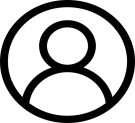
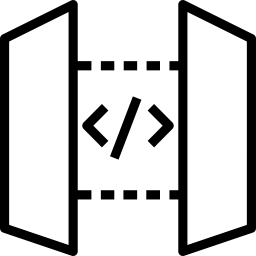
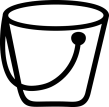
**Uploading Files To S3 Using API Gateway**

In this project, the goal is to upload a binary file to AWS S3 through an application programming interface. The whole idea behind this is to upload an object to AWS S3 automatically through a serverless application and in this case AWS Lambda. The Lambda function will be triggered by an application programming interface (API) and then later we can test the endpoint using a third-party application (Postman).

The common use case for this type of architecture is when an object needs to be uploaded to the bucket programmatically without interacting with the AWS management console.

**Architecture**

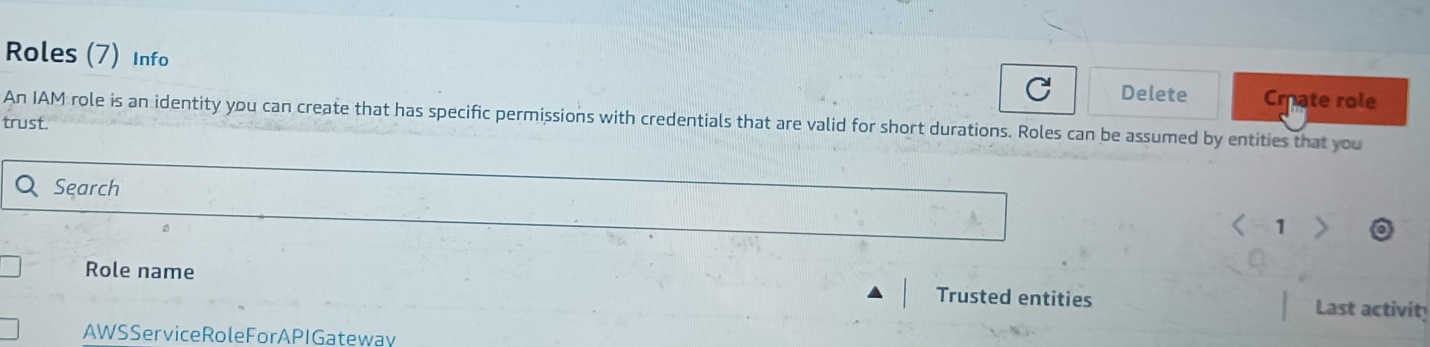
**   **

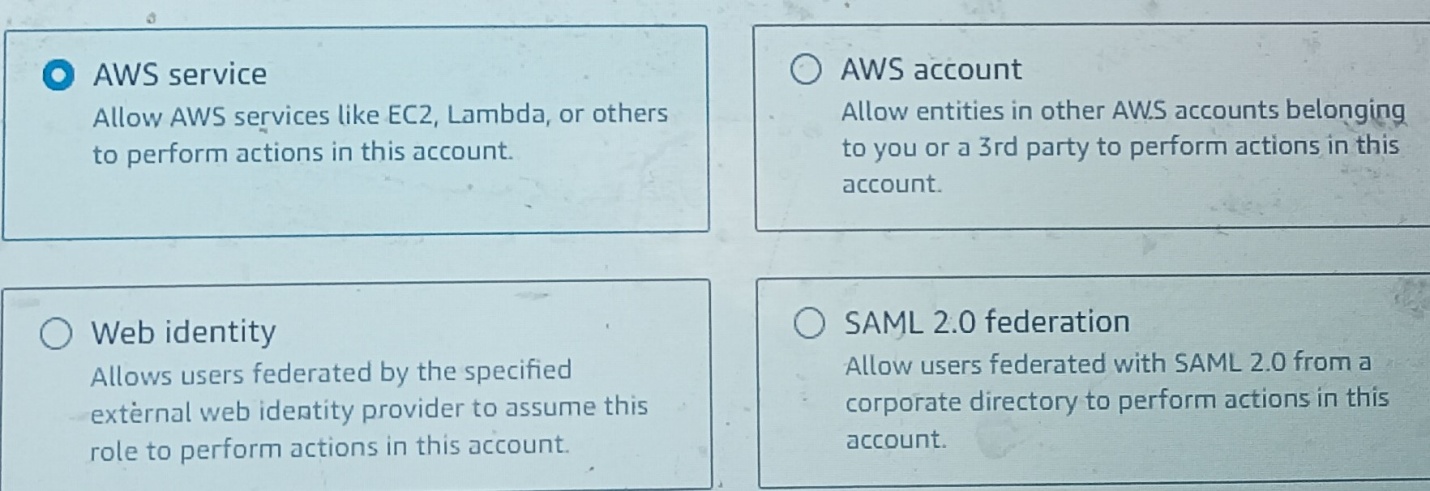
**User Api Gateway Lambda S3**

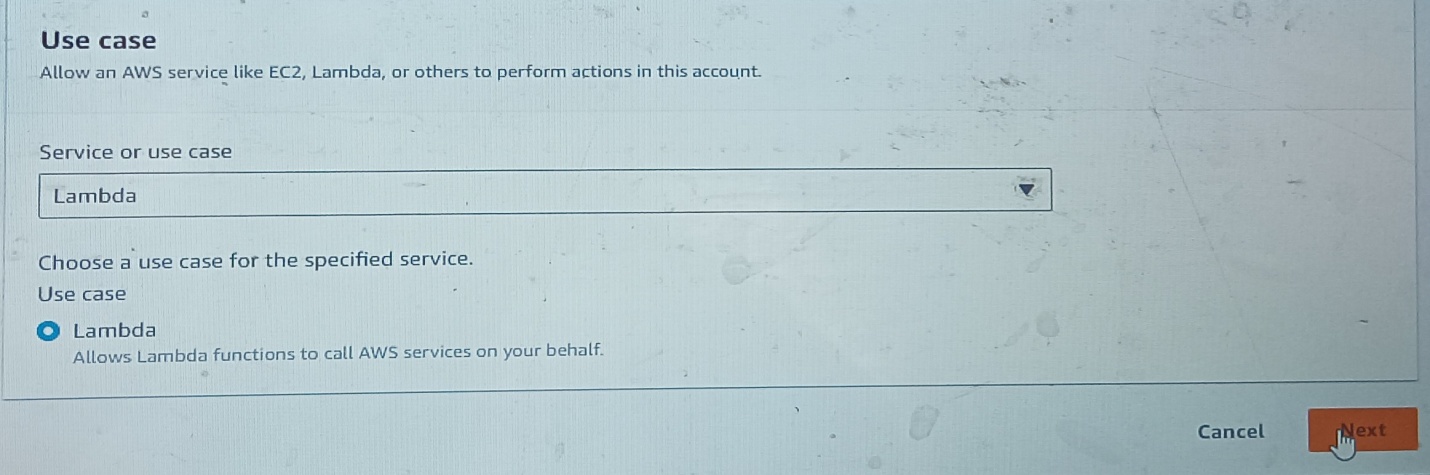
**1.Create an IAM Role**

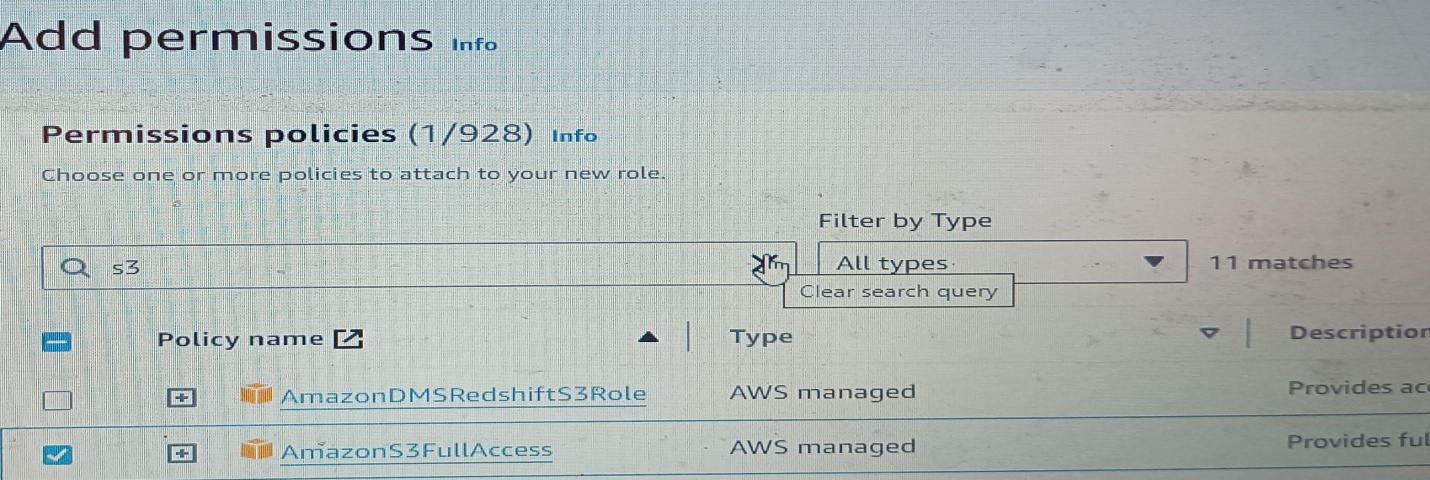
The first step is to create an IAM role for Lambda that will have the right permission to write the data in S3 using Api gateway.

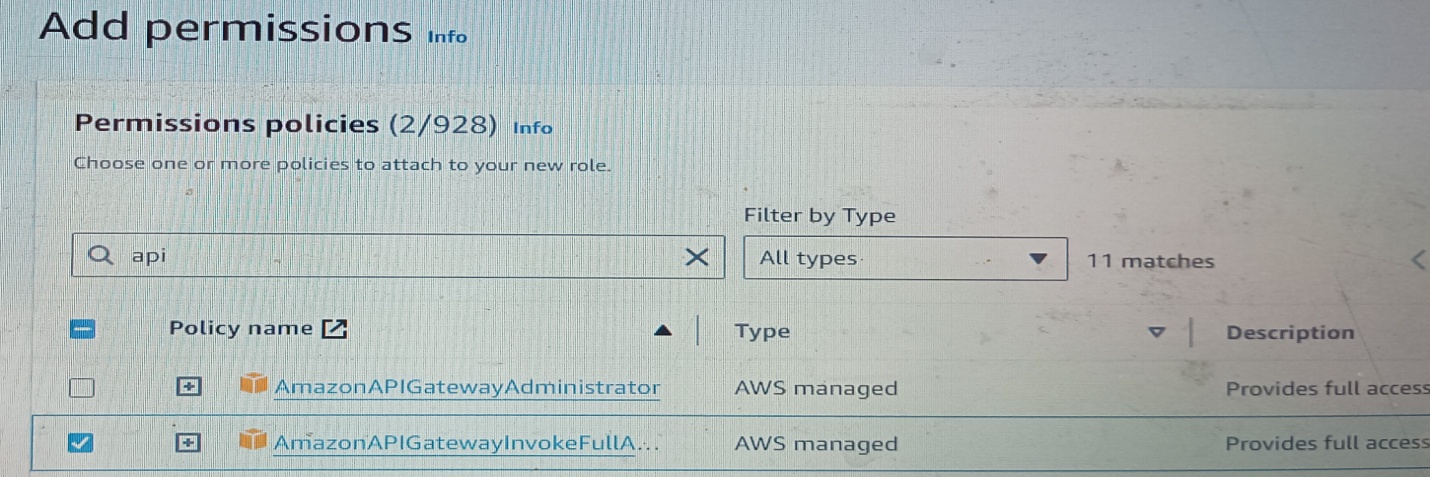
Navigate to the search bar to search for the IAM role. Click on create role Select Lambda as the use case attach the policies to the role i.e, S3 full access,Api gateway invoke full access. Name the role Create the role.

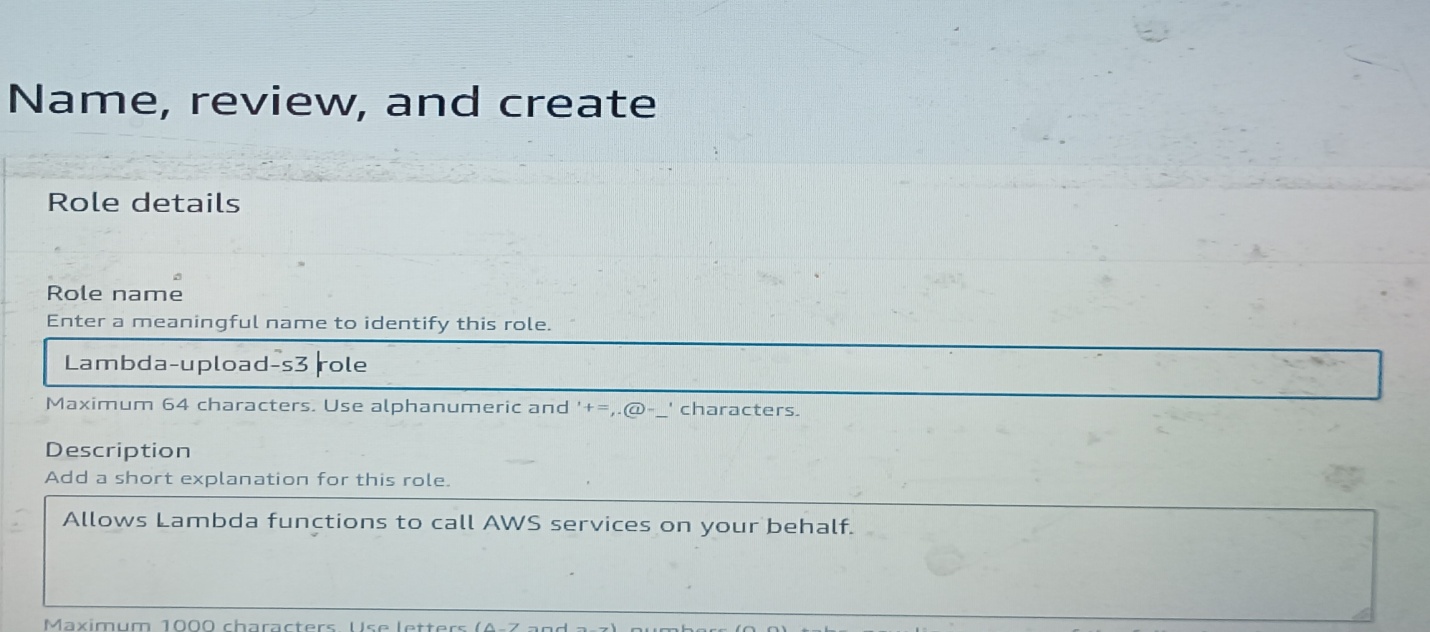








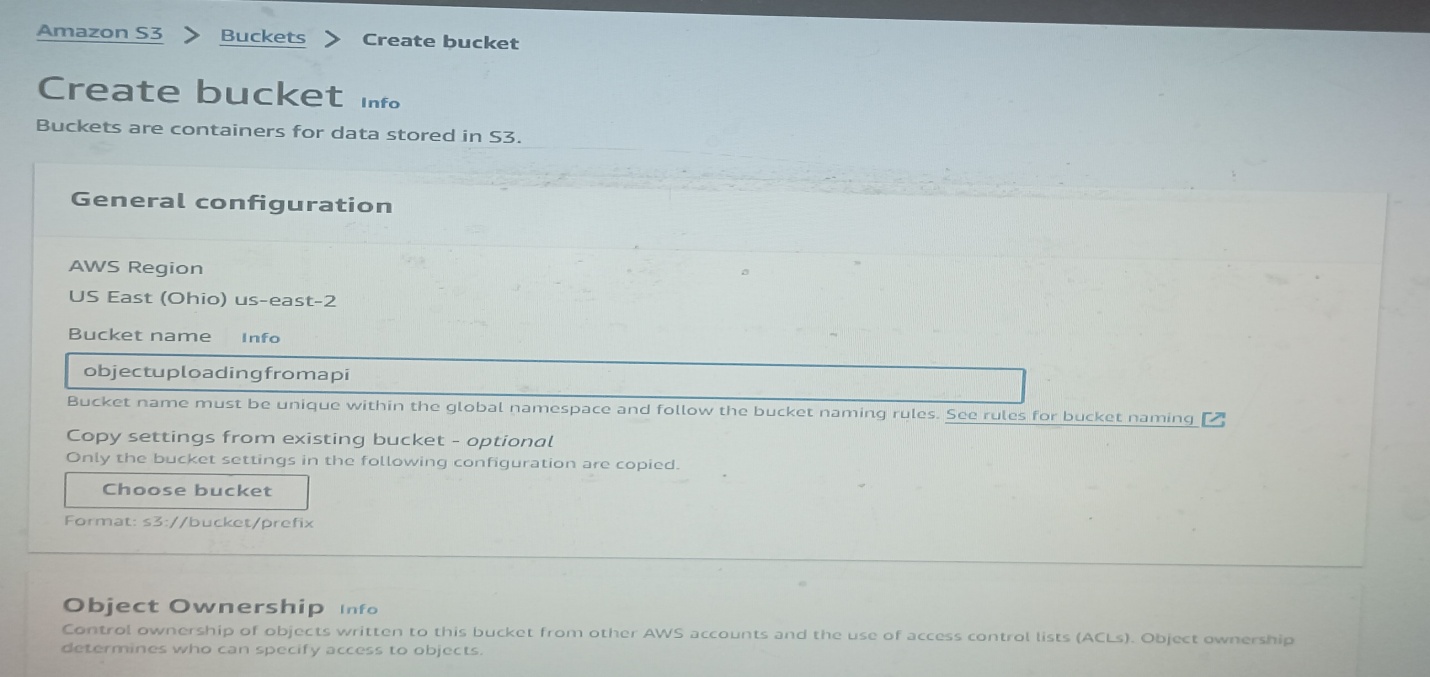




**2.Create an S3 Bucket**

The next step is to create a bucket that will serve as a destination for the file upload.

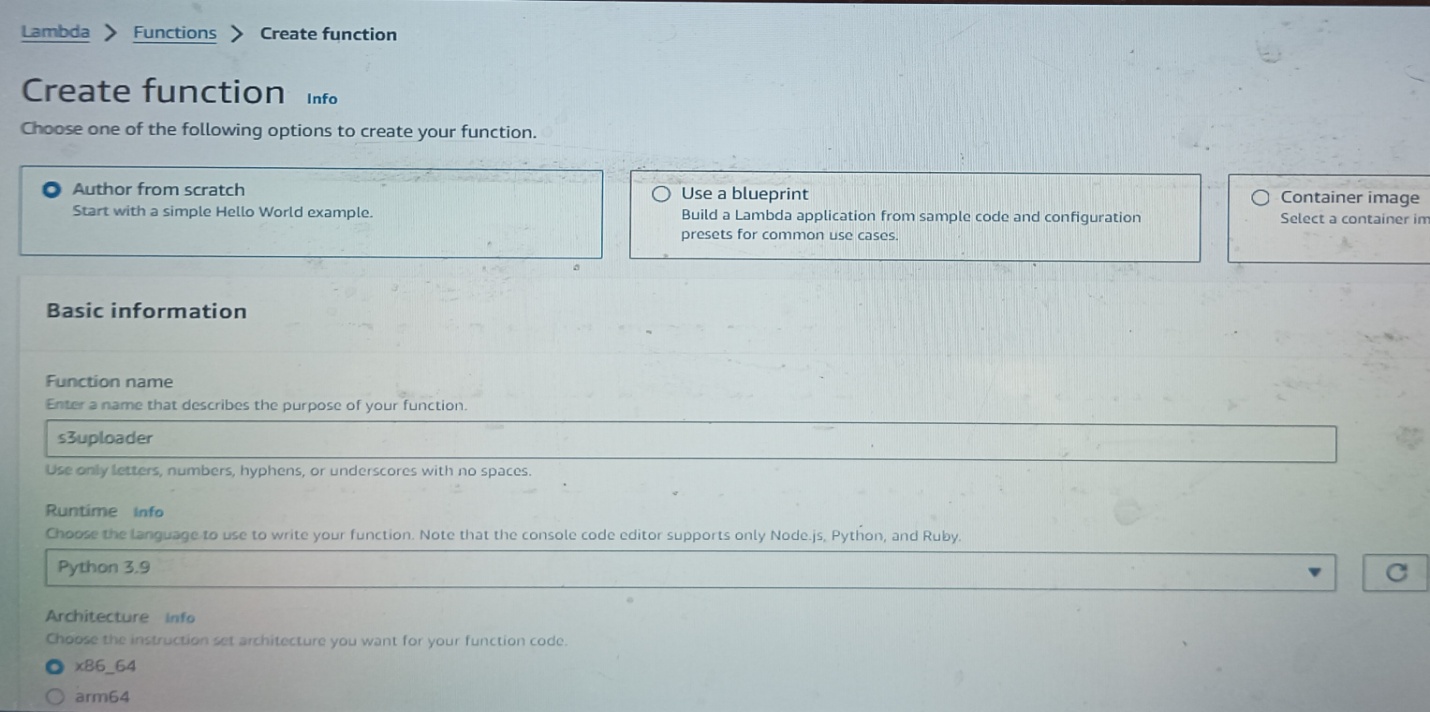
Go to S3 Click on create bucket Name the bucket (keep otherthings default) Create bucket.

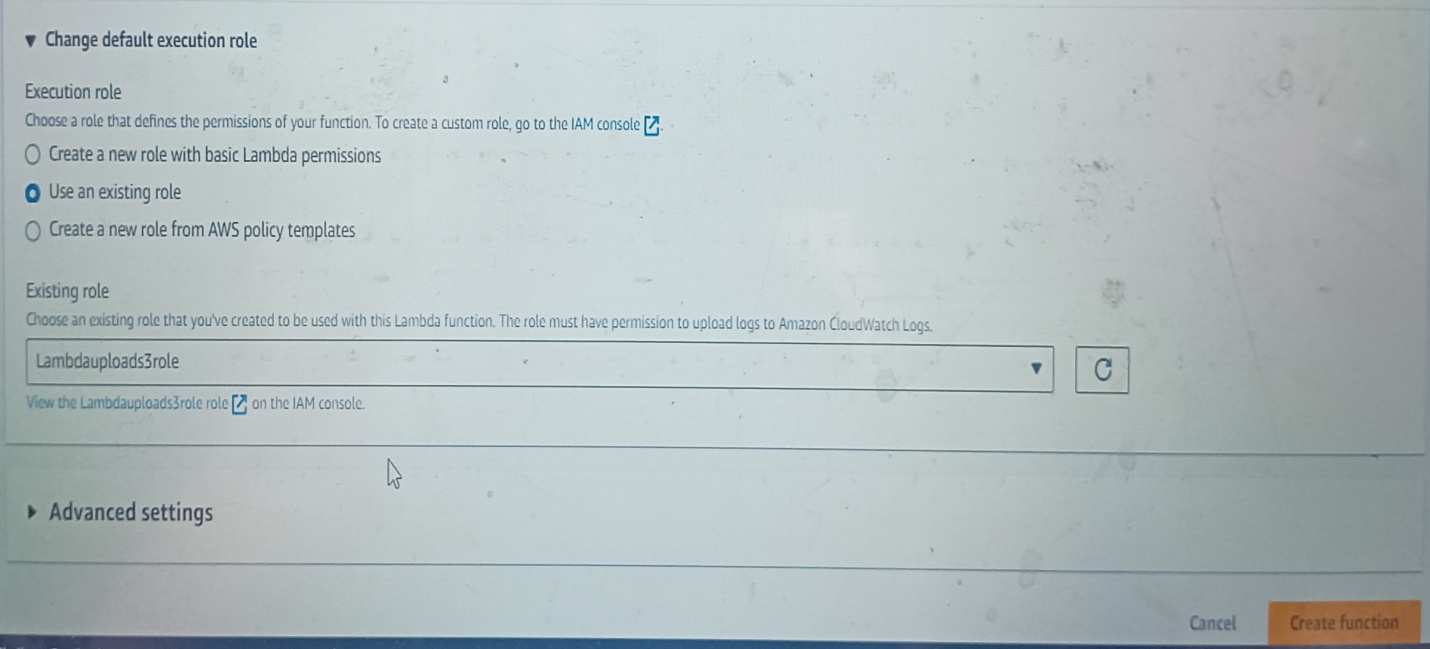


**3.Create a Lambda Function**

Go to Lambda click functions and create a function. Select the Author from Scratch option and input the basic information required. For runtime choose python3.9.

Add permissions to the function by changing the default execution role to an existing role. In this case choose existing role i.e, the role we created at the start. Then click create function.



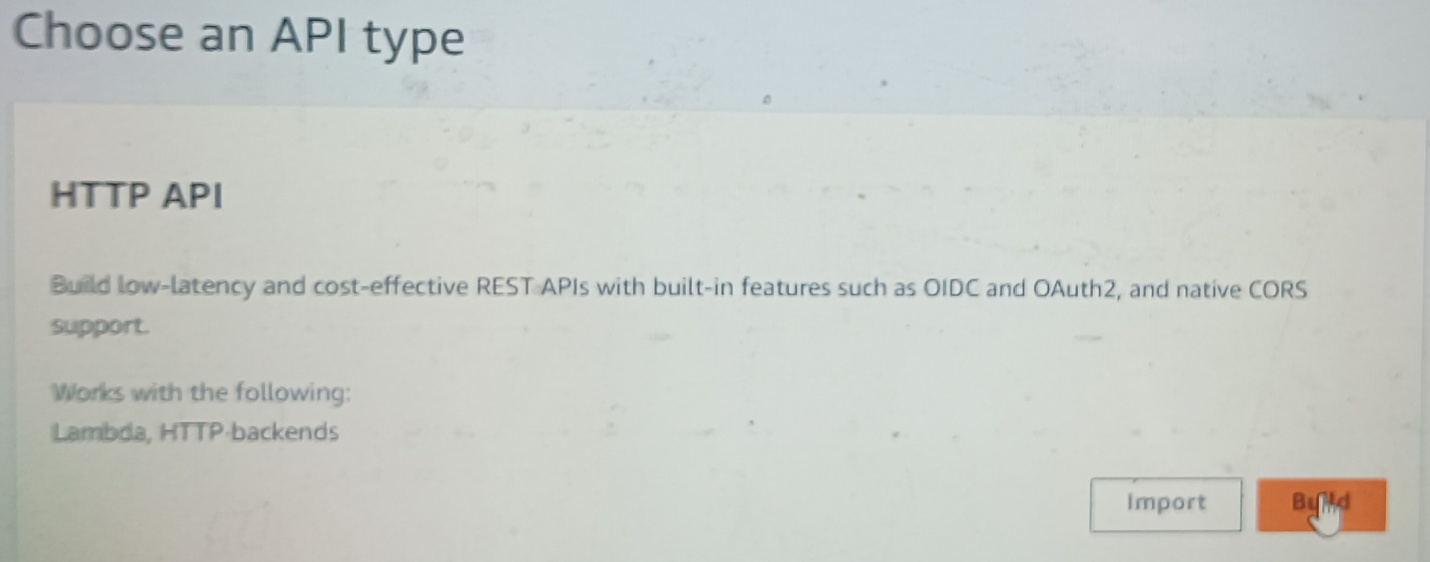


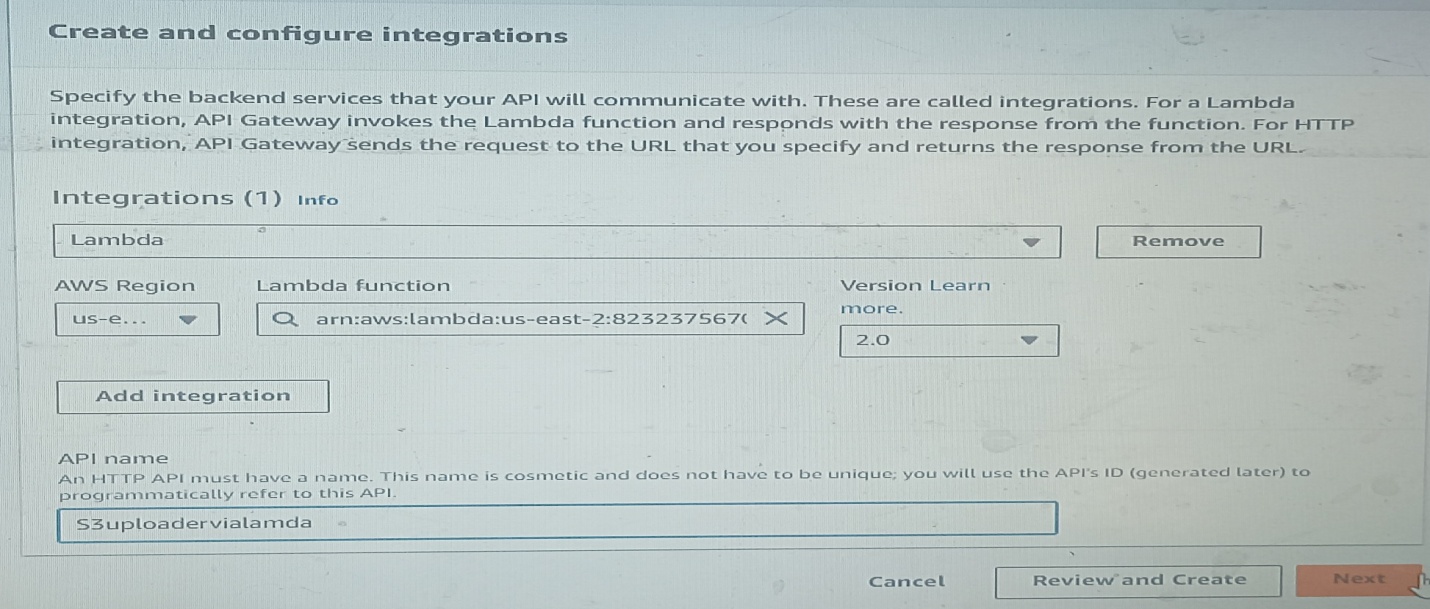
**4.Create API**

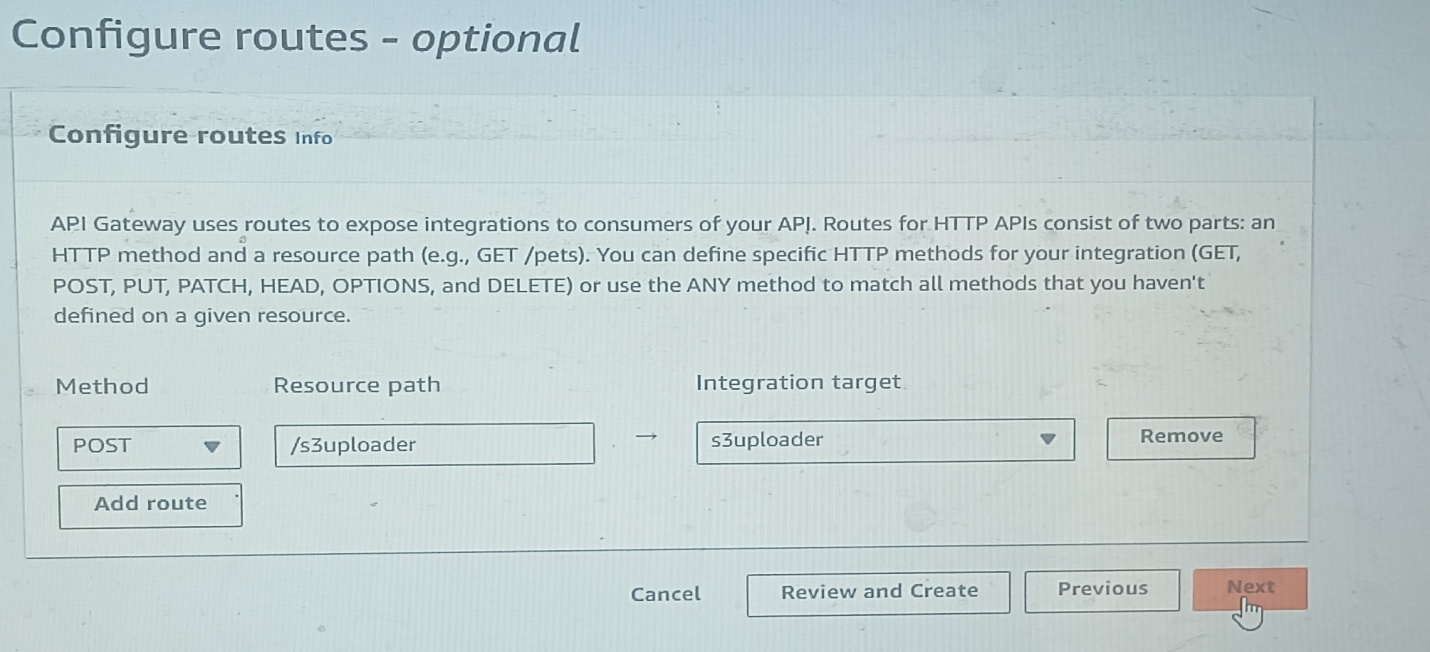
Navigate to the search bar to search for API Gateway. Click on create API. Choose an API type. The type of API we are using is HTTP API Integration Lambda( choose lambda function that we have created ) Click on Next Method ‘POST’ . Click on Next. Stage ‘Default’ Then create API.

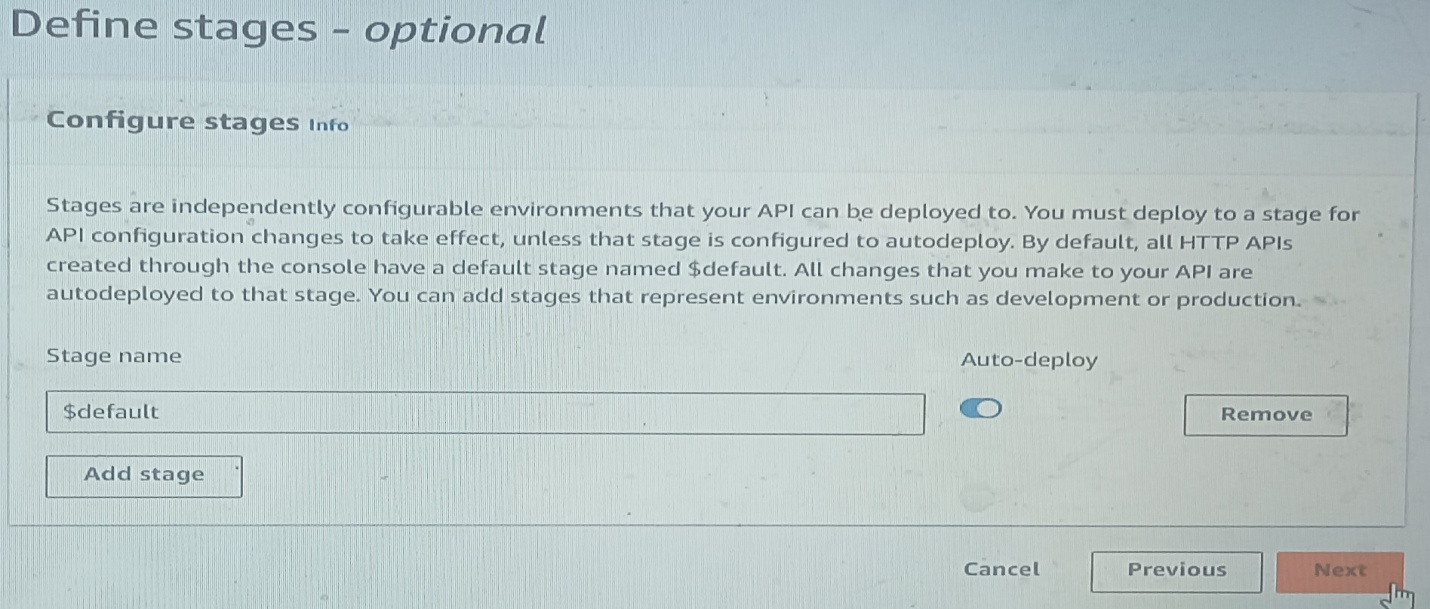
Copy the Invoke URL (endpoint) and resource path paste it somewhere (Notepad) we need that later.

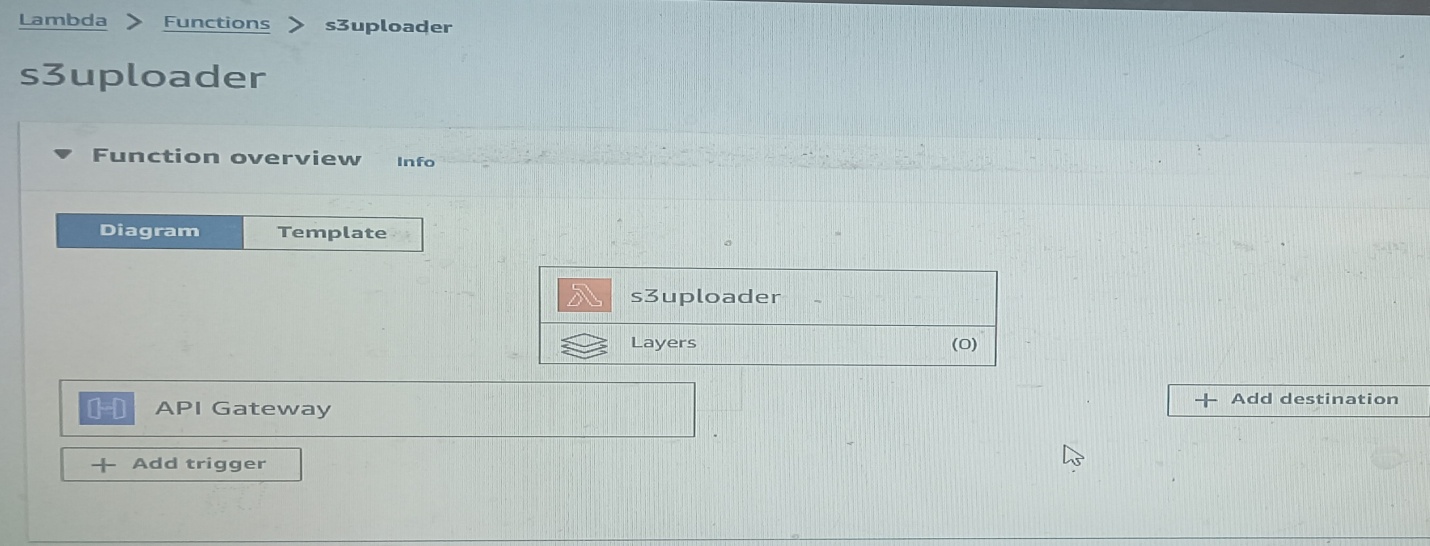
IF you check your Lambda function A triggered has been added.







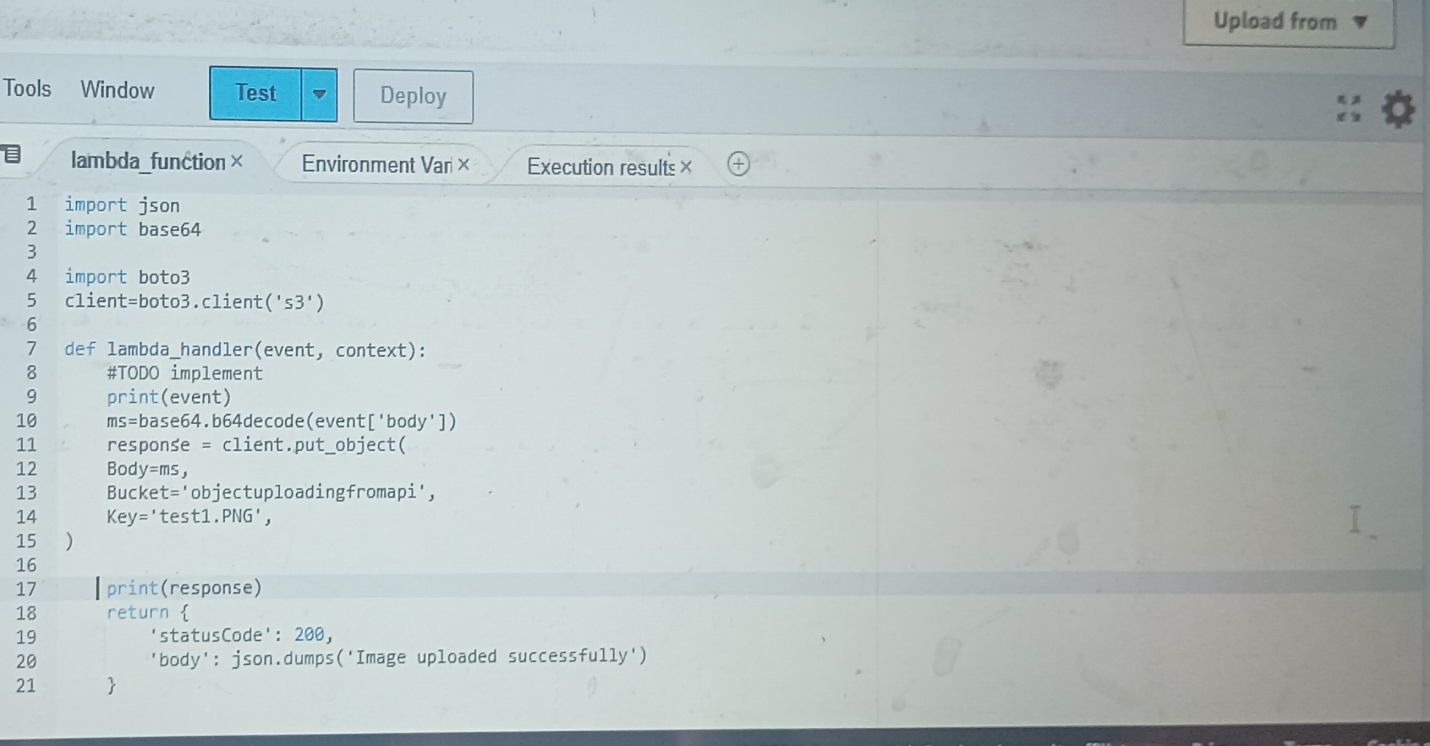




**5.The next step is to write a python script that will put a binary file in the destination bucket when the lambda function is being triggered by the API Gateway.**

Go to Lambda function. Code Put the code and deploy it .

(In the code update your S3 bucket name).



Name of your S3 bucket

**For code refer “source code” folder.**

Now let’s test the Endpoint.

**Testing the Endpoint with Postman**

a.    Open the Postman app or postman console. click on the ‘APIs’ tab from the left panel.

b.    Click on the (+) sign to open a request tab paste the Invoke URL endpoint into the request tab. Change the method to ‘POST’.

c.  Click on the ‘Body’. Select binary. This is because we are uploading a binary file. Upload a PNG file from your local machine to test the endpoint. Click send.

d. Add the resource path (/upload) to the endpoint

e. Lastly, if we check our destination bucket, we should see the file in the bucket.

