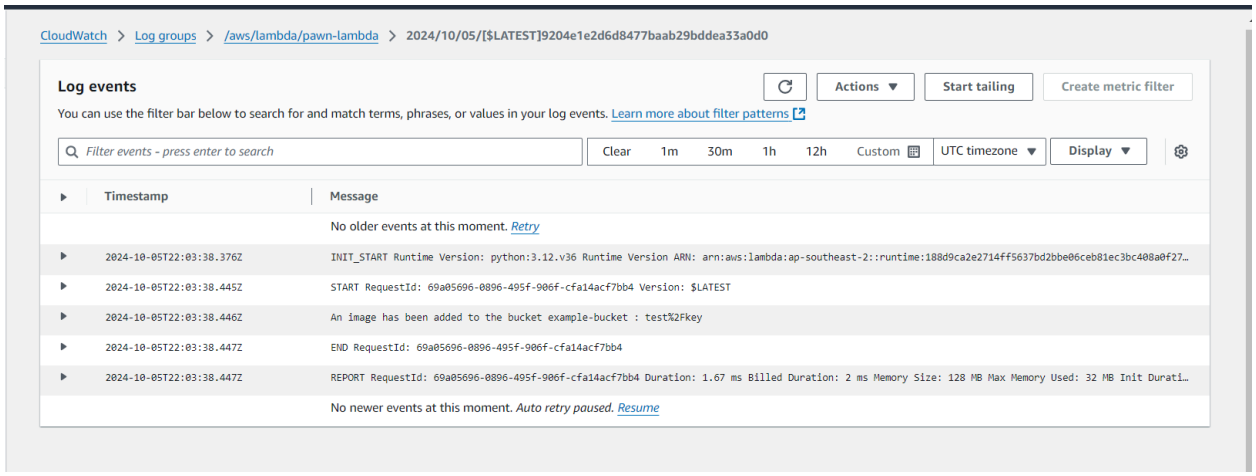


10. Now click on the test in lambda to check whether it is giving log when image is added to S3.



11. Now lets see the log on Cloud watch.To see it go to monitor section and then click on view cloudwatch logs.





Conclusion: In this experiment, we successfully created an AWS Lambda function that logs a message when an image is uploaded to an S3 bucket. It is important to note that we have to select the S3-put template in the event otherwise code will give an error.The function was successfully triggered by S3 object uploads, validating the functionality of Lambda's event-driven architecture. This experiment demonstrated how Lambda can efficiently respond to S3 events and how to troubleshoot common issues with event structure.