ASSIGNMENT - 15
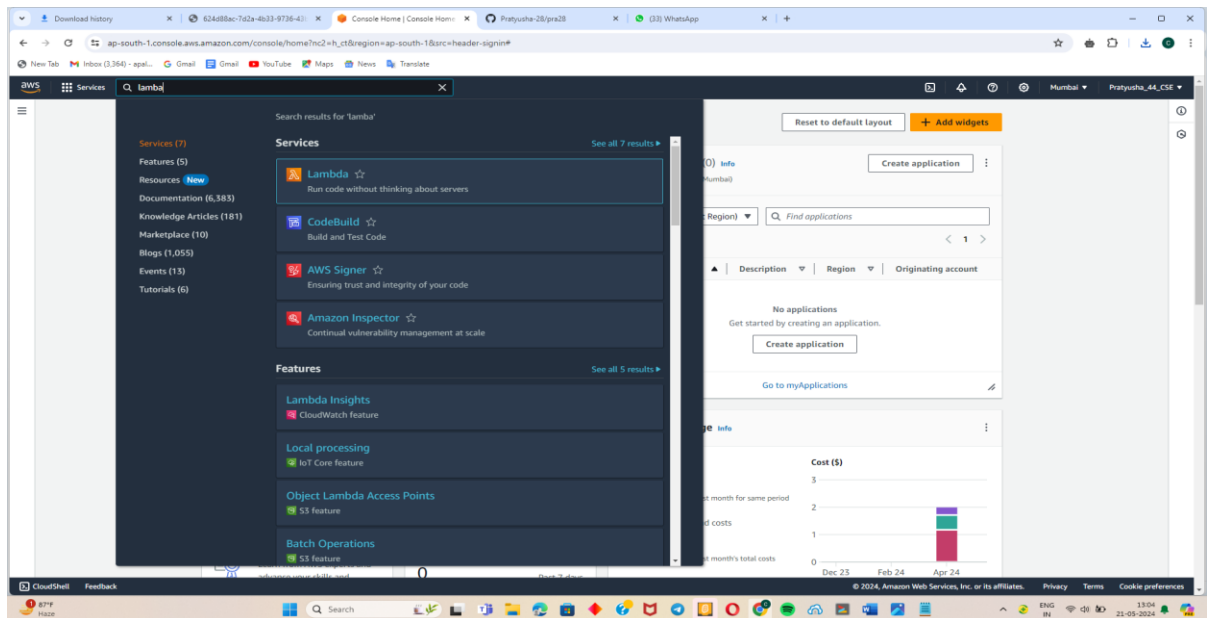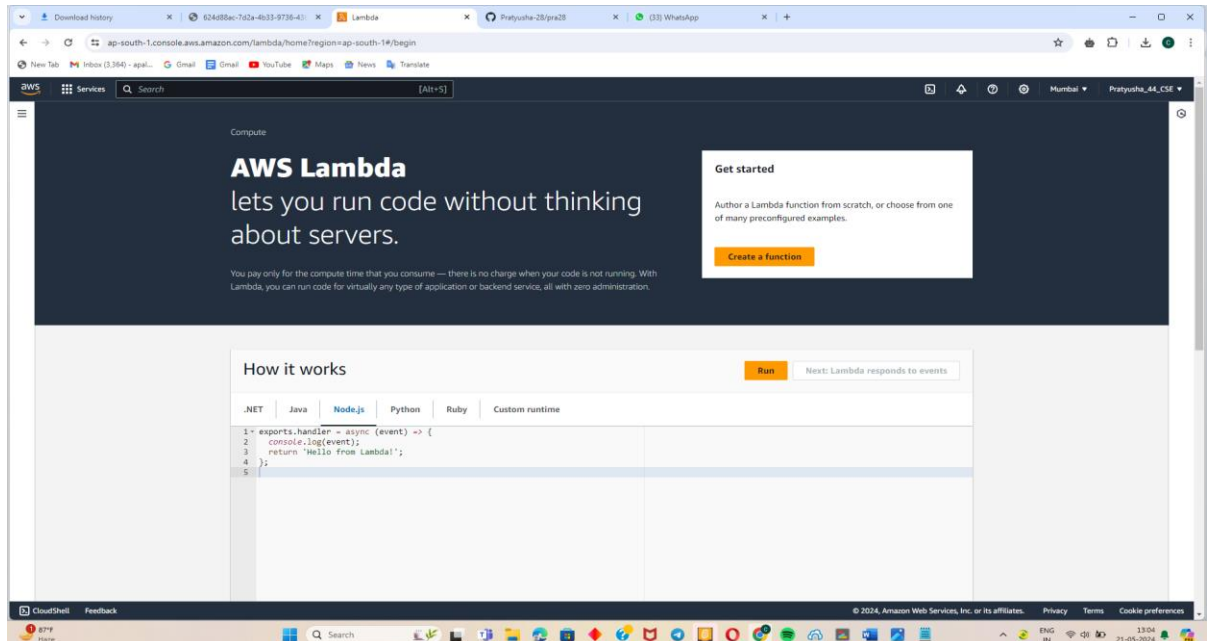
PROBLEM STATEMENT ->

Create a Serverless Computing Service. To Create Serverless Computing Service
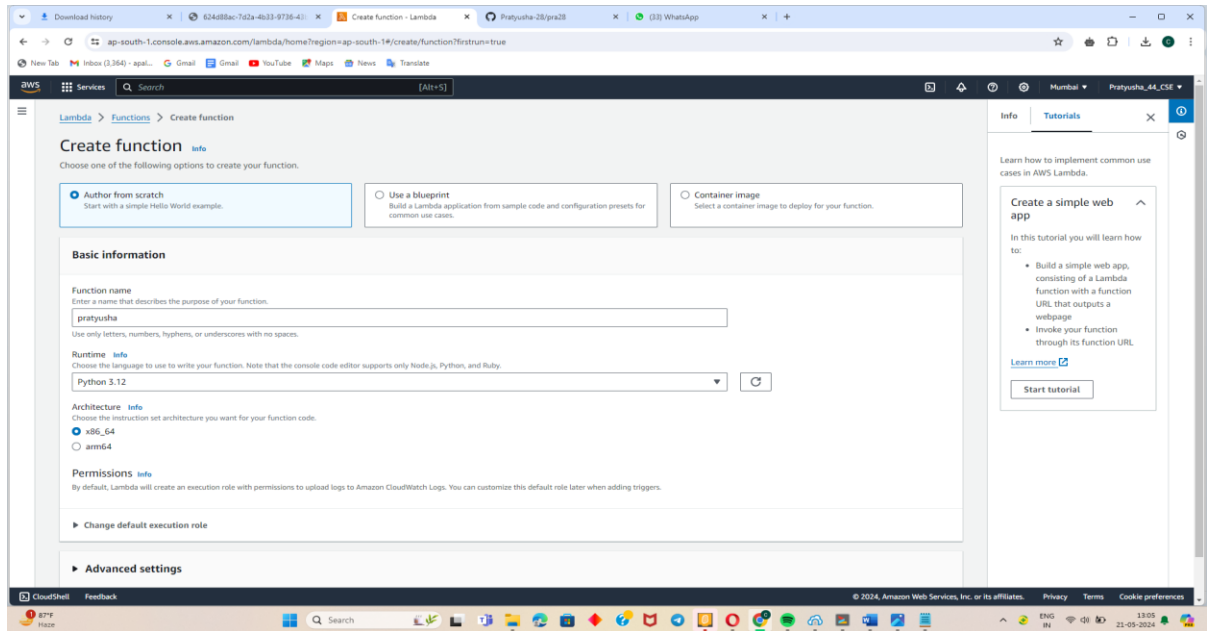
Procedure -

1. Search for Lambda and click on it.

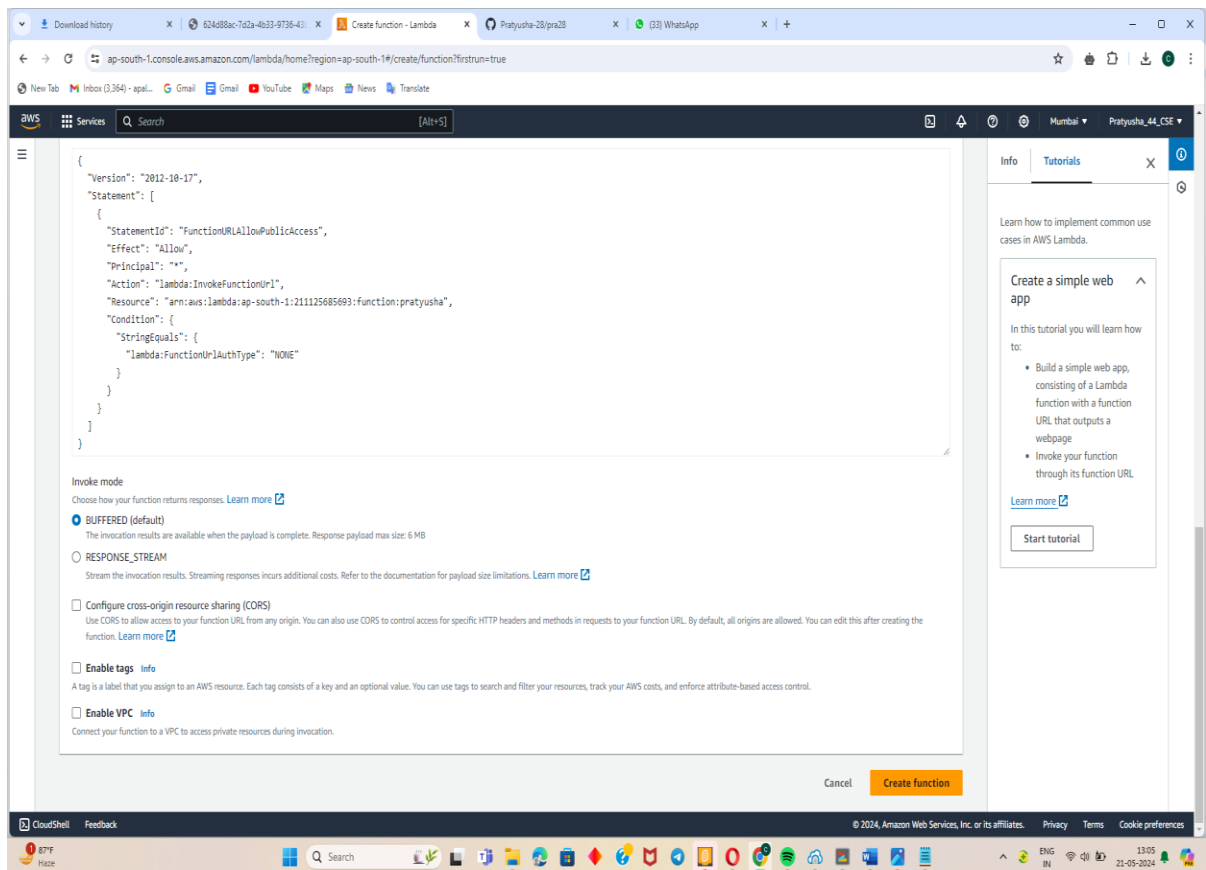2. Click on Create a function button.

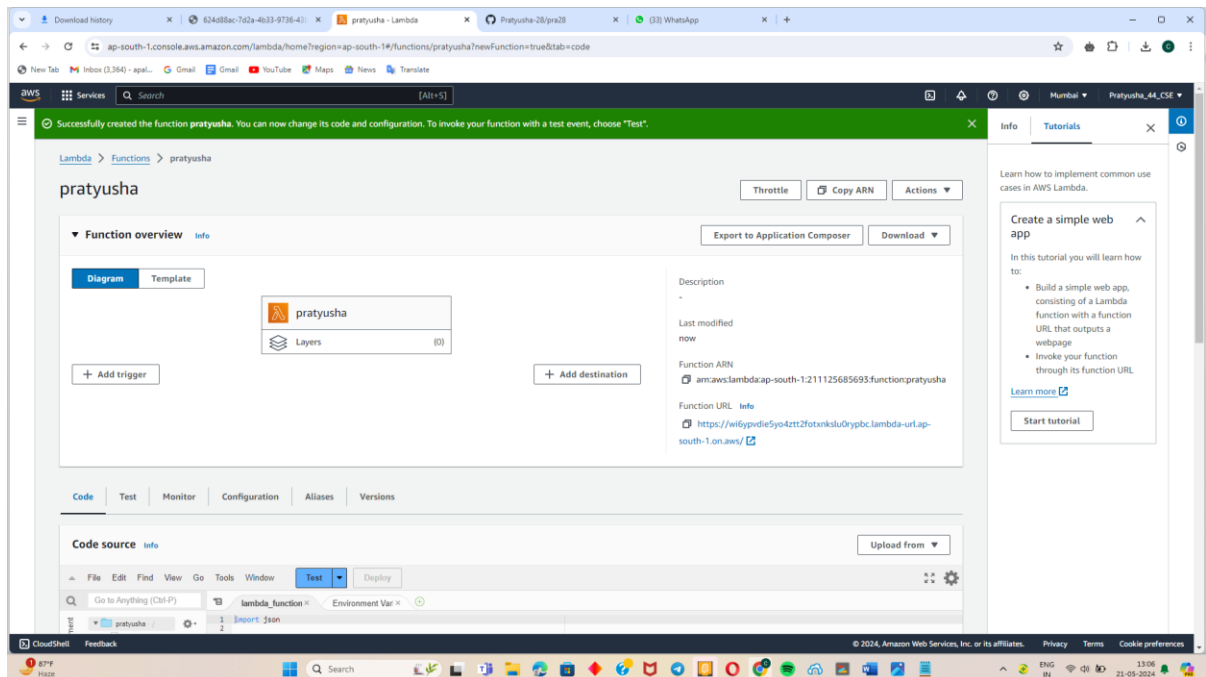3. Give a name to the function and under runtime select Python.

4. Under the Advanced settings, check the Enable Function URL checkbox & under Auth type, select None.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "StatementId": "FunctionURLAllowPublicAccess",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "lambda:InvokeFunctionUrl",
            "Resource": "arn:aws:lambda:ap-south-1:211125685693:function:pratyusha",
            "Condition": {
                "StringEquals": {
                    "lambda:FunctionUrlAuthType": "NONE"
                }
            }
        }
    ]
}
```

Invoke mode

Choose how your function returns responses. Learn more ↗

- ● BUFFERED (default)
  The invocation results are available when the payload is complete. Response payload max size: 6 MB
- ○ RESPONSE_STREAM
  Stream the invocation results. Streaming responses incurs additional costs. Refer to the documentation for payload size limitations. Learn more ↗

☐ **Configure cross-origin resource sharing (CORS)**
Use CORS to allow access to your function URL from any origin. You can also use CORS to control access for specific HTTP headers and methods in requests to your function URL. By default, all origins are allowed. You can edit this after creating the function. Learn more ↗

☐ **Enable tags** Info
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources, track your AWS costs, and enforce attribute-based access control.

☐ **Enable VPC** Info
Connect your function to a VPC to access private resources during invocation.

Cancel   **Create function**

Python 3.12 ▾

Architecture Info
Choose the instruction set architecture you want for your function code.

- ● x86_64
- ○ arm64

Permissions Info
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

▸ Change default execution role

▾ **Advanced settings**

☐ **Enable Code signing** Info
Use code signing configurations to ensure that the code has been signed by an approved source and has not been altered since signing.

☑ **Enable function URL** Info
Use function URLs to assign HTTP(S) endpoints to your Lambda function.

Auth type
Choose the auth type for your function URL. Learn more ↗

- ○ AWS_IAM
  Only authenticated IAM users and roles can make requests to your function URL.
- ● NONE
  Lambda won't perform IAM authentication on requests to your function URL. The URL endpoint will be public unless you implement your own authorization logic in your function.

Function URL permissions

ⓘ When you choose auth type **NONE**, Lambda automatically creates the following resource-based policy and attaches it to your function. This policy makes your function public to anyone with the function URL. You can edit the policy later. To limit access to authenticated IAM users and roles, choose auth type **AWS_IAM**.

▾ View policy statement

5. Click on Create Function.

6. Click on the function URL to see the output.

7. New tab will be open.