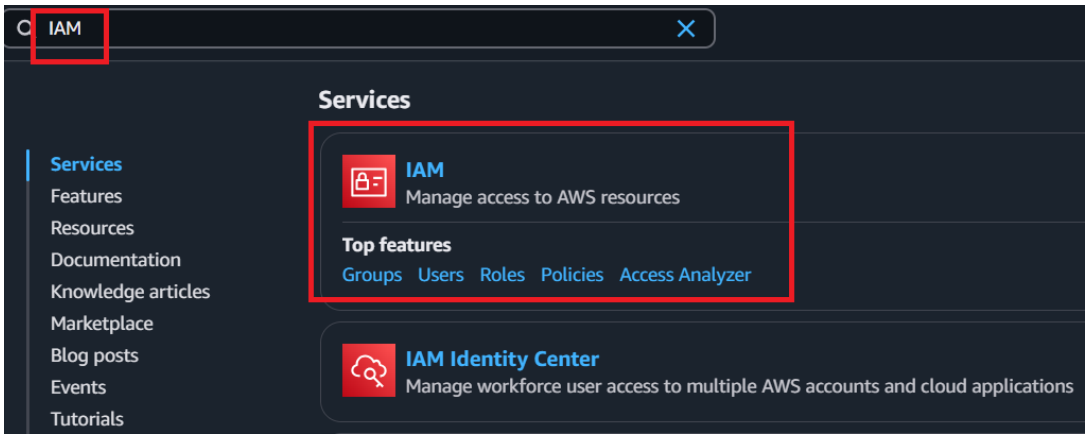
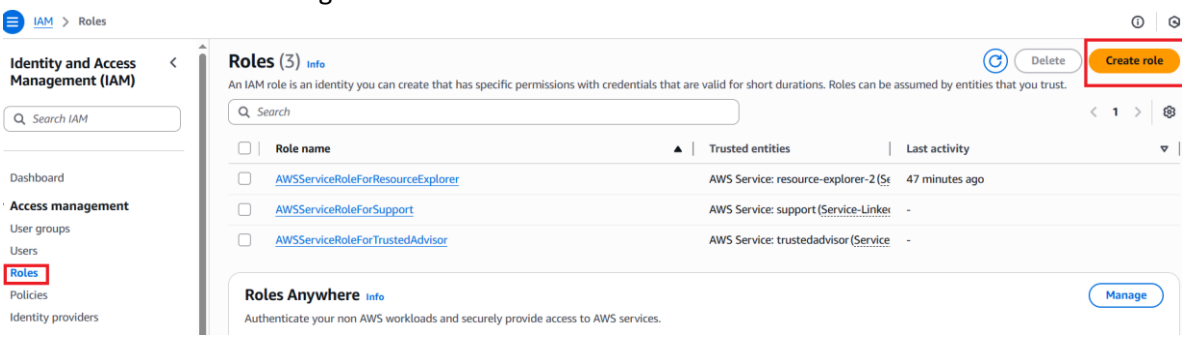
CREATE EVERY AWS SERVICE IN Asia Pacific(Mumbai)

1. Setup the IAM Roles in AWS.

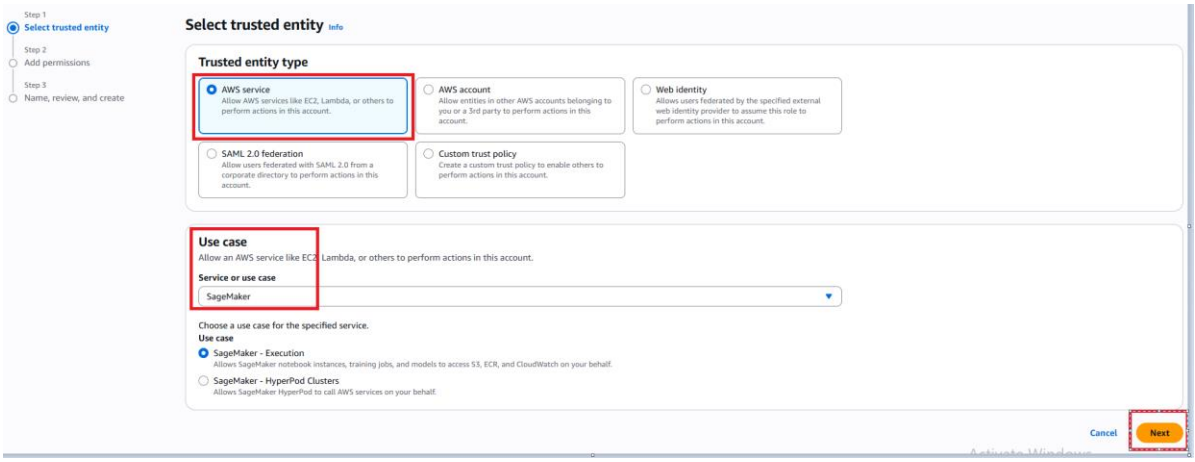
a. Search for IAM

i. 

b. On the left side we can see the “Roles” click on this. We will create the Role here. This roles we will add in sagemaker and in Lambda. To create click on “Create Role”



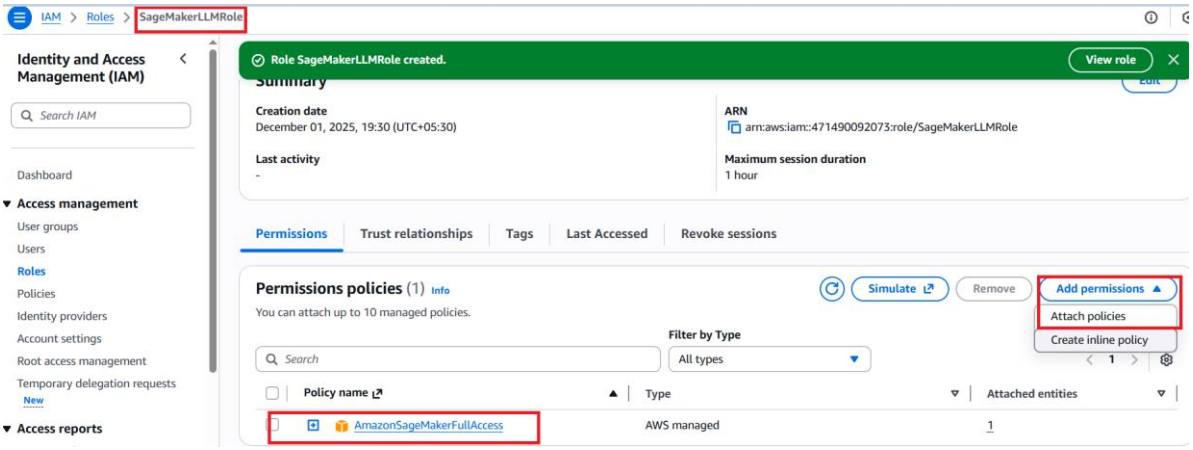
c. Choose “AWS Service” then under the “Usecase”. Search for the “SageMaker” then click on “Next”.



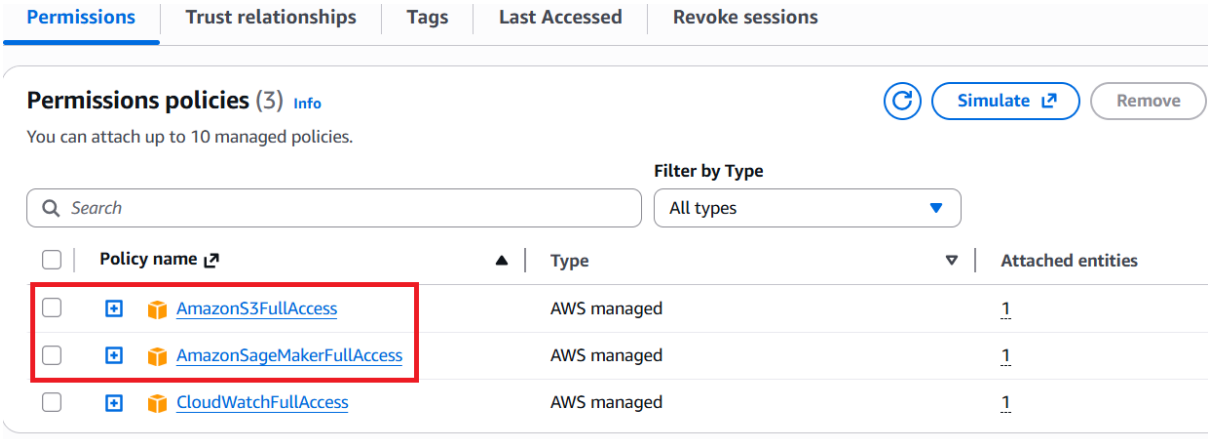
d. Again click on “Next”.

e. We have to write the “Role name” then finally click on “Create Role” then we can see the Role which was created. Click on that Role.

f. AsofNow we can see only one policy i.e, “AmazonSageMakerFullAccess”. Now we will add permissions here.To Add Goto “Add permissions”=>”Click on Attach Policies”. If we want to add any custom policies we will go with “Create incline policies”



g. Add “CloudWatchFullAccess” and “AmazonS3FullAccess” these 2policies then finally click on “Add permissions”.Now you should see these 2more policies under your role.

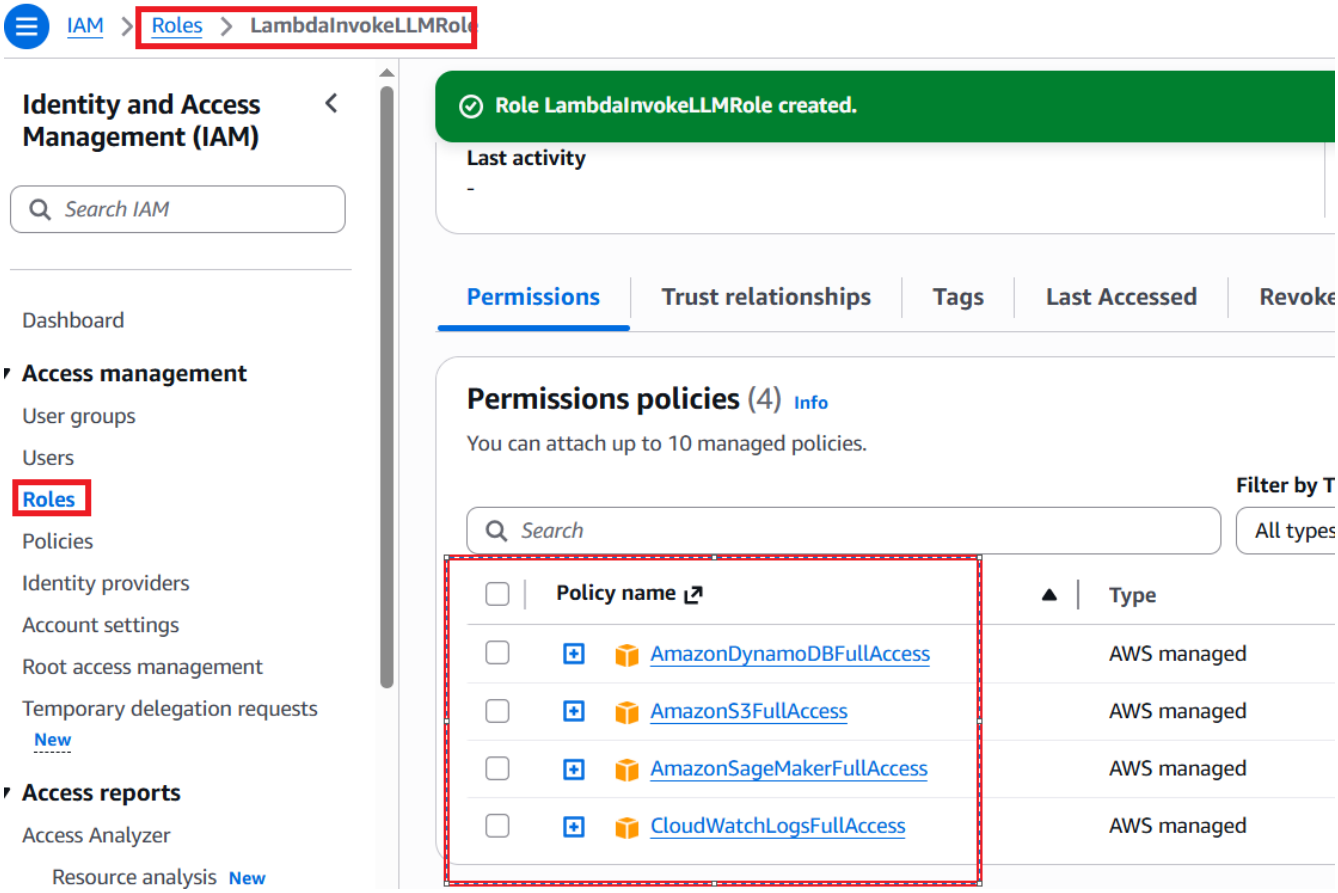


2. Now we will setup or create the “Lambda Role”.

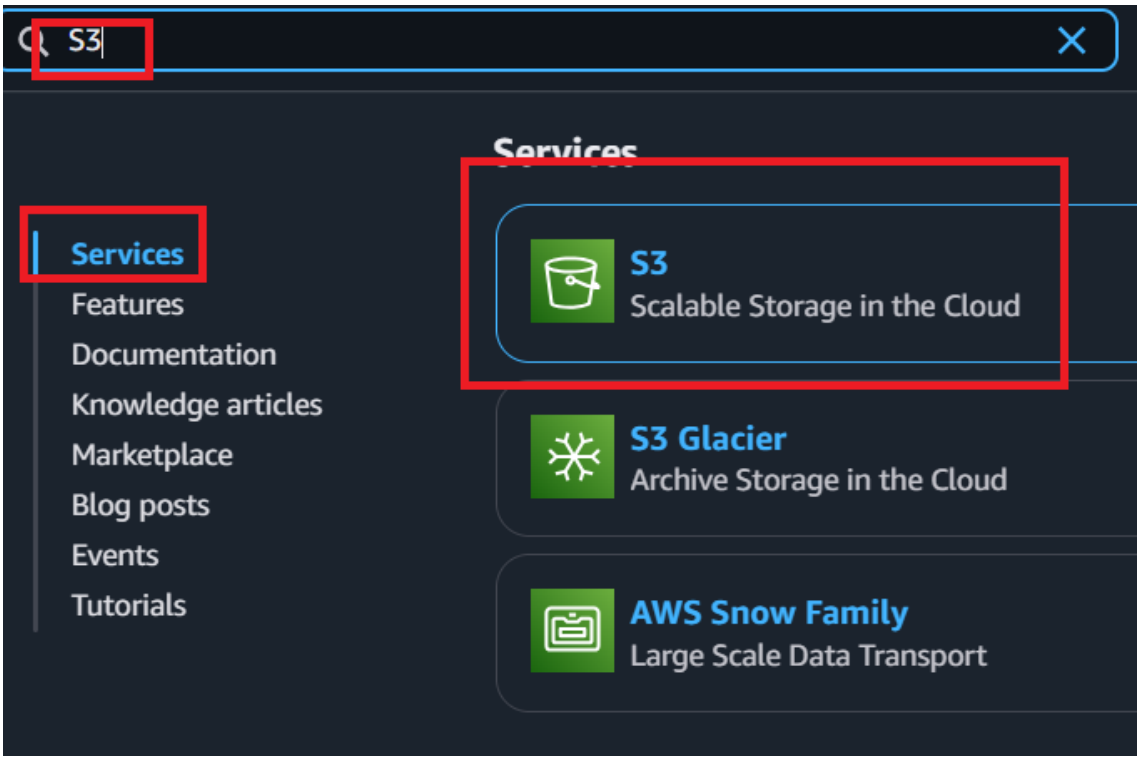
a. Goto Roles=>Click on “Create Role”.
b. Select the “AWS service”. Under “usecase ” search the “Lambda” then select the “Lambda” then click on “Next”.
c. Now we will add below policies here.

   i. AmazonSageMakerFullAccess
   ii. AmazonDynamoDBFullAccess
   iii. AmazonS3FullAccess
   iv. CloudWatchLogsFullAccess

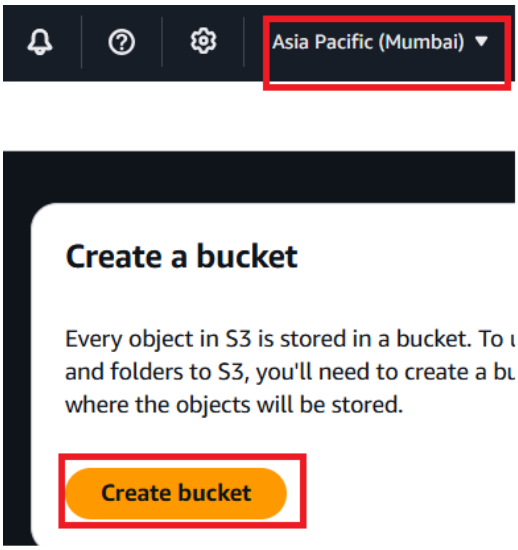
d. Click on “Next”
e. Finally give any “Role name” then click on “Create role”.



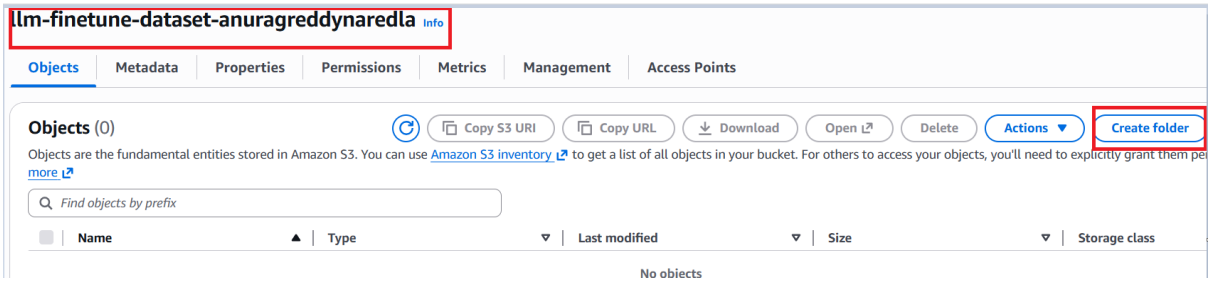
3. **Create the S3bucket for keep the data and LLM finetuned Model.**



a. Select the “Asia Pacific (Mumbai)” then click on “Create bucket”.



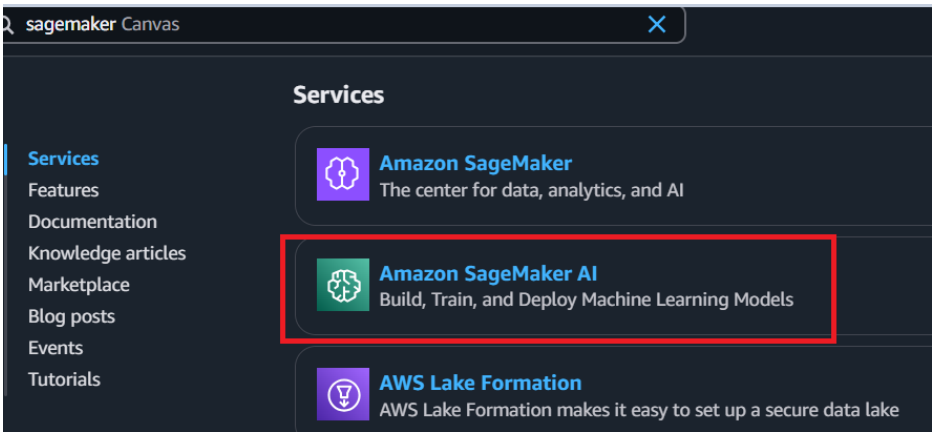
b. Under Bucket name give name like this : llm-finetune-dataset-<yourname>.
c. Keep everything same then finally click on “Create bucket”.
d. Under this bucket we will create a folder to create select the created bucket name then click on “Create folder”. Then give the name as “dataset”.



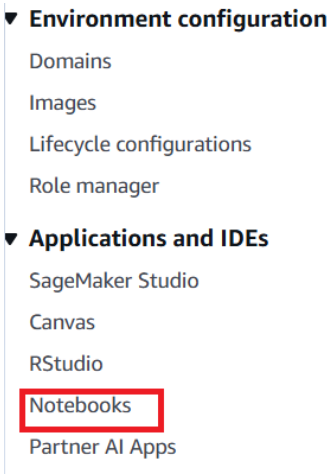
e. Now we will create one more bucket for storing the Finetuned LLM model.
f. Click on “Create bucket”=> Give the name like this: llm-model-artifacts-<yourname> then click on Create bucket Now we will create one more folder for storing the Models with this name i.e, “models”.

4. Now we will create the “AWSSageMakerInstance”

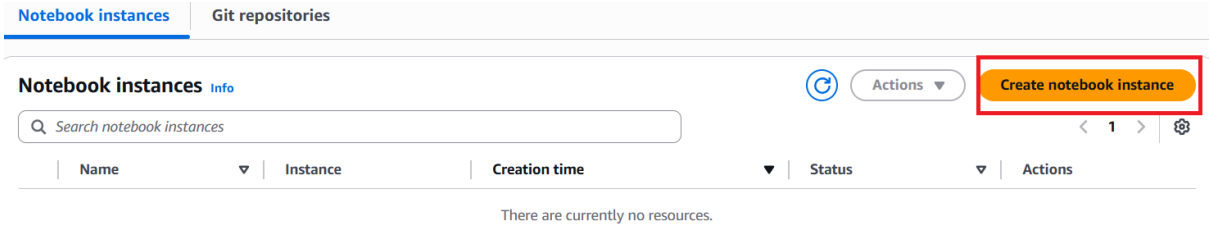
a. In the sagemaker we will setup the instance.
b. Search the sagemaker and select the “SageMaker AI”.



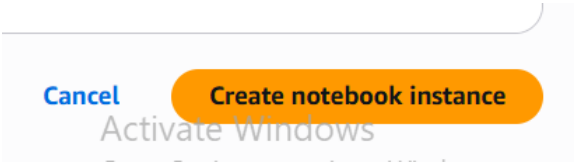
c. On the left side we can see the “Notebooks”. Click on this notebooks.



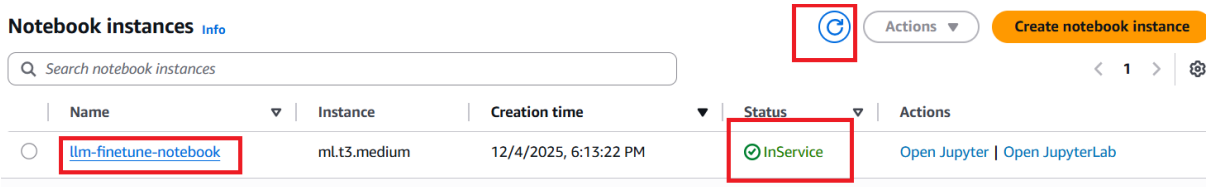
d. Now click on “Create notebook instance”.



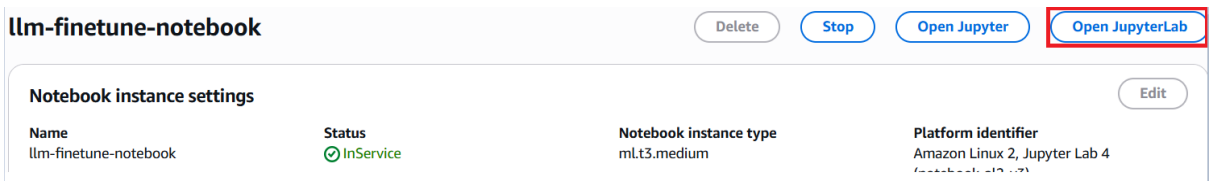
e. Under the “Notebook instance settings” give any name under “Notebook instance name”. keep remaining asit is then click on “Create notebook instance” which will be displayed down.



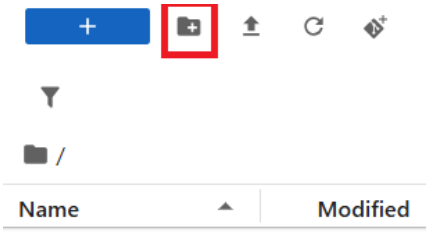
f. After creating notebook instance give sometime to get the status as “InService” to get this click on refresh button.



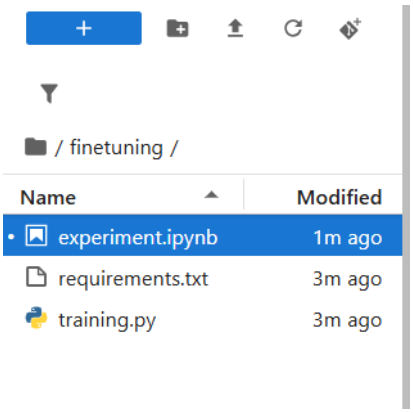
g. Click on the created notebook name Now we can see the option “open jupyterlab” with this we can get the terminal aswell.with the otherone we will not get terminal.



h. Now we will create the folder by click on “+” icon on top left



1. Create the folder “finetuning”
2. Open the folder to open double click on that folder
3. Now create file by rightclick “Newfile” give the name as training.py and create one more file i.e, requirements.txt
4. We will create New notebook by clicking on “New Notebook” then select the “conda_python3” as the environment.



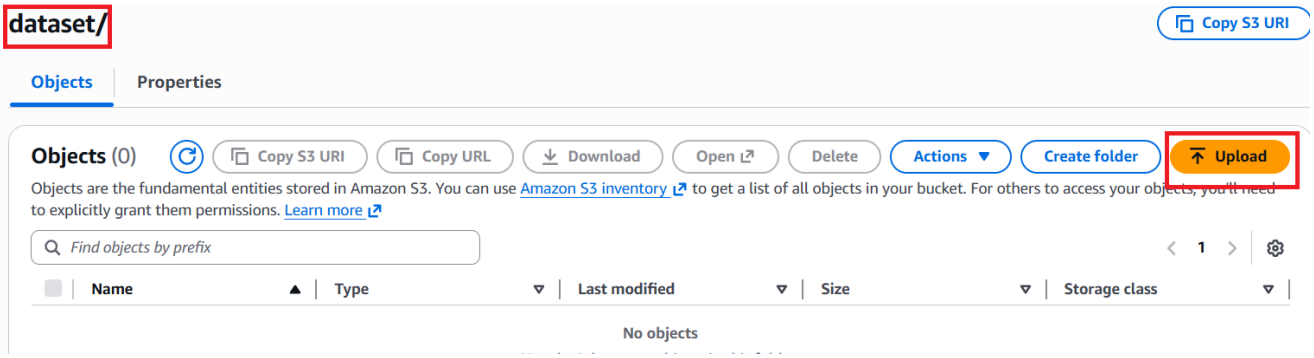
5. Now we will open the terminal type below commands.
   a. ls

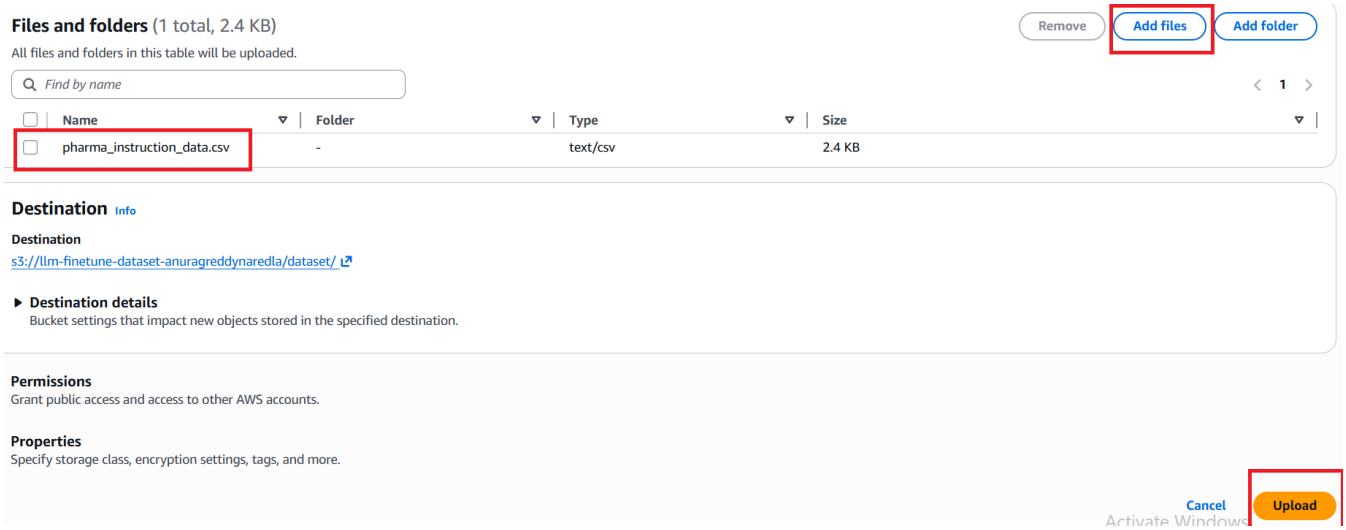
5. Now we will write the code in these files.

a. https://github.com/Anuragreddy-Naredla/Finetuning-on-aws/

i. Here I will do the Instruct finetuning here which means I will take the model on top of this model I will perform the InstructionFineTuning.

ii. Put the data on S3 to put the data Go to S3bucket click on the created S3bucket name the click on the “dataset/” folder then click on the upload button then click on “Add files” button give the “pharma_instruction_data.csv” take this .csv file from my github then click on the checkbox then finally click on “upload”.





iii. Take the code and run it in sagemaker

iv.create virtual environment.

a. conda create -n hf python=3.10 -y
b. conda activate hf
c. pip install –r requirements.txt
   OR
a. Type the “which python” command in terminal in amazon sagemaker notebook
b. Take the path and type “source <GIVE YOUR PATH HERE>”

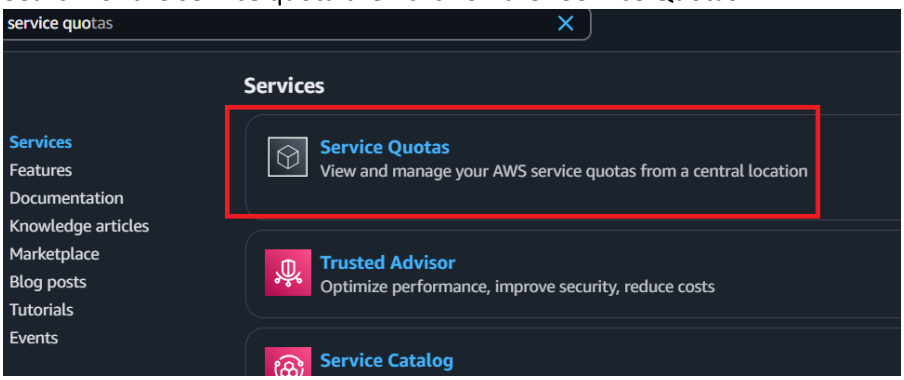
   source /home/ec2-user/anaconda3/bin/activate python3

c. pip install –r requirements.txt

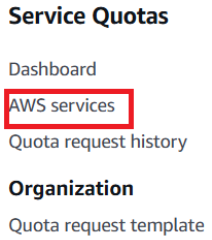
v. Run “experiments.ipynb” from finetuning folder. Here we will get the error which means My sagemaker doesnot allow to train the models inside the SageMaker notebook. Due to this we will take the diff approach for this diff approach we will use the “estimator_launcher.ipynb” and “train.py” which means we will load this model inside the container

vi. Run the “estimator_launcher.ipynb” file before running this file perform below steps.

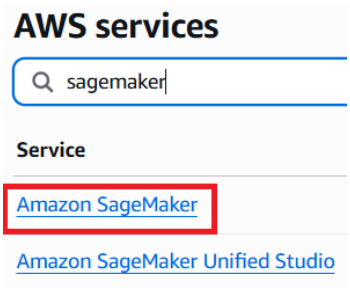
1. Search for the service quota then click on the “Service Quotas”.



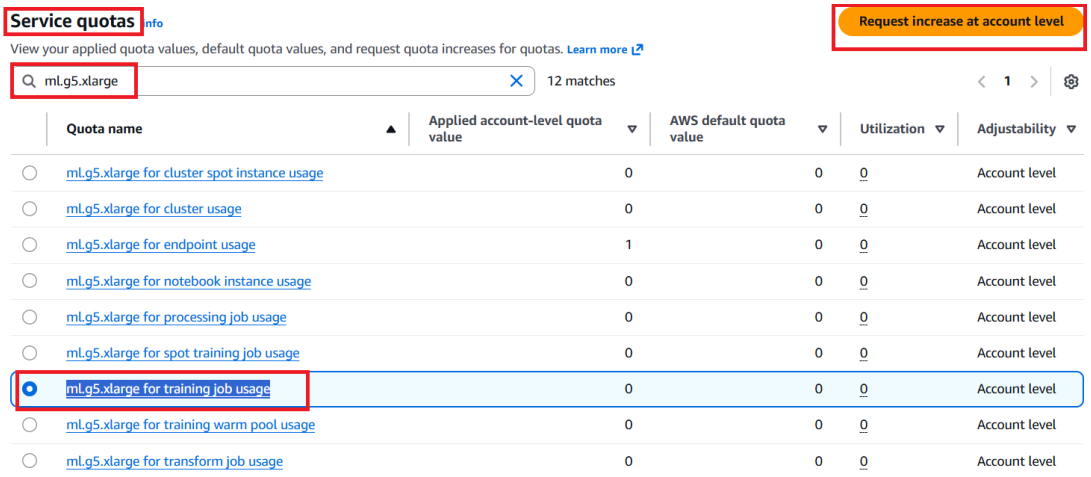
2. On top left click on “AWS services”.

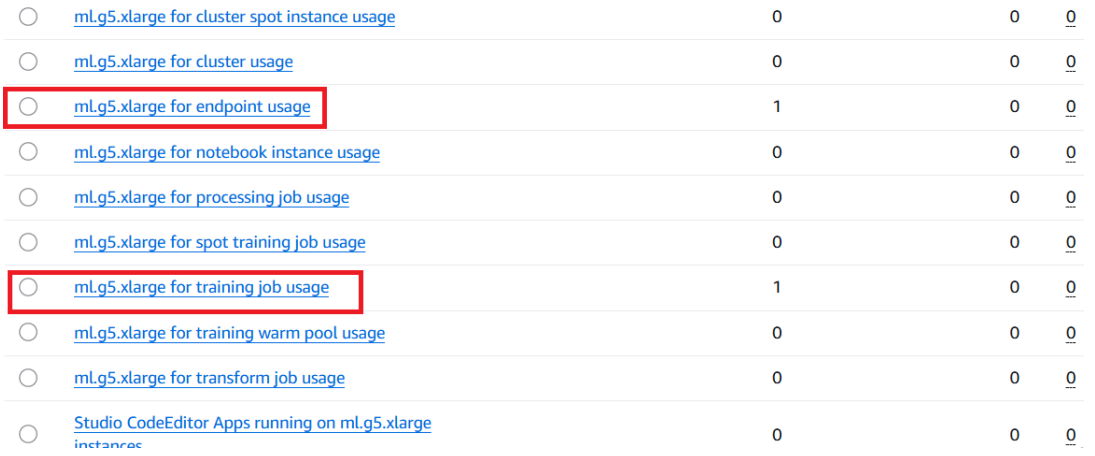


3. Search for sagemaker then click on the “Amazon SageMaker”.

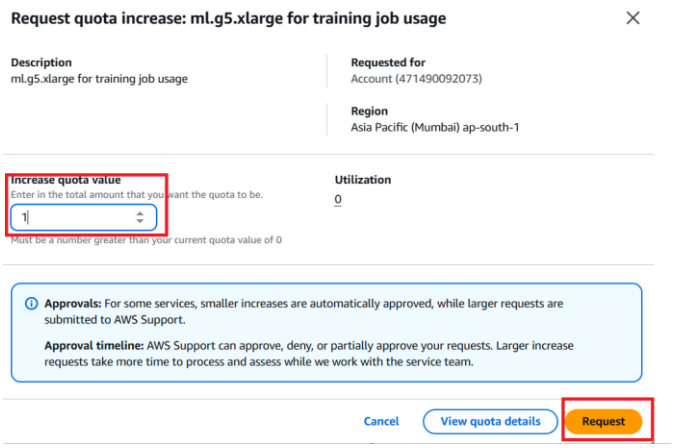


4. Select the “ml.g5.xlarge for training job usage” and “ml.g5.xlarge for endpoint usage” then click on “Request increase at account level”.

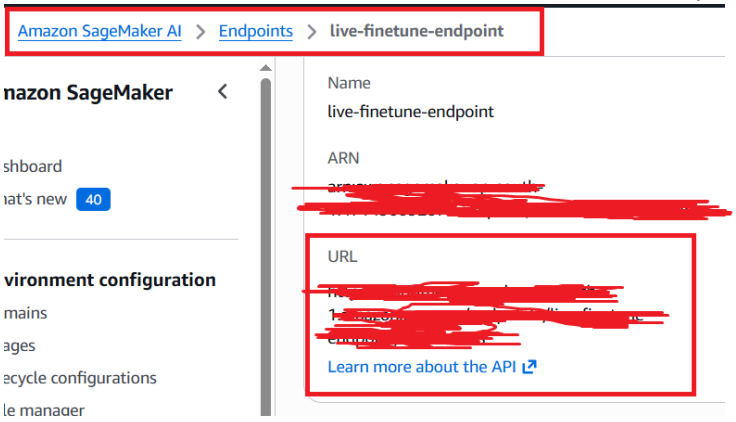




5. Under increase quota value give 1 then click on “Request”



vii. Now run the estimator_laucnher.ipynb notebook we will get the finetuned model which will be saved in the s3bucket then we will deploy it.



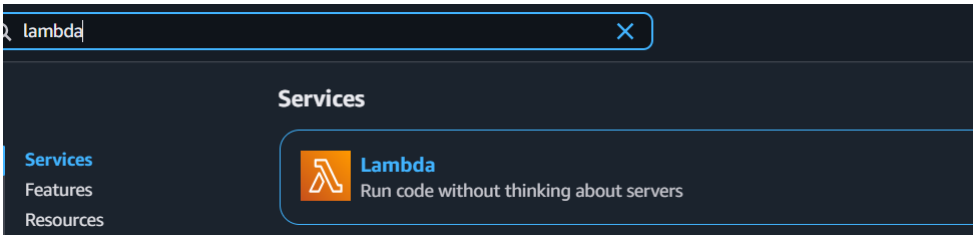
COPY THIS URL AND PASTE IT IN SOMEWHERE IN NOTEPAD

6.Deployment (We will use the Lambda and APIGateWay we will create endpoint here then we will use DynamoDB)

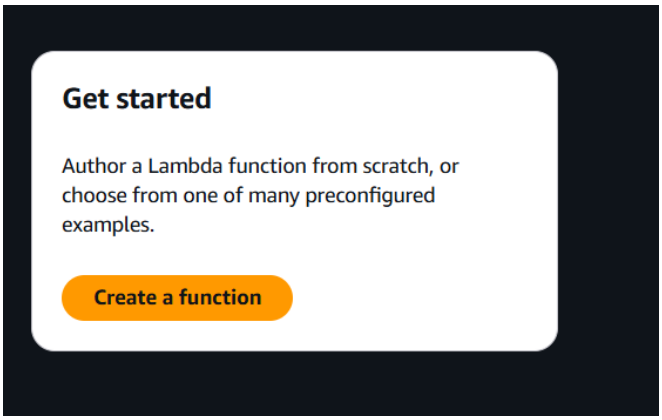
a. Above trained model we will deploy it.

b. Create the Lambda

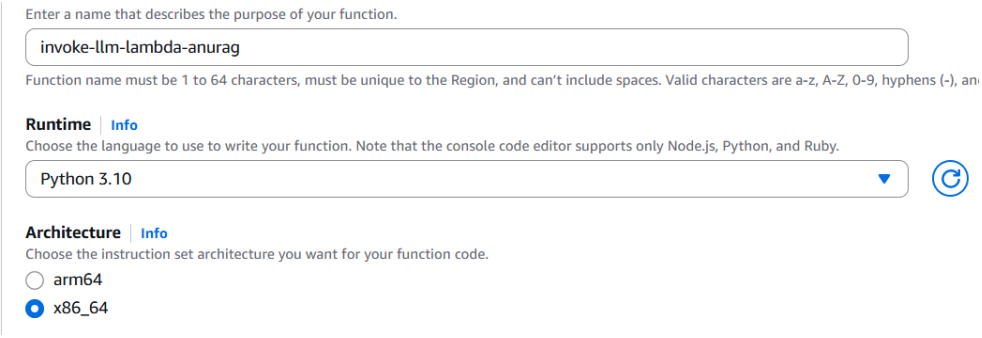
i. Search in console then click on lambda



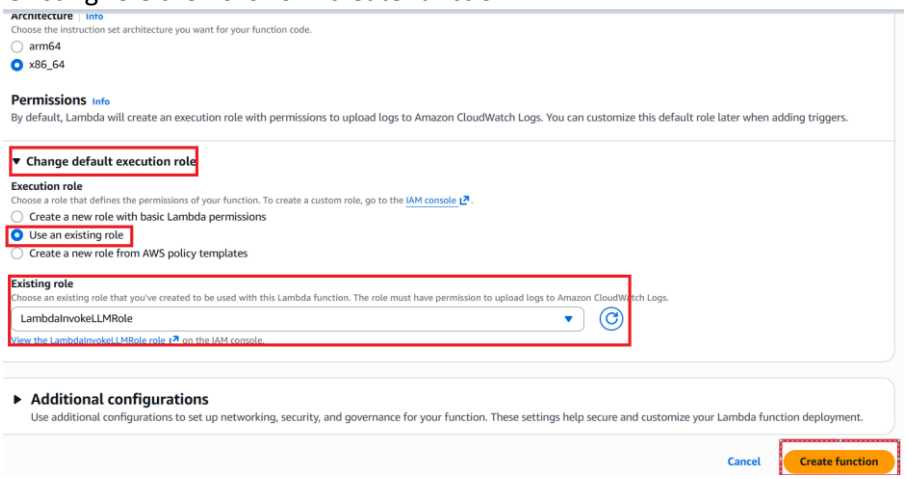
ii. Now we will create a function. Click on it



iii. Give the function name, under Python select the python 3.10

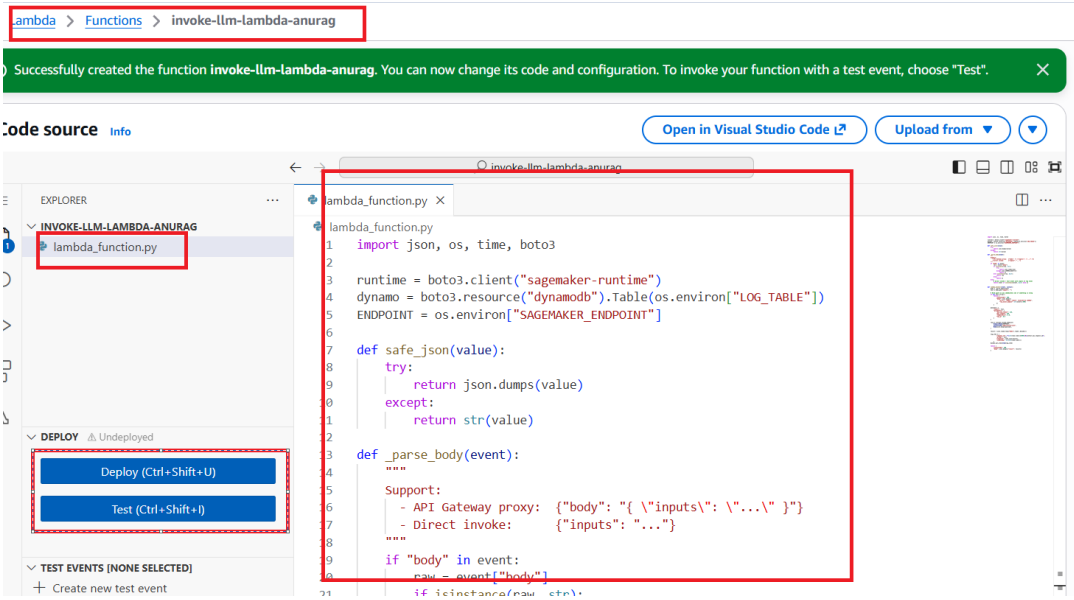


iv. We have to add the Roles to create click on "Change default execution role". Givethe existing role then click on "create function".



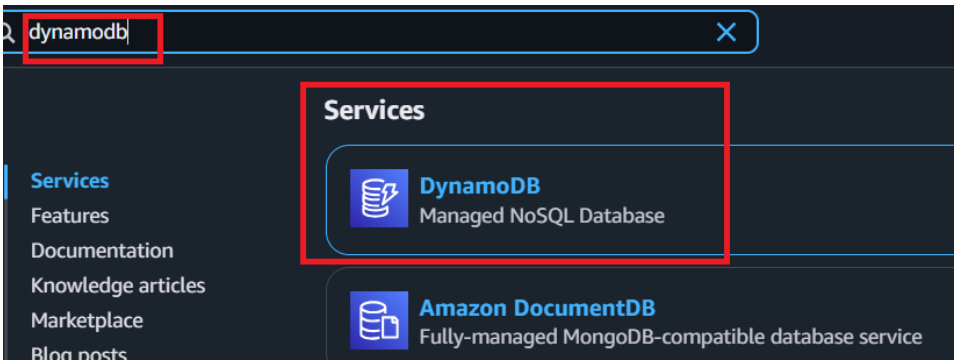
v. Copy the code from lambda_function.py from my github then paste in the created lambda function. Then perform STEP c which we will create the dynamodb then Finally click on Deploy. For testing click on test give a name then give this code {“body”:”\inputs\”:\”Summarize the text about heart disease\”}”

1. This code is used to invoke the endpoint of aws sagemaker

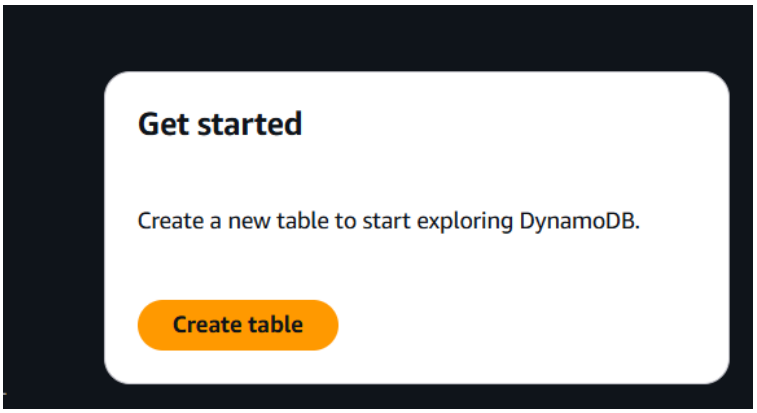


c. Create the DynamoDB for storing the logs

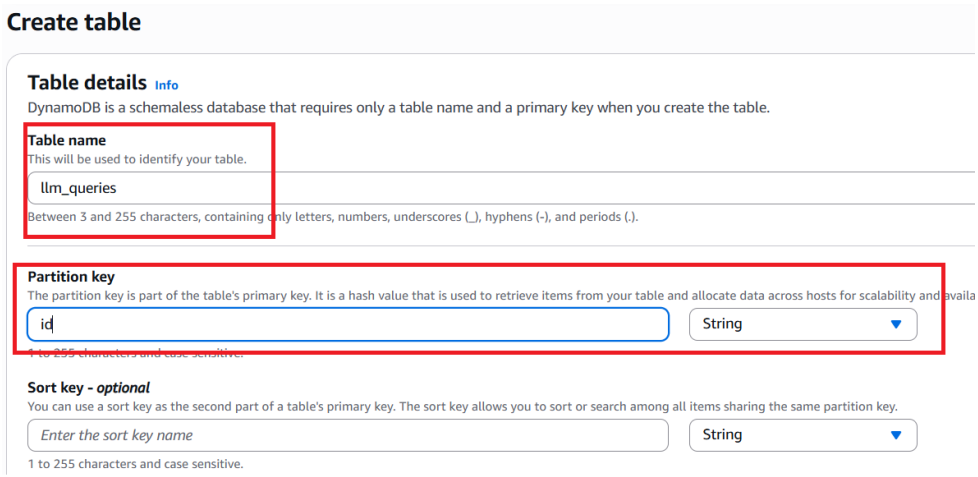
i. Search for dynamodb then click on dynamodb. Click on it



ii. Click on “create Table”

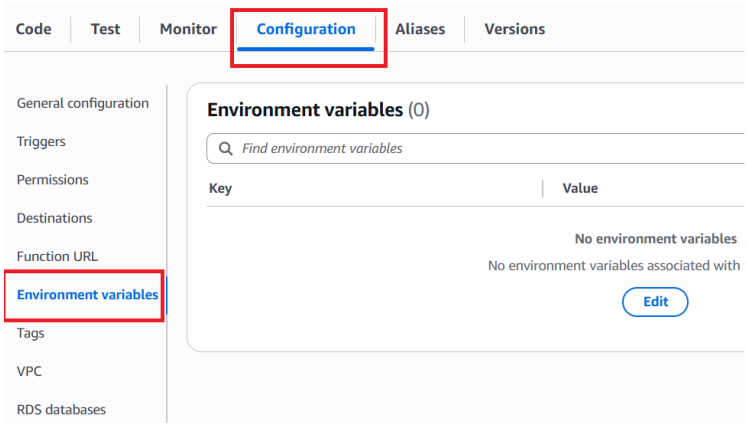


iii. Give the “Table name” and “Partition key” then finally click on “Create table” which will be displayed on down. Take the Table name ans paste in notepad this table name we will use in lambda function.

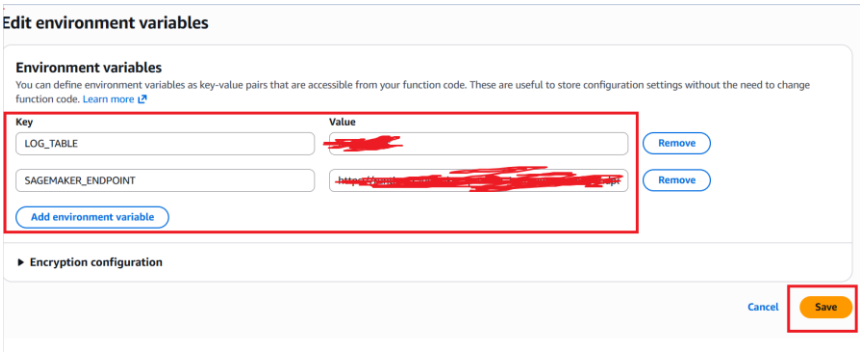


iv. Now we need to add the 2environment variables Dynamodb “logtable” and “Sagemaker_endpoint”

1. Open the configuration in lambda function only then click on “Environment variables”

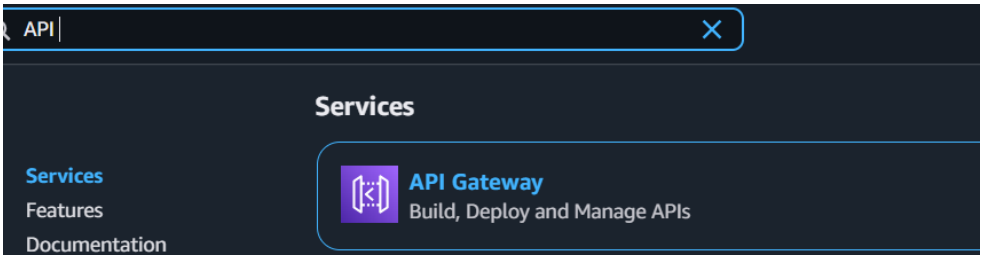


2. Click on “Edit”.
    a. Give the “LOG_TABLE” as key and value as the created dynamodb name
    b. Give the “SAGEMAKER_ENDPOINT” as key and value as the “live-finetune-endpoint” where we have given in the estimator_launcher.ipynb then click on Save

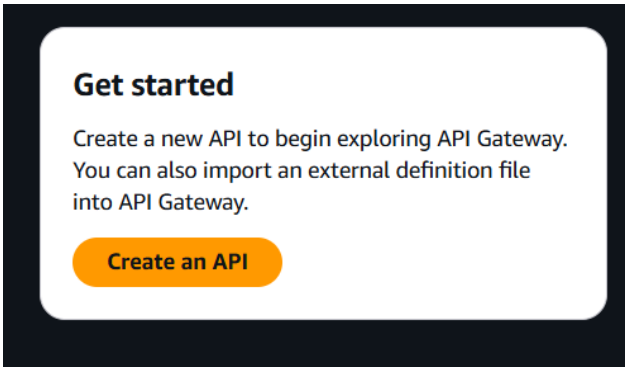


d. Now we will create API Gateway

i. Search for the API Gateway then click on that

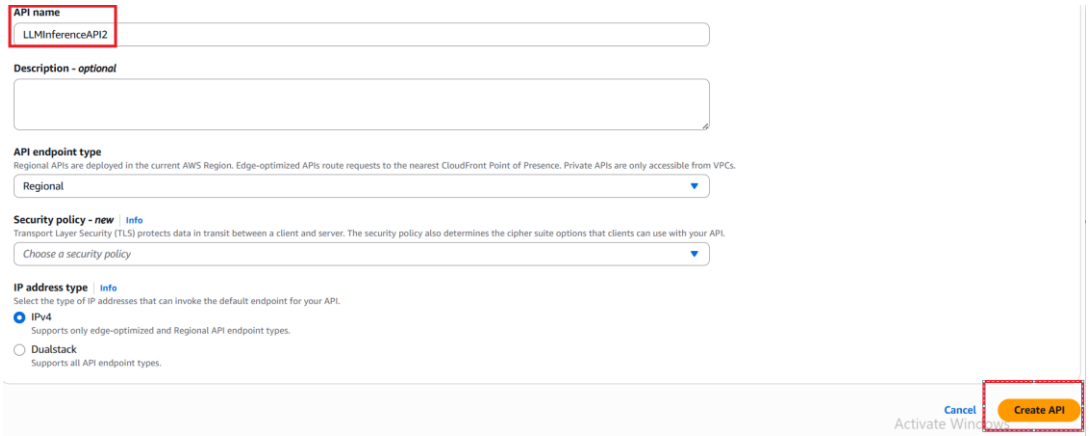


ii. Click on “Create an API”.

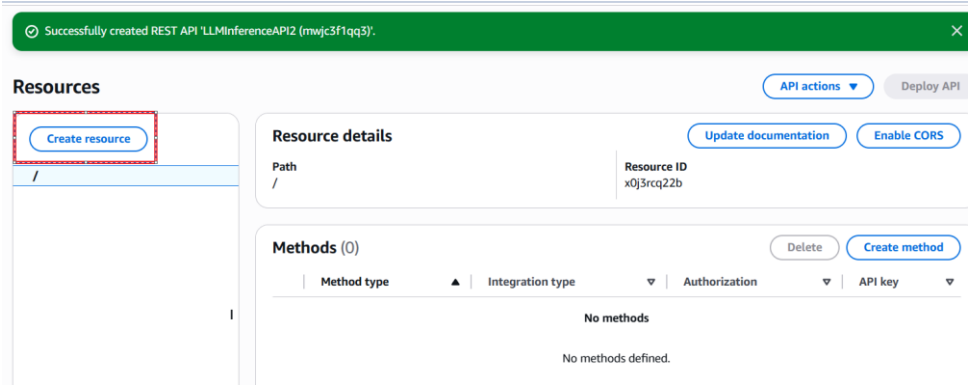


iii. Choose the “REST API” click on “Build”.

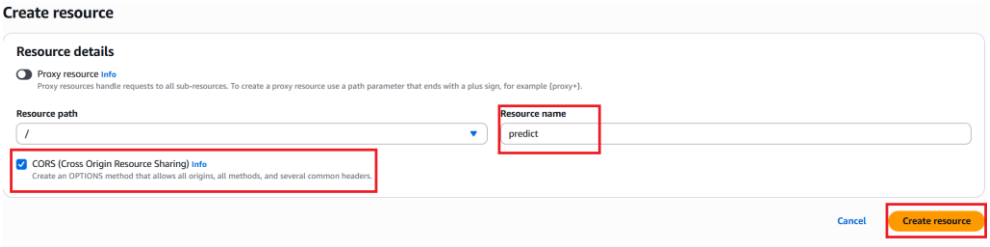
iv. Give any Name under “API name” then keepeverything bydefault then click on “Create API”.



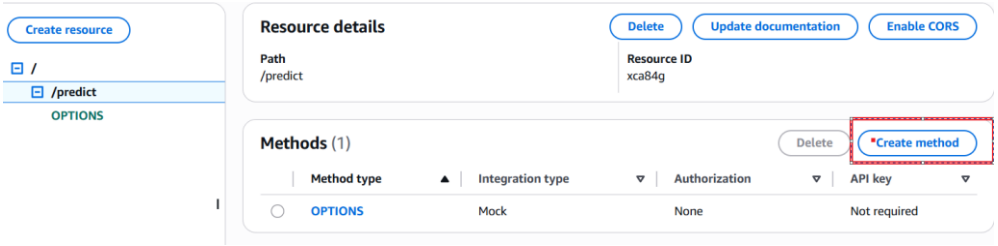
v. Click on “Create resource”.



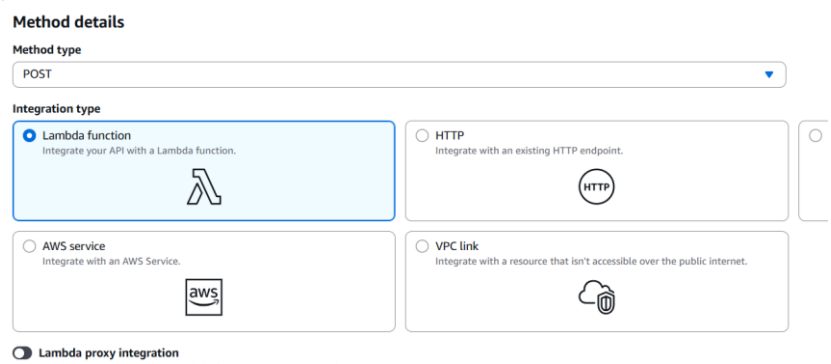
vi. Give the Resource Name. click on CORS checkbox then click on “Create Resource”.



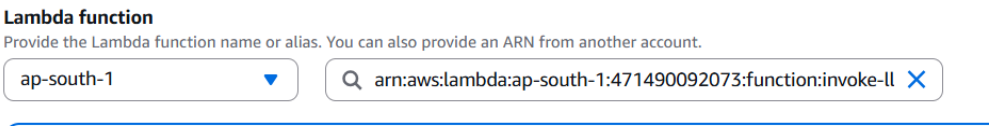
vii. Now we will create the “Method”. To create click on “Create method”.



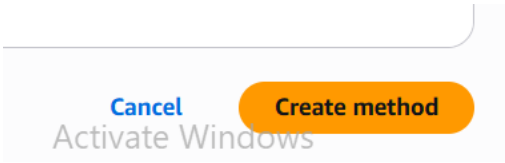
viii. Under the method type select the “POST” method.



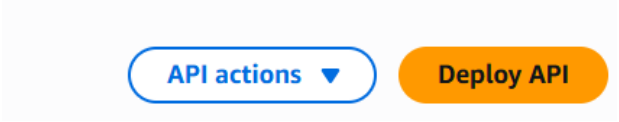
ix. Under the Lambda function select the “ap-south-1” then give the lambda function.



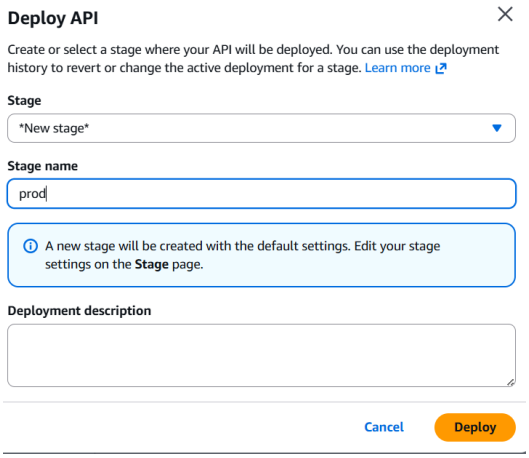
x. Finally click on “Create method”.



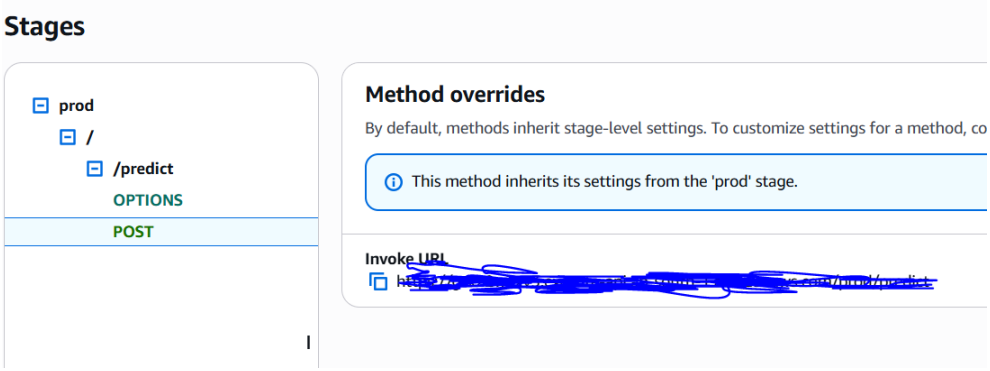
xi. Then click on “Deploy API” before refresh the page for enabling.



xii. Select the Newstage give name click on deploy.

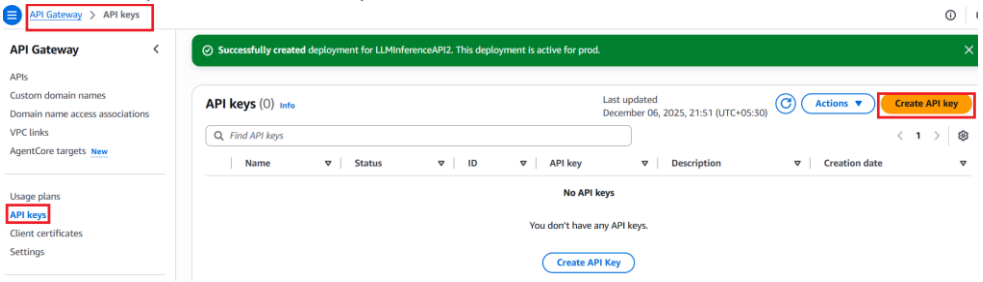


xiii. This is our URL.

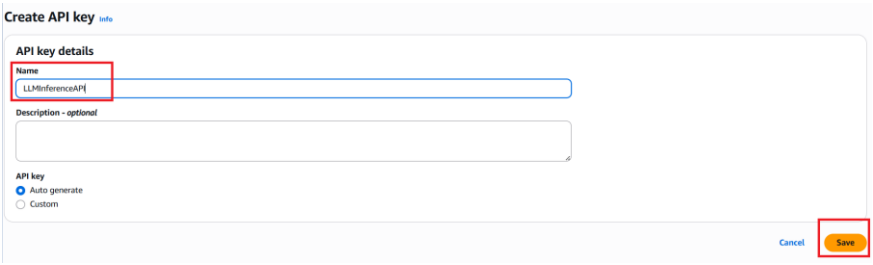


xiv. Now we will create a API key.

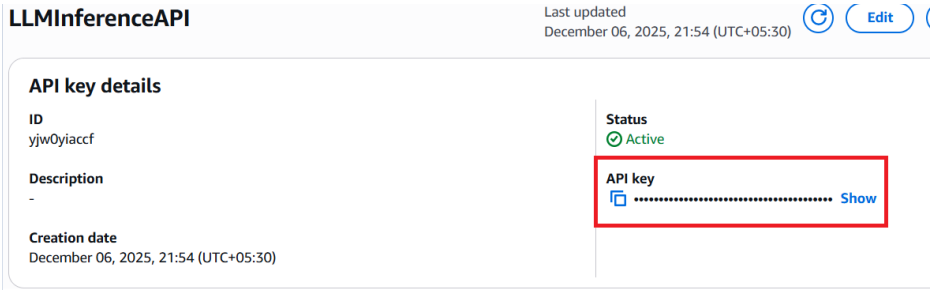
1. Goto API keys=>Create API key



2. Give any name click on “Save”.



3. Copy the key and paste it somewhere.

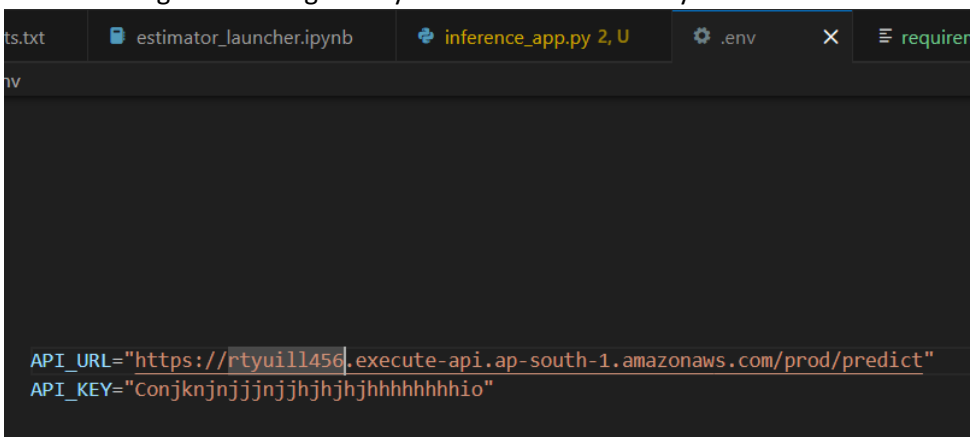


6. With the help of inference_app.py we will hit our application

a. Create a virtual environment in local
   i. uv python list
   ii. uv venv env --python cpython-3.11.13-windows-x86_64-none
   iii. env/Scripts/activate
   iv. source env/Scripts/activate→ for bash terminal
   v. uv pip install -r requirements_inference.txt

b. streamlit run inference_app.py

c. Create .env file then give the API gateway Invoke URL and API key in that .env file



d. streamlit run rag_app_ui.py –server.port 8502

e. In .env give all these variables
   i. Env variables:
   ii. GROQ_API_KEY=""
   iii. GOOGLE_API_KEY=""
   iv. TAVILY_API_KEY=""
   v. OPENAI_API_KEY=""
   vi. API_URL = "" # Replace with your actual API Gateway endpoint
   vii. API_KEY= ""

7. Below are the screenshots of my application.

a. Run “streamlit run rag_app_ui.py –server.port 8502”.
b. Enter the Question then click on “Generate Answer” to get the answer from Vectordatabase and LLM.