	Marwadi University Faculty of Engineering and Technology Department of Information and Communication Technology	
Subject: Cloud Developing (01CT0720)	Aim: Deploy the static website using S3 and apply bucket policy to make it secure.	
Experiment No: 02	Date:	Enrolment No: 92200133030

Aim :- Deploy the static website using S3 and apply bucket policy to make it secure.

Lab overview and objectives
In this lab, you use Amazon Simple Storage Service (Amazon S3) to host a static website. You will also implement architectural best practices to protect and manage your data.
After completing this lab, you should be able to:

- Create a static website by using Amazon S3.
- Apply a bucket policy on an S3 bucket to configure customized data protection.
- Upload objects to an S3 bucket by using the AWS SDK for Python (Boto3).
- Configure the website that is hosted on Amazon S3 to be accessible only from a specific IP address range, and test the configuration.

AWS service restrictions
In this lab environment, access to AWS services and service actions might be restricted to the ones that are needed to complete the lab instructions. You might encounter errors if you attempt to access other services or perform actions beyond the ones that are described in this lab.

Scenario

Sofía is eager to start building a website for the café. She has some Python development skills and she's learning more about how to develop solutions on the cloud. Nikhil is a secondary school student who also works at the café. He has some skills in graphic design and a strong interest in learning about cloud computing.
The café has a single location in a large city. It mostly gains new customers when someone walks by, notices the café, and decides to try it. The café has a reputation for high-quality desserts and coffees, but their reputation is limited to people who have visited, or who have heard about them from their customers.
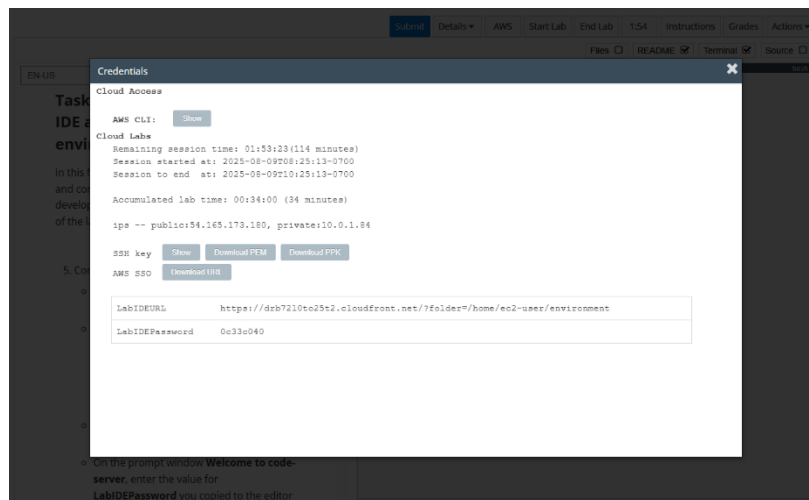
Sofía suggests to Frank and Martha (her parents and the owners of the café) that they should expand community awareness of what the café has to offer. The café doesn't have a web presence, and it doesn't currently use any cloud computing services. However, that situation is about to change.

In this lab, you will play the role of Sofía and take the first steps needed create a website for the café. It's time to get started!

A business request for the café: Create a static website

Sofía mentions to Nikhil that she would like to build a proof-of-concept website for Frank and Martha that will showcase the café's offerings visually. It would also provide business details, such as the location of the store, business hours, and telephone number. Nikhil is eager to see how she will build the website, and he agrees to create the graphics that she will use.
For this first challenge, you will take on the role of Sofía. You will use the AWS Command Line

	Marwadi University Faculty of Engineering and Technology Department of Information and Communication Technology		
Subject: Cloud Developing (01CT0720)	**Aim: Deploy the static website using S3 and apply bucket policy to make it secure.**		
Experiment No: 02	**Date:**		**Enrolment No: 92200133030**

Interface (AWS CLI) and the SDK for Python to configure Amazon S3 so that it hosts a basic website for the café.

Task 1: Connecting to VS Code IDE and configuring the environment

In this first task, you will connect to VS Code IDE and configure the environment to support the development that you will work on during the rest of the lab.

5. Connect to the VS Code IDE.
    o At the top of these instructions, choose Details followed by **AWS: Show**
    o Copy values from the table **similar** to the following and paste it into an editor of your choice for use later.
        ▪ **LabIDEURL**
        ▪ **LabIDEPassword**
    o In a new browser tab, paste the value for **LabIDEURL** to open the VS Code IDE.
    o On the prompt window **Welcome to code-server**, enter the value for **LabIDEPassword** you copied to the editor earlier, choose **Submit** to open the VS Code IDE.

	Marwadi University **Faculty of Engineering and Technology** **Department of Information and Communication Technology**
Subject: Cloud Developing (01CT0720)	**Aim: Deploy the static website using S3 and apply bucket policy to make it secure.**
Experiment No: 02	**Date:**	**Enrolment No: 92200133030**

6. Install the AWS SDK for Python.
   - o In the VS Code bash terminal (located at the bottom of the IDE), run the following commands:

   ```
   sudo pip3 install boto3
   ```


7. Download and extract the files that you will need for this lab.
   - In the same terminal, run the following command:

   ```
   wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-
   200-ACCDEV-2-91558/02-lab-s3/code.zip -P /home/ec2-user/environment
   ```

   - The code.zip file is downloaded to the VS Code IDE. The file is listed in the left navigation pane.

	Marwadi University
![Marwadi University logo] MARWADI University Marwadi Chandarana Group	**Marwadi University** **Faculty of Engineering and Technology** **Department of Information and Communication Technology**
Subject: Cloud Developing (01CT0720)	**Aim: Deploy the static website using S3 and apply bucket policy to make it secure.**
Experiment No: 02	**Date:** **Enrolment No: 92200133030**

```
[ec2-user@ip-10-0-1-84 environment]$ wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACCDEV-2-91558/02-lab-s3/code.zip -
P /home/ec2-user/environment
--2025-08-09 15:36:41--  https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACCDEV-2-91558/02-lab-s3/code.zip
Resolving aws-tc-largeobjects.s3.us-west-2.amazonaws.com (aws-tc-largeobjects.s3.us-west-2.amazonaws.com)... 52.92.187.162, 3.5.77.156, 52.92.1
80.50, ...
Connecting to aws-tc-largeobjects.s3.us-west-2.amazonaws.com (aws-tc-largeobjects.s3.us-west-2.amazonaws.com)|52.92.187.162|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5816045 (5.5M) [application/zip]
Saving to: '/home/ec2-user/environment/code.zip'

code.zip                      100%[===================================================================>]   5.55M  10.7MB/s    in 0.5s

2025-08-09 15:36:42 (10.7 MB/s) - '/home/ec2-user/environment/code.zip' saved [5816045/5816045]

[ec2-user@ip-10-0-1-84 environment]$
```

- Run the following commands to extract the file:
  ```
  unzip code.zip
  ```

```
[ec2-user@ip-10-0-1-84 environment]$ unzip code.zip
Archive:  code.zip
  extracting: python_3/permissions.py
  extracting: resources/website/callback.html
  extracting: resources/website/all_products.json
  extracting: resources/website/all_products_on_offer.json
  extracting: resources/website/config.js
  extracting: resources/website/beans.json
  extracting: resources/website/spint.md
  extracting: resources/website/index.html
  extracting: resources/website/favicon.ico
  extracting: resources/website/scripts/main.js
  extracting: resources/website/scripts/pastries.js
  extracting: resources/website/scripts/coffee.js
  extracting: resources/website/scripts/navigation.js
  extracting: resources/website/scripts/jquery-3.6.0.min.js
  extracting: resources/website/styles/beans.css
  extracting: resources/website/styles/main.css
  extracting: resources/website/styles/pastries.css
  extracting: resources/website/styles/reset.css
  extracting: resources/website/styles/navigation.css
  extracting: resources/website/styles/coffee.css
  extracting: resources/website/images/main_coffee_shop.png
  extracting: resources/website/images/main_coffee_shop_1.png
  extracting: resources/website/images/logo.png
  extracting: resources/website/images/not_expanded.png
  extracting: resources/website/images/expanded.png
  extracting: resources/website/images/items/garlic_bagel.jpeg
  extracting: resources/website/images/items/chocolate_cupcake.png
  extracting: resources/website/images/items/chocolate_cake.png
  extracting: resources/website/images/items/chocolate_cake_slice.jpeg
  extracting: resources/website/images/items/chocolate_chip_cupcake.jpeg
  extracting: resources/website/images/items/strawberry_cupcake.jpeg
  extracting: resources/website/images/items/cinnamon_doughnut.jpeg
  extracting: resources/website/images/items/eclair.jpeg
  extracting: resources/website/images/items/peanutbutter_and_chocolate_cupcake.png
  extracting: resources/website/images/items/blueberry_jelly_doughnut.jpeg
  extracting: resources/website/images/items/cherry_pie_slice.png
  extracting: resources/website/images/items/lemon_pie.jpeg
```

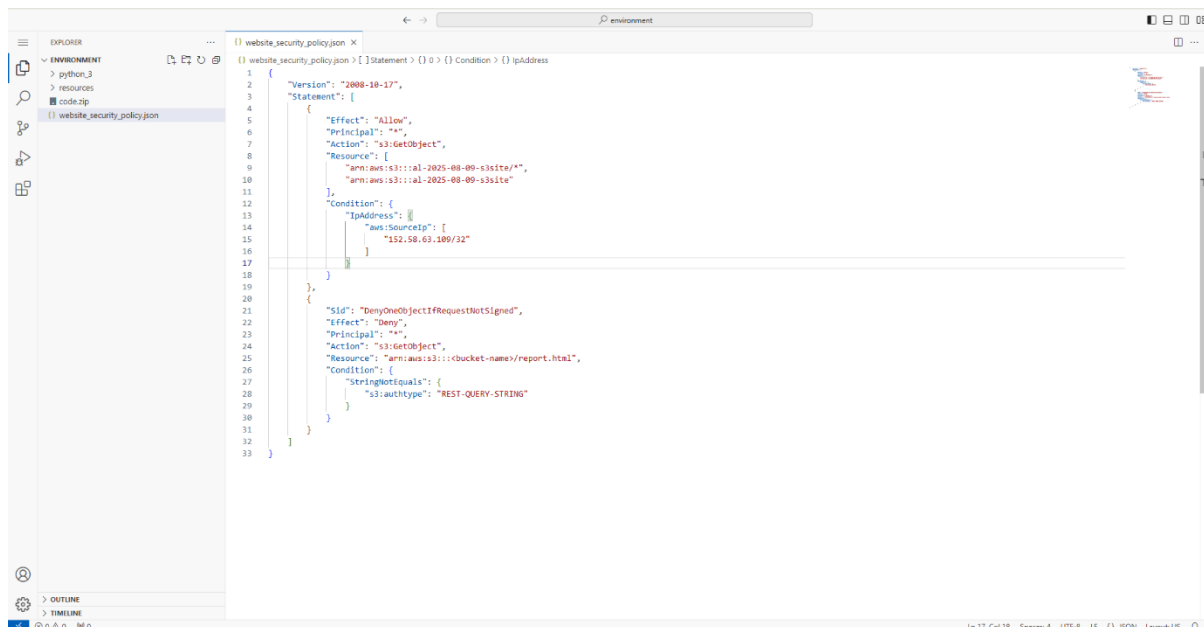|	Marwadi University Faculty of Engineering and Technology Department of Information and Communication Technology
Subject: Cloud Developing (01CT0720)	**Aim: Deploy the static website using S3 and apply bucket policy to make it secure.**
Experiment No: 02	**Date:**	**Enrolment No: 92200133030**

8. Verify the AWS CLI version 2.
   Verify the CLI version:
   ```
   aws --version
   ```

```
[ec2-user@ip-10-0-1-84 environment]$ aws --version
aws-cli/2.27.57 Python/3.9.23 Linux/6.1.147-172.266.amzn2023.x86_64 source/x86_64.amzn.2023
[ec2-user@ip-10-0-1-84 environment]$
```

Task 2: Creating an S3 bucket by using the AWS CLI

In this task, you will create an S3 bucket to host your website. You will complete this task by using the AWS CLI.

9. In the VS Code bash terminal, run the following command, replacing *<bucket-name>* with your bucket name.

   For the *bucket name*, use the following items, separated by dashes (-):
   - Your initials in lowercase
   - Today's date in the format *YYYY-MM-DD*
   - The word *s3site*

   For example, *Sofía Martínez* might name the bucket *sm-2022-08-26-s3site*.
   ```
   aws s3api create-bucket --bucket <bucket-name> --region us-east-1
   ```

```
[ec2-user@ip-10-0-1-84 environment]$ aws s3api create-bucket --bucket al-2025-08-09-s3site --region us-east-1
{
    "Location": "/al-2025-08-09-s3site"
}
[ec2-user@ip-10-0-1-84 environment]$
```

10. Use the AWS Management Console to confirm that the bucket was created, and observe the current permissions settings on the bucket.

    o In the AWS Management Console, choose **Services** and select **S3**. You should see that the bucket was created.

    The **Access** column for the bucket indicates that *Bucket and objects not public.*

    o Choose the bucket name and then choose **Permissions**.
    o Notice that **Block *all* public access** is turned on.
    o Choose **Edit**, de-select **Block *all* public access**.
    o Select **Block public access to buckets and objects granted through new access control lists (ACLs)**.
    o Select **Block public access to buckets and objects granted through any access control lists (ACLs)**.

	Marwadi University Faculty of Engineering and Technology Department of Information and Communication Technology
Subject: Cloud Developing (01CT0720)	**Aim: Deploy the static website using S3 and apply bucket policy to make it secure.**
Experiment No: 02	**Date:**	**Enrolment No: 92200133030**

- o Select **Block public and cross-account access to buckets and objects through any public bucket or access point policies**.
- o Choose **Save changes**.
- o In the confirm settings dialog, type in confirm and choose **Confirm**.

Note: you will configure the bucket with a bucket policy later in this lab.

- o Scroll further down and observe that the bucket currently doesn't have a bucket policy attached to it.
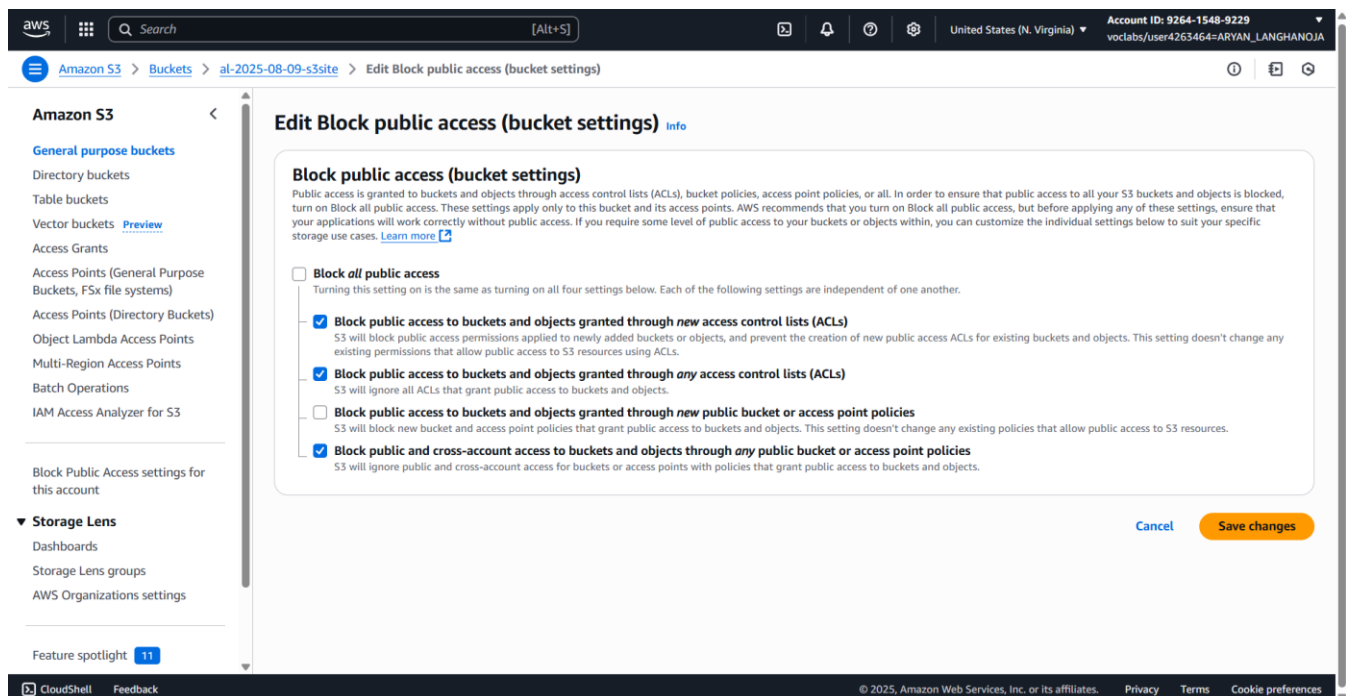
Task 3: Setting a bucket policy on the bucket by using the SDK for Python

Now that Sofia has created the bucket, she wants to set permissions on the bucket. She wants Frank, Martha, and Nikhil to be able to access the first version of the website that she's building so that they can see the progress she's making. However, she doesn't want to expose the site to the outside world yet. She will accomplish this objective by configuring a *bucket policy* on the bucket. This bucket policy will enforce a condition.
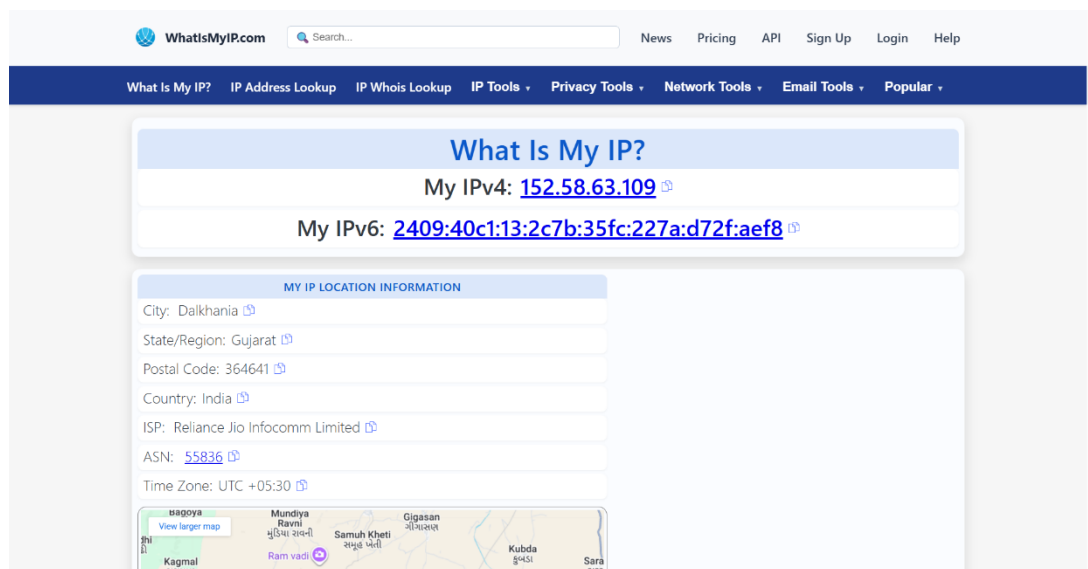
11. In the VS Code IDE, in the navigation pane, explorer menu.
- Choose Menu , choose File and choose New File .
- Name the empty file website_security_policy.json and choose Create New File (website_security_policy.json) to create the file.
- Choose OK on the prompt which shows the full file path */home/ec2-user/environment/website_security_policy.json*

	Marwadi University
![Marwadi University logo]	**Marwadi University** **Faculty of Engineering and Technology** **Department of Information and Communication Technology**
Subject: Cloud Developing (01CT0720)	**Aim: Deploy the static website using S3 and apply bucket policy to make it secure.**
Experiment No: 02	**Date:** **Enrolment No: 92200133030**

- In the policy document, replace all three <bucket-name> entries with your actual bucket name.
- Finally, replace <ip-address> with the IP address that is being used to connect your computer to the internet. You can find your IP address by visiting whatismyip.com

Note: you won't be able to view the website if you use 0.0.0.0 as the allowed aws:SourceIp, because Amazon S3 is smart enough to recognize that means public access, and since you just set the bucket permissions to Block *all* public access, S3 will block your access. Therefore ensure you use your specific IPv4 address.

The file is saved automatically.

	Marwadi University
Marwadi University **Faculty of Engineering and Technology** **Department of Information and Communication Technology**	
Subject: Cloud Developing (01CT0720)	**Aim: Deploy the static website using S3 and apply bucket policy to make it secure.**
Experiment No: 02	**Date:** **Enrolment No: 92200133030**

12. Apply the bucket policy to your bucket by using the SDK for Python.
- In the navigation pane, expand the **python_3 directory** and open the **permissions.py** file (which contains starter code).
- Edit the file by replacing *<bucket-name>* with the name of your bucket.
- **Note:** Changes are saved automatically.
- Finally, in the terminal, navigate to the **python_3** directory and run the following code:

```
cd python_3
python3 permissions.py
```

- If the command completed successfully, you should see the message *DONE* in the terminal output.

```
● [ec2-user@ip-10-0-1-84 environment]$ cd python_3/
● [ec2-user@ip-10-0-1-84 python_3]$ python3 permissions.py
  DONE
○ [ec2-user@ip-10-0-1-84 python_3]$ █
```

Task 4: Uploading objects to the bucket to create the website

Now that Sofía has configured permissions on the bucket, she needs to upload the website objects to the bucket. Fortunately, Sofía created the website code some time ago, and it's in the resources directory in the VS Code IDE. Nikhil created the graphics for the website, and he's excited to see what the website will look like. Sofía is now ready to upload the files. To complete this task, she will use the VS Code IDE Bash terminal to issue a simple recursive file upload command and disable caching (since the website is still being developed).

13. Run the code in the terminal. You should still be in the python_3 directory.
Be sure to replace <bucket-name> below with your actual bucket name

```
aws s3 cp ../resources/website s3://<bucket-name>/ --recursive --cache-
control "max-age=0"
```

	Marwadi University
![Marwadi University Logo] Marwadi University Marwadi Chandarana Group	**Marwadi University** **Faculty of Engineering and Technology** **Department of Information and Communication Technology**
Subject: Cloud Developing (01CT0720)	**Aim: Deploy the static website using S3 and apply bucket policy to make it secure.**
Experiment No: 02	**Date:**	**Enrolment No: 92200133030**

Task 5: Testing access to the website

Now that Sofía has uploaded the website files to the S3 bucket, she must test the site and verify that it loads when a user accesses it through a virtual secure endpoint.

14. Load the website in a browser tab.
    o Return to the browser tab with the Amazon S3 console.
    o Choose your bucket name, and then choose Objects.
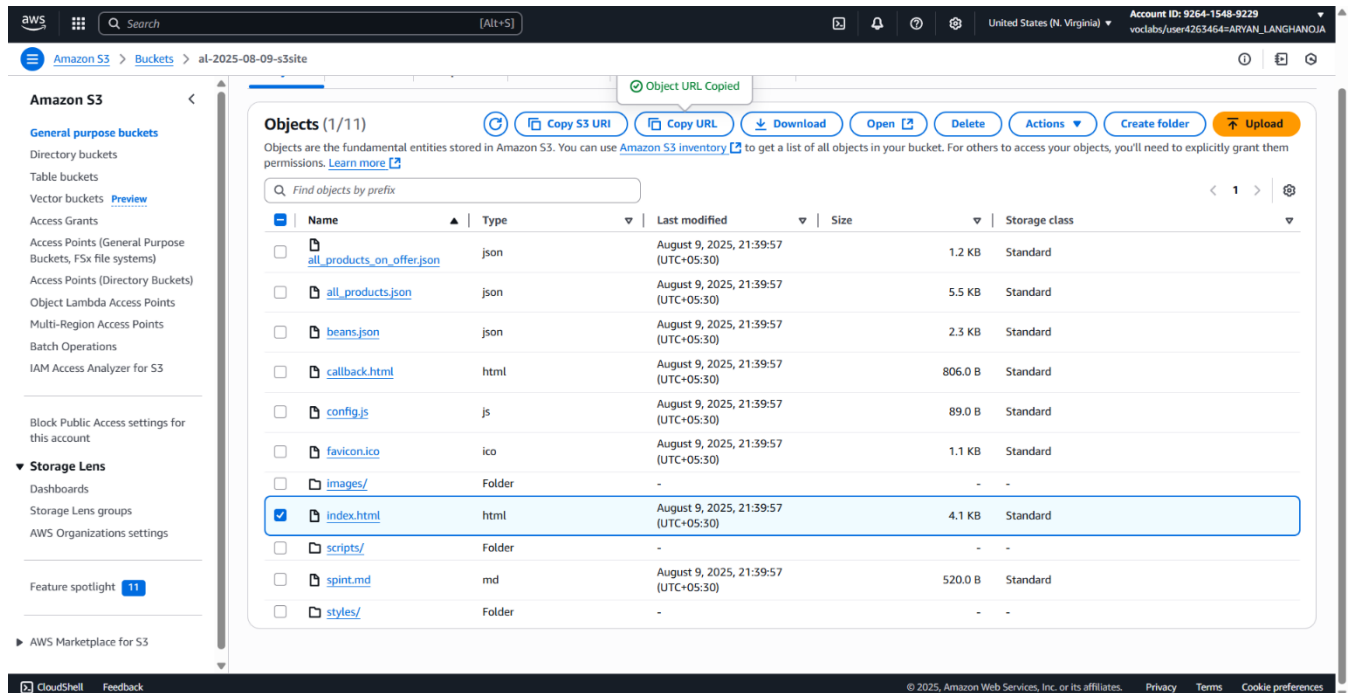
If the files you just uploaded do not display, choose the refresh icon to view them.

    o Choose the index.html file.
    o Copy the Object URL. It will be in the following format. https://<bucket-name>.s3.amazonaws.com/index.html
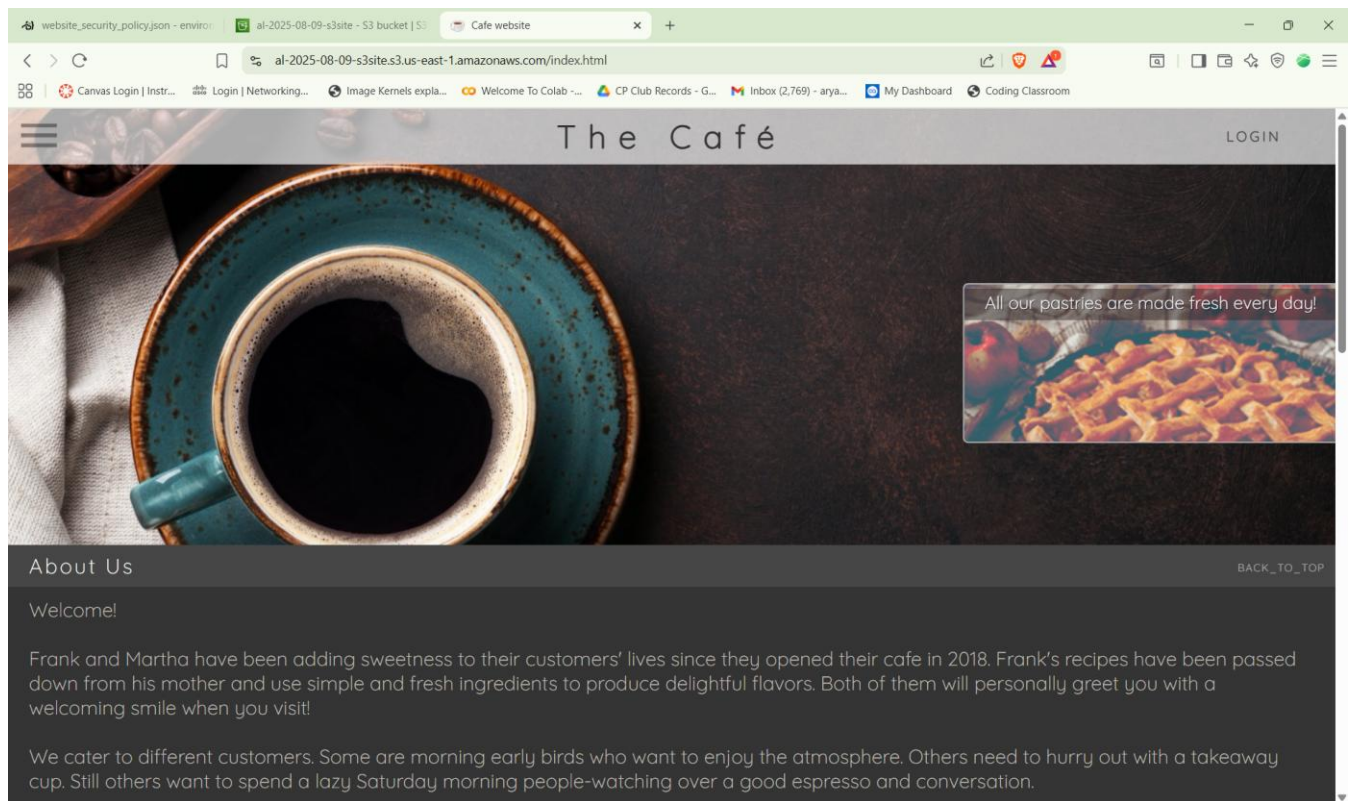
Note: In this lab scenario, the S3 bucket you created is intentionally *not* configured for static website hosting (a feature available in the bucket properties). Instead, you will access the website using the Object URL of the index.html file.

    o Verify that your website displays by pasting that full URI into your browser. Ensuring you are on the same network (as it will block anything other than the IPv4 you specified earlier)

	Marwadi University Faculty of Engineering and Technology Department of Information and Communication Technology
Subject: Cloud Developing (01CT0720)	**Aim: Deploy the static website using S3 and apply bucket policy to make it secure.**
Experiment No: 02	**Date:**	**Enrolment No: 92200133030**

	Marwadi University
MARWADI University Marwadi Chandarana Group	**Faculty of Engineering and Technology** **Department of Information and Communication Technology**
Subject: Cloud Developing (01CT0720)	**Aim: Deploy the static website using S3 and apply bucket policy to make it secure.**
Experiment No: 02	**Date:** **Enrolment No: 92200133030**

Conclusion:-

- In this Lab I Learned the AWS S3(Simple Storage System) Service. In that I had Learned how to create the S3 bucket from terminal using command :-

  ```
  aws s3api create-bucket --bucket <bucket-name> --region us-east-1
  ```

- I Learned About the Access policies of S3 bucket and modification of it via S3 management dashboard.

- After that I had Learned how to define the security policy using .json file

- After that I had uploaded all the website files to the s3 bucket.

  ```
  aws s3 cp ../resources/website s3://<bucket-name>/ --recursive --cache-control "max-age=0"
  ```

Result :-

Total score	**15/15**
[Task 2] The S3 bucket was created	5/5
[Task 3] Policy applied to bucket	5/5
[Task 4] index.html was copied to S3	5/5