	Marwadi University **Faculty of Engineering and Technology** **Department of Information and Communication Technology**	
Subject: Cloud Developing (01CT0720)	**Aim: Automating application deployment using a CI/CD pipeline.**	
Experiment No: 14	**Date:**	**Enrolment No: 92200133030**

Aim :- Automating application deployment using a CI/CD pipeline.

Lab overview and objectives

In this lab, you will create an AWS CodeCommit repository (also known as a repo) and an AWS CodePipeline pipeline. You will configure the pipeline to automatically apply updates to the café website as changes are saved to the repository.

After completing this lab, you should be able to:

- Create a new CodeCommit repository
- Clone and update a CodeCommit repository
- Create a pipeline by using CodePipeline

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

Now that the café website is in production, Frank wants a reliable process in place to track code changes and update the site when changes are made. He asked Sofía to find a way to centralize the website code and add version control. He has also asked if it's possible to automatically update the website instead of manually running scripts and uploading files when changes are made.

Mateo, a café regular and AWS consultant who specializes in automating repeatable processes, was chatting with Sofía about his work. He mentioned using CodeCommit to collaborate on projects with other developers. Sofía shared that she was researching CodeCommit to centralize the café website's code and asked Mateo for suggestions about automating updates to the site. Mateo suggested using CodePipeline because it easily integrates with both CodeCommit and Amazon Simple Storage Service (Amazon S3).

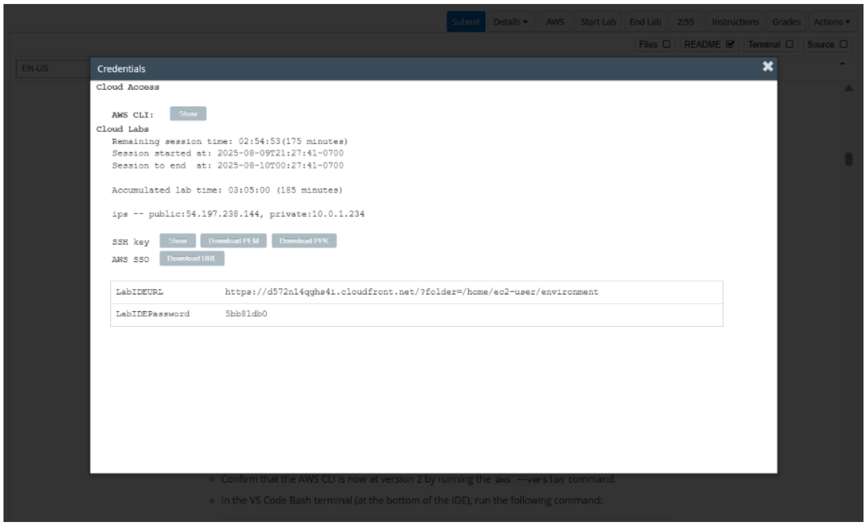
When you *start* the lab, many services are deployed for you. In this lab, you will focus on the services that are represented in the following diagram. As in past labs, the VS Code IDE is used as the development environment. The developer runs commands from VS Code IDE to update the code in the S3 bucket that hosts the website. The second bucket in the diagram will be used to set up a continuous integration and continuous delivery (CI/CD) pipeline.

By the *end* of this lab, you will have created the architecture in the following diagram. You will have created a CodeCommit repository to store the website code. You will have also created a CodePipeline pipeline to automate updates to the website when changes are pushed to the CodeCommit repository. Artifacts that CodePipeline uses to deploy and update your website application will be hosted on an S3 bucket that is separate from the bucket that hosts the café website.

Task 1: Preparing the lab

Connect to the VS Code IDE.

1. At the top of these instructions, choose Details followed by **AWS: Show**
2. Copy values from the table **similar** to the following and paste it into an editor of your choice for use later.
    a. **LabIDEURL**
    b. **LabIDEPassword**

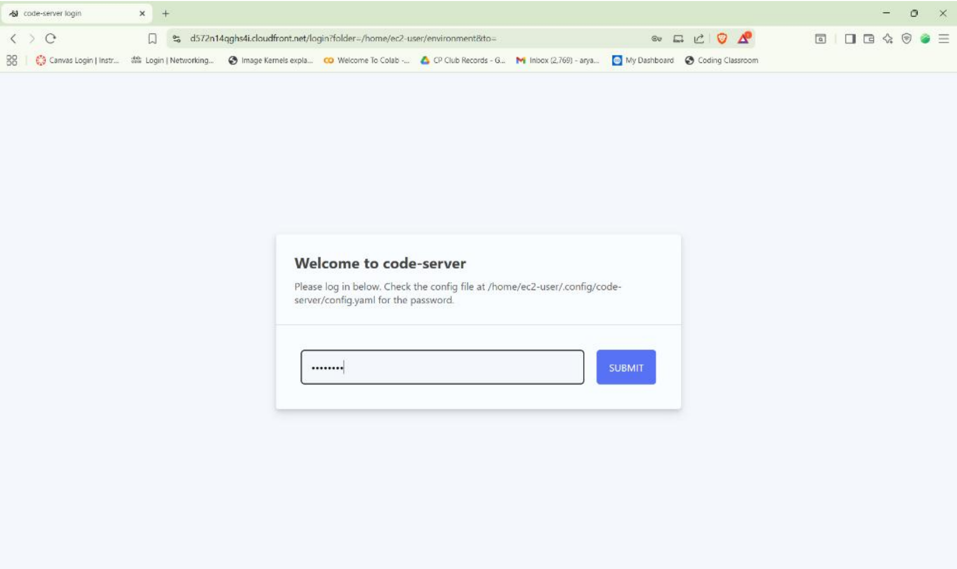


```
Cloud Access

  AWS CLI:  Show
Cloud Labs
  Remaining session time: 02:54:53(175 minutes)
  Session started at: 2025-08-09T21:27:41-0700
  Session to end  at: 2025-08-10T00:27:41-0700

  Accumulated lab time: 03:05:00 (185 minutes)

  ips -- public:54.197.238.144, private:10.0.1.234

  SSH key  Show  Download PEM  Download PPK
  AWS SSO  Download URL
```

LabIDEURL	https://d572n14qghs4i.cloudfront.net/?folder=/home/ec2-user/environment
LabIDEPassword	5bb81db0

- Confirm that the AWS CLI is now at version 2 by running the aws --version command.
- In the VS Code Bash terminal (at the bottom of the IDE), run the following command:

3. In a new browser tab, paste the value for **LabIDEURL** to open the VS Code IDE.
4. 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.



Welcome to code-server

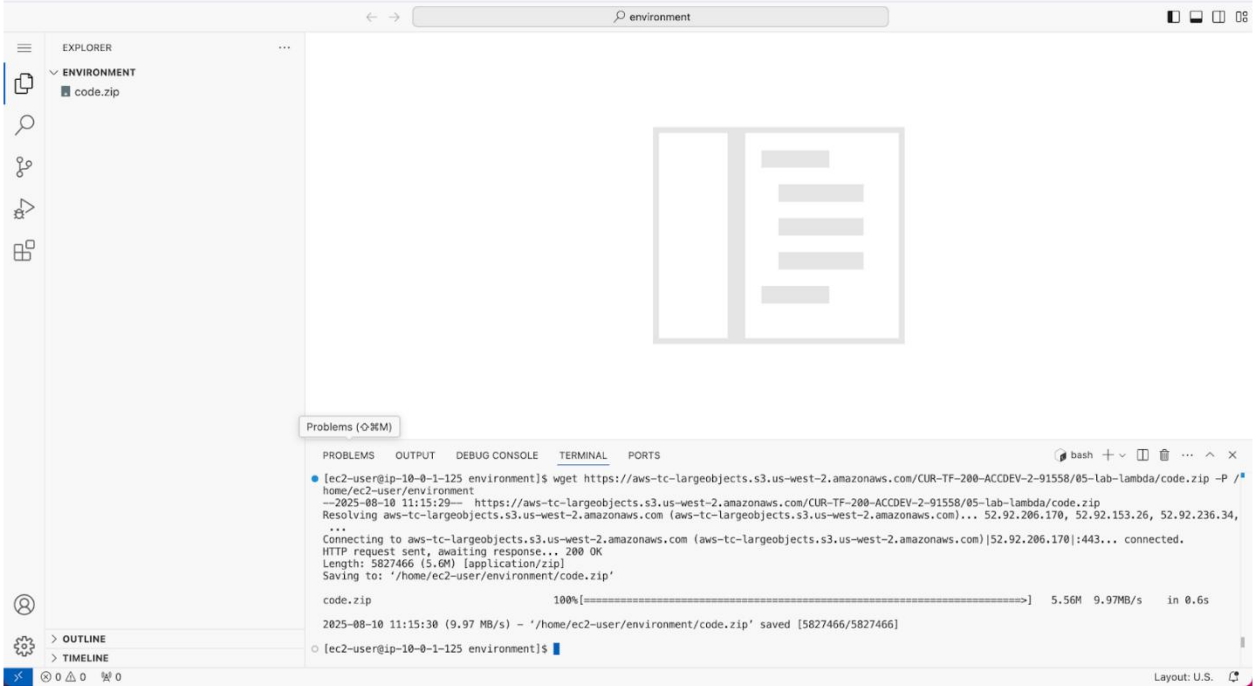
Please log in below. Check the config file at /home/ec2-user/.config/code-server/config.yaml for the password.

••••••••

SUBMIT

5. Download and extract the files that you need for this lab.
   - In the VS Code bash terminal (located at the bottom of the IDE), run the following commands:

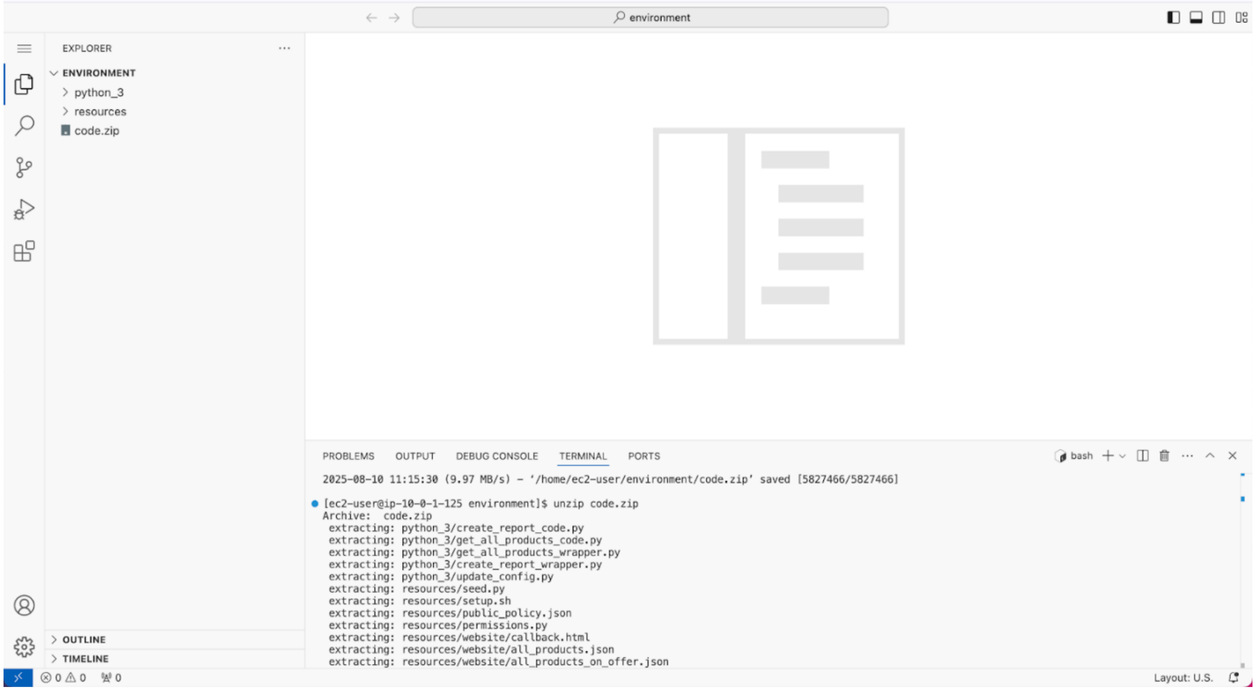
```
wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACCDEV-2-91558/13-lab-ci-cd/code.zip -P /home/ec2-user/environment
```



```
[ec2-user@ip-10-0-1-125 environment]$ wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACCDEV-2-91558/05-lab-lambda/code.zip -P /
home/ec2-user/environment
--2025-08-10 11:15:29--  https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACCDEV-2-91558/05-lab-lambda/code.zip
Resolving aws-tc-largeobjects.s3.us-west-2.amazonaws.com (aws-tc-largeobjects.s3.us-west-2.amazonaws.com)... 52.92.206.170, 52.92.153.26, 52.92.236.34,
 ...
Connecting to aws-tc-largeobjects.s3.us-west-2.amazonaws.com (aws-tc-largeobjects.s3.us-west-2.amazonaws.com)|52.92.206.170|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5827466 (5.6M) [application/zip]
Saving to: '/home/ec2-user/environment/code.zip'

code.zip                    100%[=======================================================================>]   5.56M  9.97MB/s    in 0.6s

2025-08-10 11:15:30 (9.97 MB/s) - '/home/ec2-user/environment/code.zip' saved [5827466/5827466]

[ec2-user@ip-10-0-1-125 environment]$ 
```

6. You should see that the **code.zip** file was downloaded to the VS Code IDE and is now in the left navigation pane.

- Extract the file by running the following command:
  `unzip code.zip`

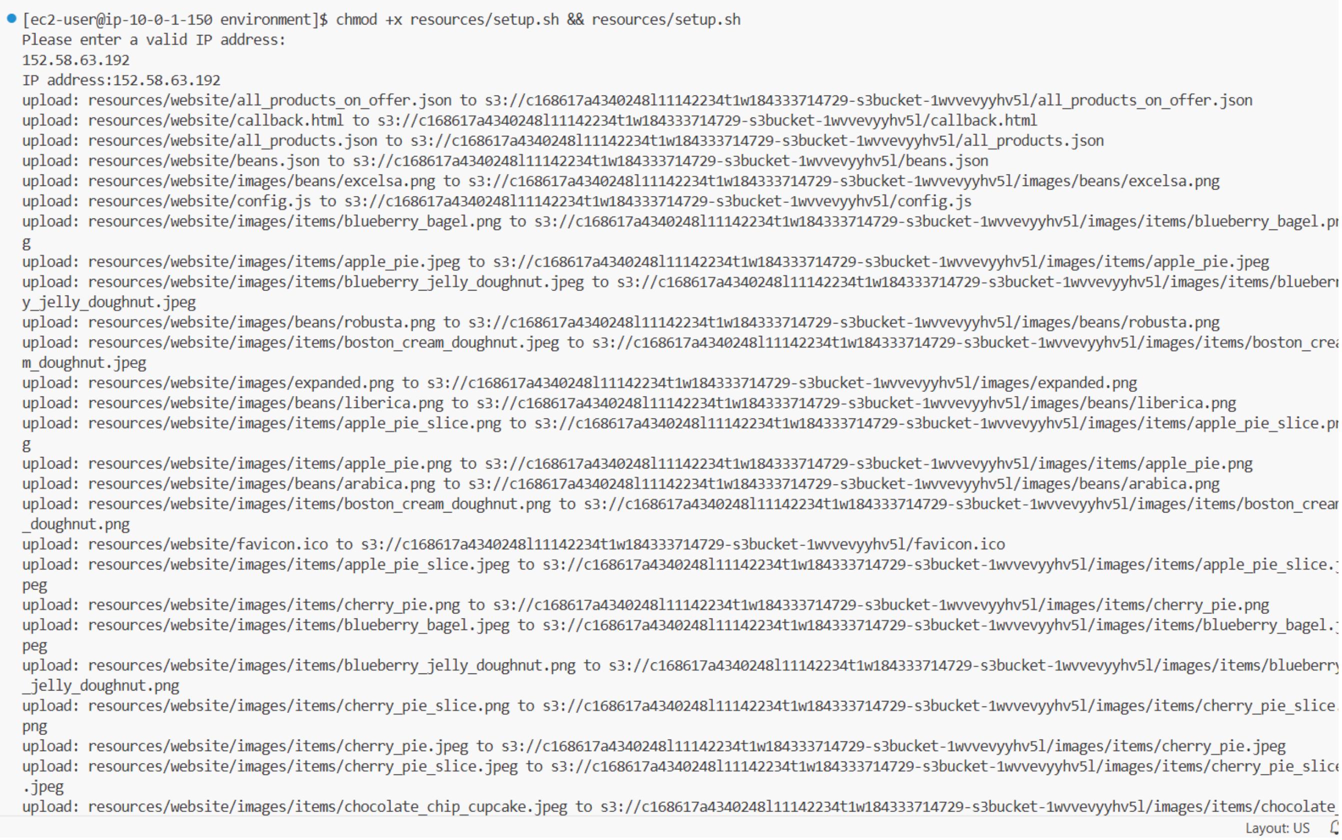


```
2025-08-10 11:15:30 (9.97 MB/s) - '/home/ec2-user/environment/code.zip' saved [5827466/5827466]

[ec2-user@ip-10-0-1-125 environment]$ unzip code.zip
Archive:  code.zip
 extracting: python_3/create_report_code.py
 extracting: python_3/get_all_products_code.py
 extracting: python_3/get_all_products_wrapper.py
 extracting: python_3/create_report_wrapper.py
 extracting: python_3/update_config.py
 extracting: resources/seed.py
 extracting: resources/setup.sh
 extracting: resources/public_policy.json
 extracting: resources/permissions.py
 extracting: resources/website/callback.html
 extracting: resources/website/all_products.json
 extracting: resources/website/all_products_on_offer.json
```

7. Run a script that upgrades the version of the AWS CLI installed on the VS Code IDE.
   - To set permissions on the script and then run it, run the following commands in the Bash terminal:

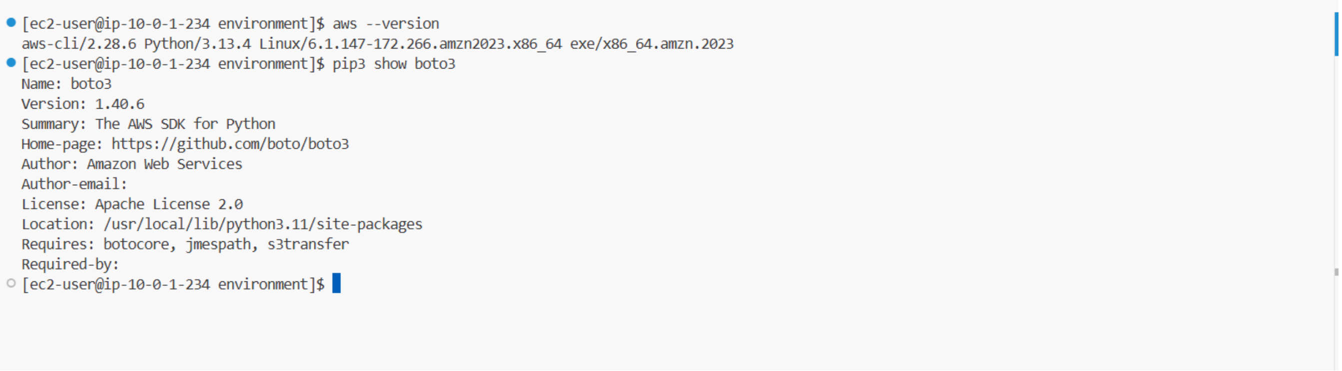
```
chmod +x ./resources/setup.sh && ./resources/setup.sh
```

The script will prompt you for the **IP address** by which your computer is known to the internet. Use www.whatismyip.com to discover this address and then paste the IPv4 address into the command prompt and finish running the script.



```
[ec2-user@ip-10-0-1-150 environment]$ chmod +x resources/setup.sh && resources/setup.sh
Please enter a valid IP address:
152.58.63.192
IP address:152.58.63.192
upload: resources/website/all_products_on_offer.json to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/all_products_on_offer.json
upload: resources/website/callback.html to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/callback.html
upload: resources/website/all_products.json to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/all_products.json
upload: resources/website/beans.json to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/beans.json
upload: resources/website/images/beans/excelsa.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/beans/excelsa.png
upload: resources/website/config.js to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/config.js
upload: resources/website/images/items/blueberry_bagel.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/blueberry_bagel.pr
g
upload: resources/website/images/items/apple_pie.jpeg to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/apple_pie.jpeg
upload: resources/website/images/items/blueberry_jelly_doughnut.jpeg to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/blueberr
y_jelly_doughnut.jpeg
upload: resources/website/images/beans/robusta.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/beans/robusta.png
upload: resources/website/images/items/boston_cream_doughnut.jpeg to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/boston_crea
m_doughnut.jpeg
upload: resources/website/images/expanded.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/expanded.png
upload: resources/website/images/beans/liberica.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/beans/liberica.png
upload: resources/website/images/items/apple_pie_slice.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/apple_pie_slice.pr
g
upload: resources/website/images/items/apple_pie.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/apple_pie.png
upload: resources/website/images/beans/arabica.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/beans/arabica.png
upload: resources/website/images/items/boston_cream_doughnut.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/boston_cream
_doughnut.png
upload: resources/website/favicon.ico to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/favicon.ico
upload: resources/website/images/items/apple_pie_slice.jpeg to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/apple_pie_slice.j
peg
upload: resources/website/images/items/cherry_pie.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/cherry_pie.png
upload: resources/website/images/items/blueberry_bagel.jpeg to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/blueberry_bagel.j
peg
upload: resources/website/images/items/blueberry_jelly_doughnut.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/blueberry
_jelly_doughnut.png
upload: resources/website/images/items/cherry_pie_slice.png to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/cherry_pie_slice.
png
upload: resources/website/images/items/cherry_pie.jpeg to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/cherry_pie.jpeg
upload: resources/website/images/items/cherry_pie_slice.jpeg to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/cherry_pie_slice
.jpeg
upload: resources/website/images/items/chocolate_chip_cupcake.jpeg to s3://c168617a4340248l11142234t1w184333714729-s3bucket-1wvevyyhv5l/images/items/chocolate_
```

8. Verify the AWS CLI version and also verify that the SDK for Python is installed.
   - Confirm that the AWS CLI is now at version 2 by running the **`aws --version`** command.
   - In the VS Code Bash terminal (at the bottom of the IDE), run the following command: **`pip3 show boto3`**

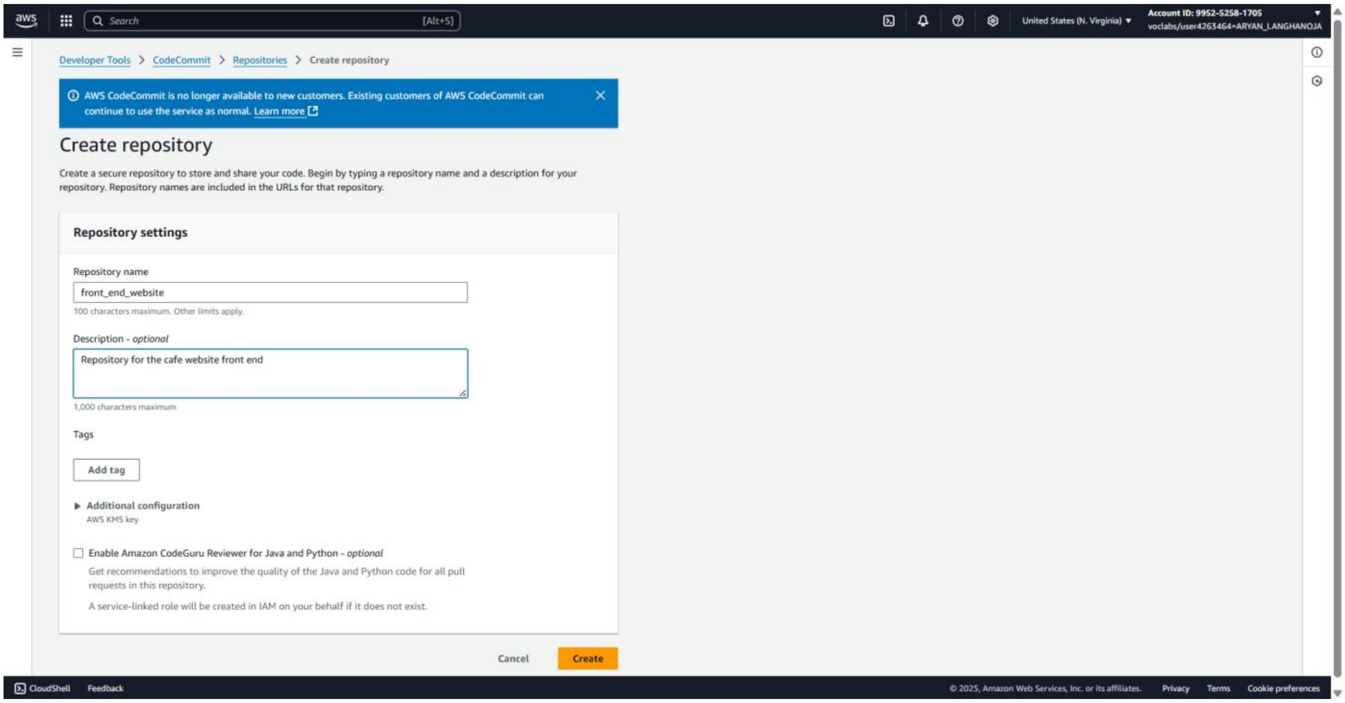


```
[ec2-user@ip-10-0-1-234 environment]$ aws --version
aws-cli/2.28.6 Python/3.13.4 Linux/6.1.147-172.266.amzn2023.x86_64 exe/x86_64.amzn.2023
[ec2-user@ip-10-0-1-234 environment]$ pip3 show boto3
Name: boto3
Version: 1.40.6
Summary: The AWS SDK for Python
Home-page: https://github.com/boto/boto3
Author: Amazon Web Services
Author-email:
License: Apache License 2.0
Location: /usr/local/lib/python3.11/site-packages
Requires: botocore, jmespath, s3transfer
Required-by:
[ec2-user@ip-10-0-1-234 environment]$
```

Task 2: Creating a CodeCommit repository

9. Create a CodeCommit repository to host the codebase.

- In the search bar at the top, search and select `CodeCommit` to open the AWS CodeCommit service console.
- From the navigation pane, choose **Create repository**.
- Configure the following settings:
  - **Repository name:** Enter `front_end_website`
  - **Description:** Enter `Repository for the cafe website front end`
  - Keep the rest of the default settings.
- Choose **Create**.



aws
Search
[Alt+S]
United States (N. Virginia) ▾
Account ID: 9952-5258-1705
voclabs/user4263464=ARYAN_LANGHANOJA
Developer Tools > CodeCommit > Repositories > Create repository
Cancel
Create
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

AWS CodeCommit is no longer available to new customers. Existing customers of AWS CodeCommit can continue to use the service as normal. Learn more

Create repository

Create a secure repository to store and share your code. Begin by typing a repository name and a description for your repository. Repository names are included in the URLs for that repository.

Repository settings

Repository name: front_end_website

100 characters maximum. Other limits apply.

Description - *optional*: Repository for the cafe website front end

1,000 characters maximum

Tags

Add tag

▸ Additional configuration

AWS KMS key

□ Enable Amazon CodeGuru Reviewer for Java and Python - *optional*

Get recommendations to improve the quality of the Java and Python code for all pull requests in this repository.

A service-linked role will be created in IAM on your behalf if it does not exist.

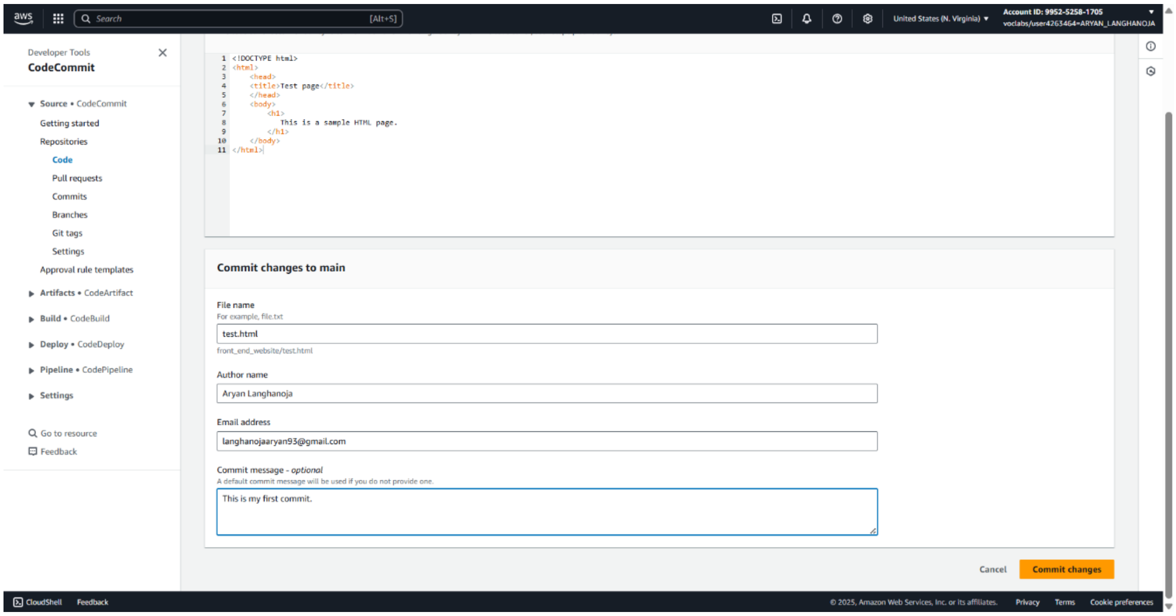
10. Create a test file.

- In the front_end_website section, choose Create file.
- Copy and paste the following code into the front_end_website text box:

```
<!DOCTYPE html>
<html>
    <head>
    <title>Test page</title>
    </head>
    <body>
        <h1>
           This is a sample HTML page.
```

```
            </h1>
        </body>
    </html>
```

- In the Commit changes to main section, configure the following settings:
- File name: Enter test.html
- Author name: Enter your name.
- Email address: Enter the same email address that you used in Task 1.
- Commit message: Enter This is my first commit.
- Choose Commit changes.



```
<!DOCTYPE html>
<html>
    <head>
    <title>Test page</title>
    </head>
    <body>
        <h1>
            This is a sample HTML page.
        </h1>
    </body>
</html>
```

Commit changes to main

File name
For example, file.txt
test.html
front_end_website/test.html

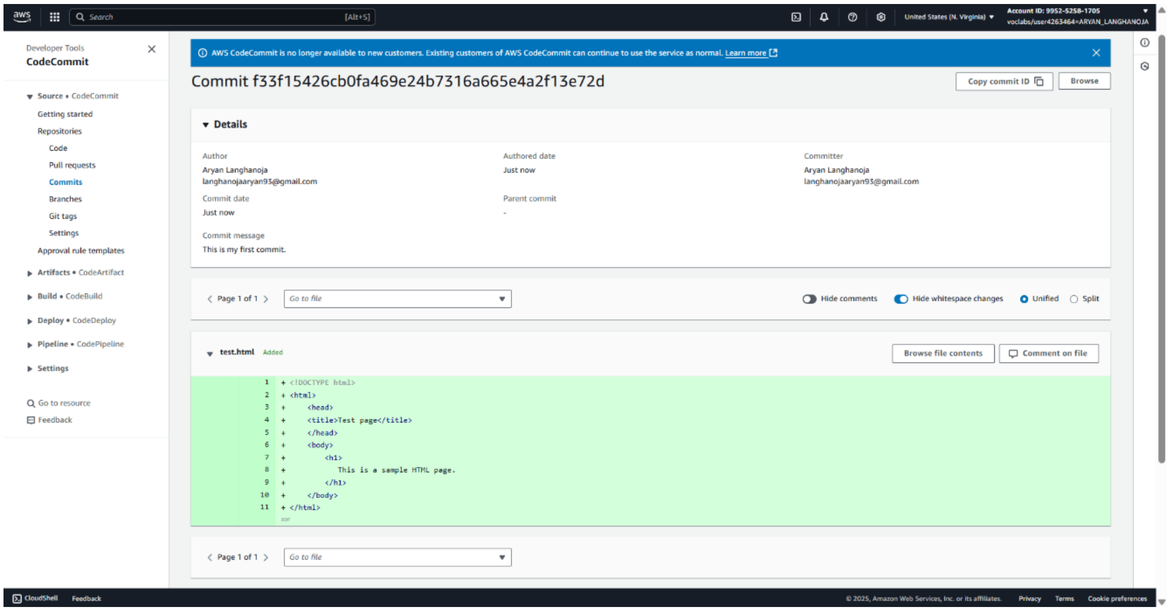
Author name
Aryan Langhanoja

Email address
langhanojaaryan93@gmail.com

Commit message - *optional*
A default commit message will be used if you do not provide one.
This is my first commit.

11. Review your commit.

- In the left navigation pane, under **Repositories**, choose **Commits**.
- Choose the link for the commit ID. Only one should be listed.
- Review the information about this commit.
- **Note:** On this page, you see the author of the commit, the commit message, the date of the commit, and the name and contents of the file that was added.



AWS CodeCommit is no longer available to new customers. Existing customers of AWS CodeCommit can continue to use the service as normal. Learn more

Commit f33f15426cb0fa469e24b7316a665e4a2f13e72d

Details

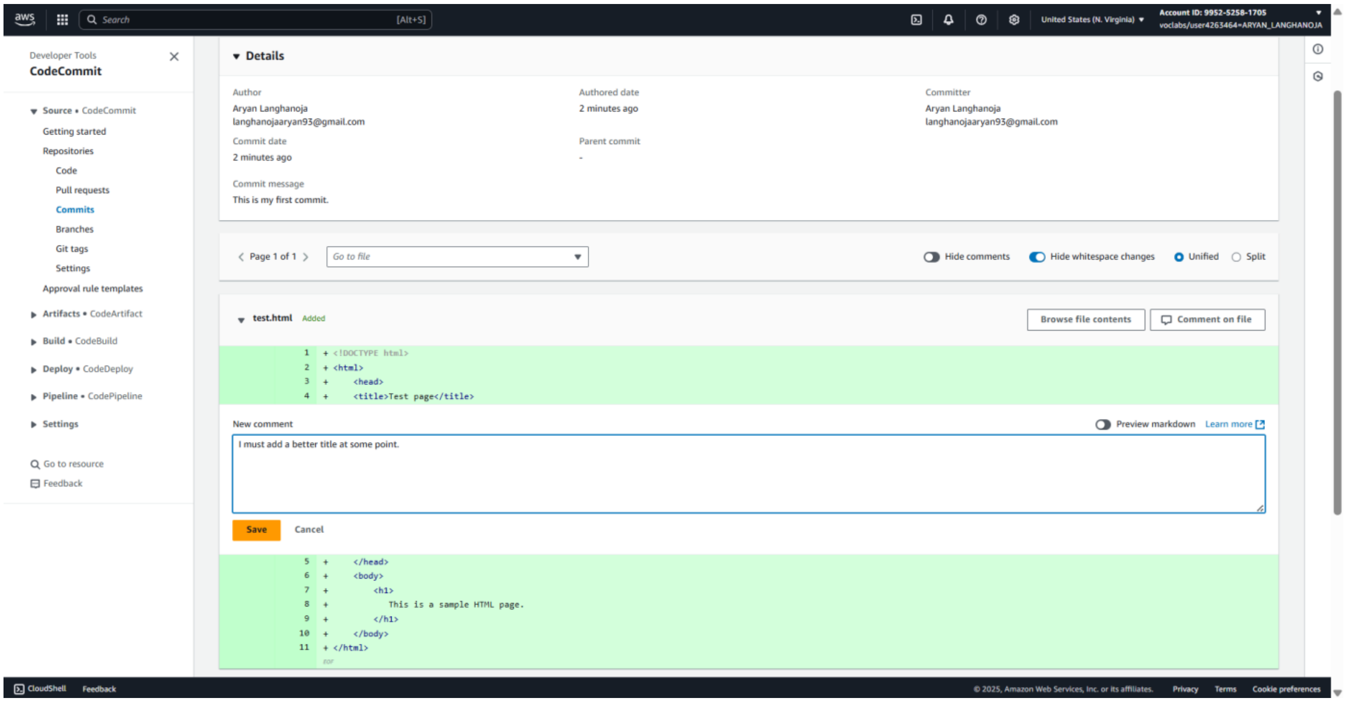
Author: Aryan Langhanoja, langhanojaaryan93@gmail.com
Authored date: Just now
Committer: Aryan Langhanoja, langhanojaaryan93@gmail.com
Commit date: Just now
Parent commit: -
Commit message: This is my first commit.

test.html Added

```
1  + <!DOCTYPE html>
2  + <html>
3  +     <head>
4  +     <title>Test page</title>
5  +     </head>
6  +     <body>
7  +         <h1>
8  +             This is a sample HTML page.
9  +         </h1>
10 +     </body>
11 + </html>
```

12. Add a comment to the committed file.

- In the **test.html** section, hover on the plus sign (+) icon on line 4.
- To the left of the line number, a comment icon appears. Choose the icon.
  - **Note:** The comment icon is a small box that contains an ellipsis.
- In the **New comment** text box, enter: `I must add a better title at some point.`
- Choose **Save**.



Details

Author: Aryan Langhanoja, langhanojaaryan93@gmail.com
Authored date: 2 minutes ago
Committer: Aryan Langhanoja, langhanojaaryan93@gmail.com
Commit date: 2 minutes ago
Parent commit: -
Commit message: This is my first commit.

test.html Added

```
1  + <!DOCTYPE html>
2  + <html>
3  +     <head>
4  +     <title>Test page</title>
```

New comment: I must add a better title at some point.

```
5  +     </head>
6  +     <body>
7  +         <h1>
8  +             This is a sample HTML page.
9  +         </h1>
10 +     </body>
11 + </html>
```

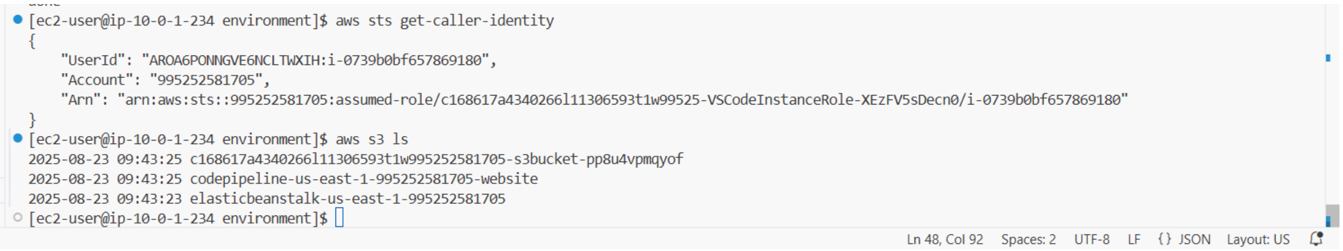
Task 3: Creating a pipeline to automate website updates

13. Return to the VS Code IDE.

- In the Explorer section, expand *Environment*, which is located in the upper-left corner.
- Expand the **resources** folder, and open the file named **cafe_website_front_end_pipeline.json**.
- This file defines configuration that will be used to deploy your new pipeline. Review the following code snippets to understand how the pipeline is configured.
- The following lines declare the AWS Identity and Access Management (IAM) role that will be associated with the pipeline.

14. Update the cafe_website_front_end_pipeline.json file:

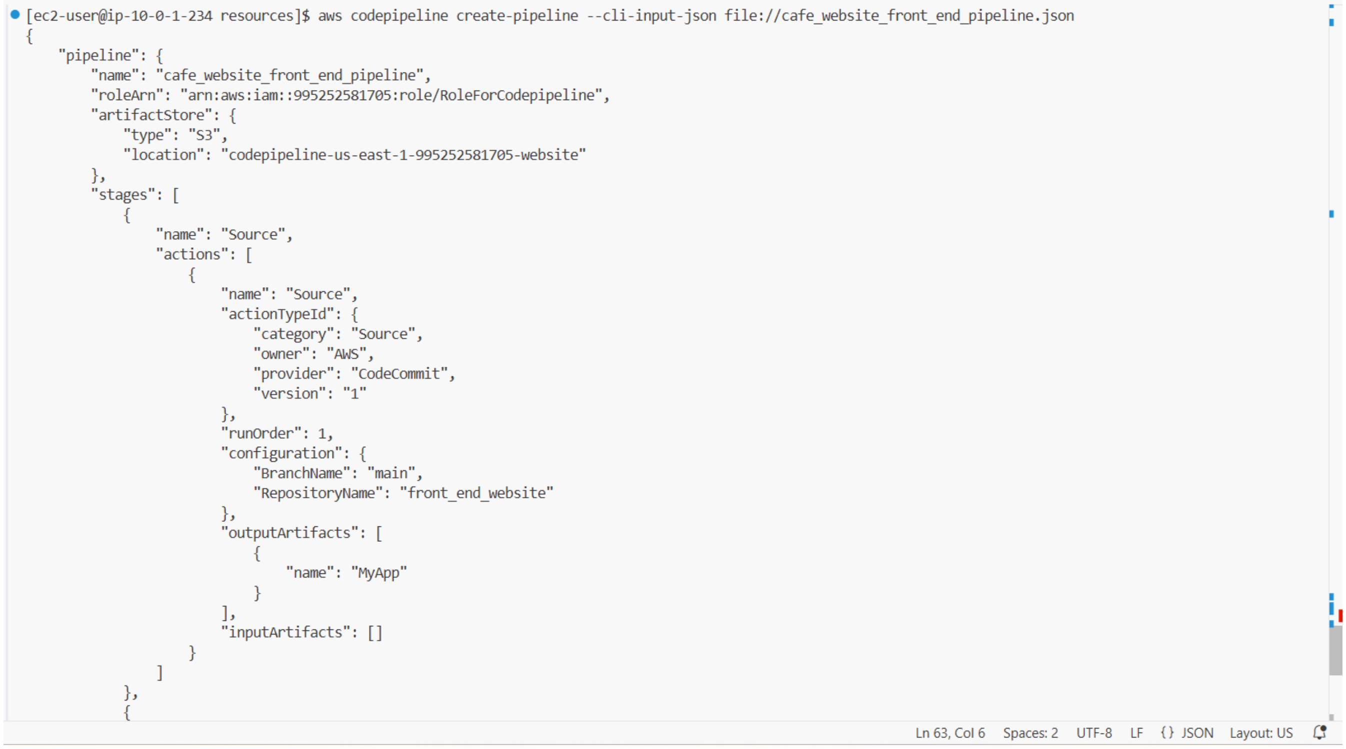
- Replace the two <*FMI_1*> placeholders with the your AWS account ID.
    - **Note:** To find your account ID, run the following command: `aws sts get-caller-identity`
- Replace the <*FMI_2*> placeholder with the name of the bucket that has *s3bucket* in the name.
    - **Note:** To retrieve a list of the S3 buckets in your account, run the following command: `aws s3 ls`
- From the navigation pane, choose menu, then choose **File > Save**.



```
[ec2-user@ip-10-0-1-234 environment]$ aws sts get-caller-identity
{
    "UserId": "AROA6PONNGVE6NCLTWXIH:i-0739b0bf657869180",
    "Account": "995252581705",
    "Arn": "arn:aws:sts::995252581705:assumed-role/c168617a4340266l11306593t1w99525-VSCodeInstanceRole-XEzFV5sDecn0/i-0739b0bf657869180"
}
[ec2-user@ip-10-0-1-234 environment]$ aws s3 ls
2025-08-23 09:43:25 c168617a4340266l11306593t1w995252581705-s3bucket-pp8u4vpmqyof
2025-08-23 09:43:25 codepipeline-us-east-1-995252581705-website
2025-08-23 09:43:23 elasticbeanstalk-us-east-1-995252581705
[ec2-user@ip-10-0-1-234 environment]$
```

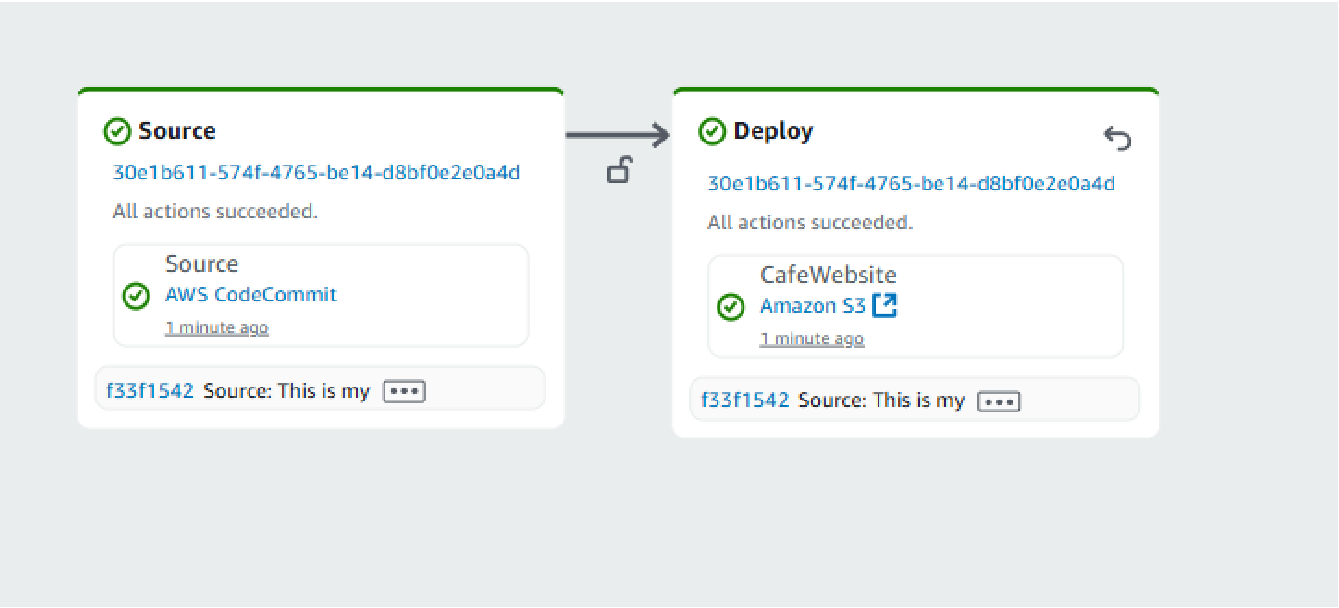
15. To create the pipeline, run the following commands.

```
cd ~/environment/resources
aws codepipeline create-pipeline --cli-input-json
file://cafe_website_front_end_pipeline.json
```



```
[ec2-user@ip-10-0-1-234 resources]$ aws codepipeline create-pipeline --cli-input-json file://cafe_website_front_end_pipeline.json
{
    "pipeline": {
        "name": "cafe_website_front_end_pipeline",
        "roleArn": "arn:aws:iam::995252581705:role/RoleForCodepipeline",
        "artifactStore": {
            "type": "S3",
            "location": "codepipeline-us-east-1-995252581705-website"
        },
        "stages": [
            {
                "name": "Source",
                "actions": [
                    {
                        "name": "Source",
                        "actionTypeId": {
                            "category": "Source",
                            "owner": "AWS",
                            "provider": "CodeCommit",
                            "version": "1"
                        },
                        "runOrder": 1,
                        "configuration": {
                            "BranchName": "main",
                            "RepositoryName": "front_end_website"
                        },
                        "outputArtifacts": [
                            {
                                "name": "MyApp"
                            }
                        ],
                        "inputArtifacts": []
                    }
                ]
            },
            {
```

16. Choose the **cafe_website_front_end_pipeline** hyperlink and review the pipeline status, as shown in the following image.



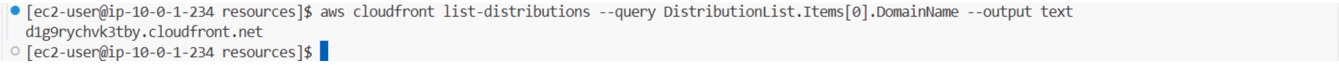
Source
30e1b611-574f-4765-be14-d8bf0e2e0a4d
All actions succeeded.
Source
AWS CodeCommit
1 minute ago
f33f1542 Source: This is my
Deploy
30e1b611-574f-4765-be14-d8bf0e2e0a4d
All actions succeeded.
CafeWebsite
Amazon S3
1 minute ago
f33f1542 Source: This is my

17. Verify the automated deployment.

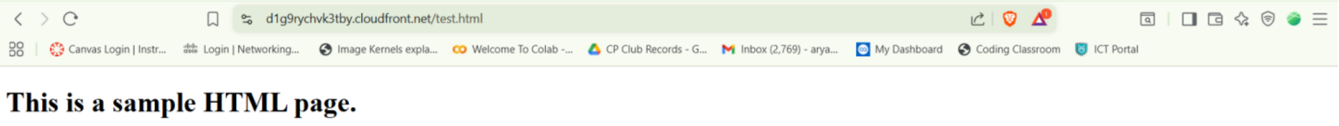
- Return to the VS Code IDE bash terminal.
- To find your Amazon CloudFront distribution domain name, run the following command:

```
aws cloudfront list-distributions --query
DistributionList.Items[0].DomainName --output text
```

- Update the following URL by replacing <*cloudfront_domain*> with the value that was returned by the previous command: https://<cloudfront_domain>/test.html
- The updated URL is similar to *https:// d1g9rychvk3tby.cloudfront.net/test.html*.
- Open a new browser tab, and enter the URL that you just created.
- You reach a sample webpage similar to the following:



```
● [ec2-user@ip-10-0-1-234 resources]$ aws cloudfront list-distributions --query DistributionList.Items[0].DomainName --output text
  d1g9rychvk3tby.cloudfront.net
○ [ec2-user@ip-10-0-1-234 resources]$ 
```



This is a sample HTML page.

Task 4: Cloning a repository in VS Code IDE

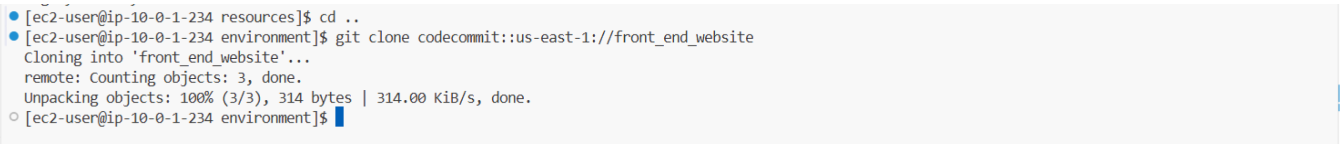
18. Retrieve the SSH clone URL for your repository.

- Navigate to the CodeCommit console.
- In the navigation pane, choose Repositories, then choose the repository created *front_end_website*.
- In the Clone URL column, choose Clone HTTPS(GRC) to copy the URL to your clipboard.

19. Clone your repository using the SSH URL.

- Return to the VS Code IDE terminal.
- Run the following command to clone the repository to your VS code IDE, replace the <<Clone URL>> with the value you copied.

```
cd ..
git clone <<Clone URL>>
```



```
● [ec2-user@ip-10-0-1-234 resources]$ cd ..
● [ec2-user@ip-10-0-1-234 environment]$ git clone codecommit::us-east-1://front_end_website
  Cloning into 'front_end_website'...
  remote: Counting objects: 3, done.
  Unpacking objects: 100% (3/3), 314 bytes | 314.00 KiB/s, done.
○ [ec2-user@ip-10-0-1-234 environment]$ 
```

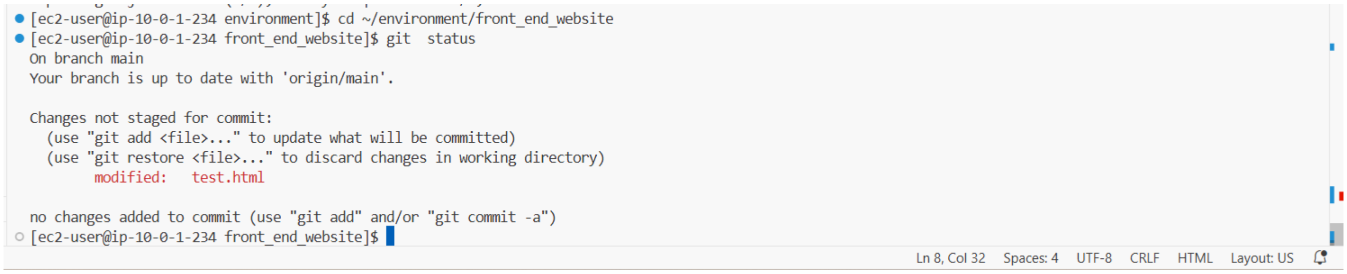
Task 5: Exploring the Git integration with the VS Code IDE

20. Explore repository branch management.

- Locate the branch icon, which is located in the lower-left corner of the IDE.
- Source Control option is opened.
- From the top, choose three dots to the right of *SOURCE CONTROL*.
- From the menu displayed, choose **Source Control Repositories**
- Your repository *front_end_website* is displayed.
- The IDE is currently set up to communicate with the **main** branch.
- Explore and edit the repository files.
- Remember that the repository was cloned to the local folder environment/front_end_website. You can open repository files in your IDE to view and edit them.
- From the explorer menu, expand the **front_end_website** folder to reveal the **test.html** file, which is shown in the following image.
- Open the **test.html** file and edit the page title on line 4. Replace the current title with the following text:
- Best test page ever.
- Save your changes.
- Number **1** now appears next to **branch** icon in the left pane. This indicates that changes have been made to the code that need to be committed to the branch.
- Hover over and you will notice a message **Source Control 1 pending change**
- **Note:** This is because you made a change to the file but it is not yet committed to thr repository main branch.

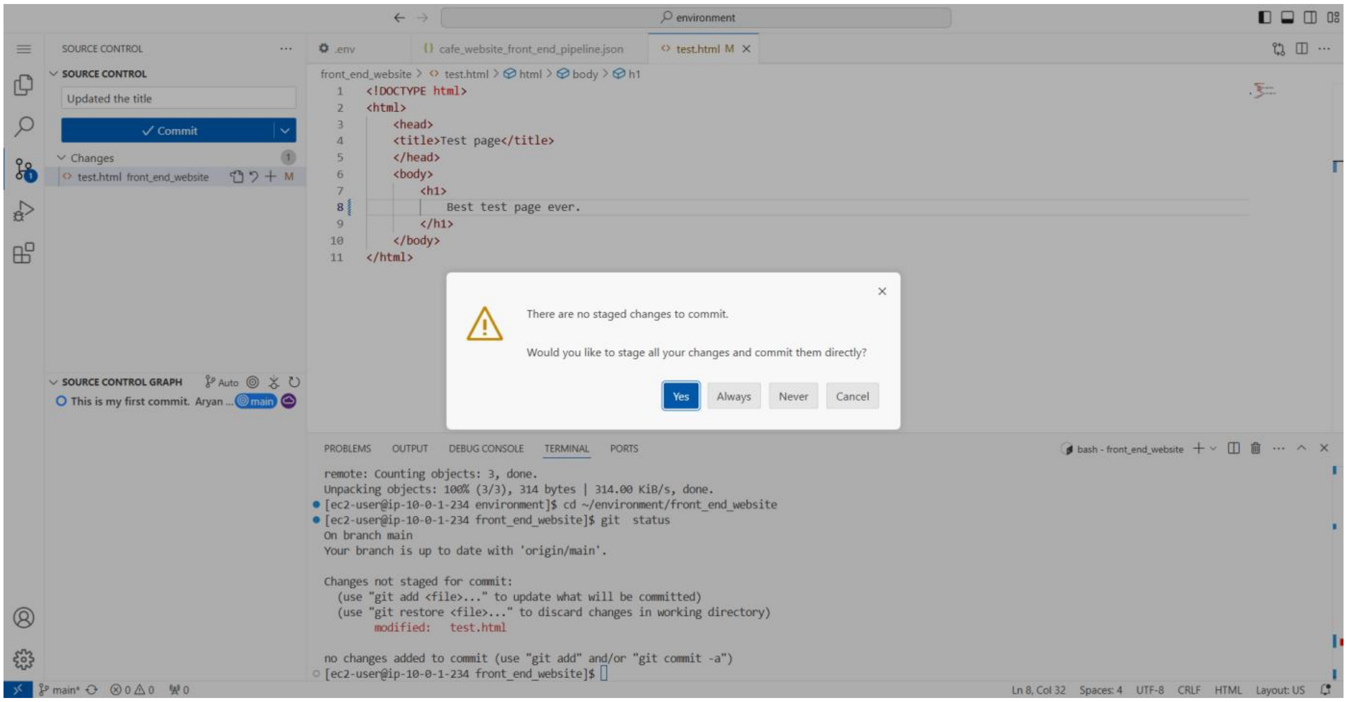
21. Commit your changes to the **main** branch.

```
cd ~/environment/front_end_website
git status
```



```
● [ec2-user@ip-10-0-1-234 environment]$ cd ~/environment/front_end_website
● [ec2-user@ip-10-0-1-234 front_end_website]$ git  status
  On branch main
  Your branch is up to date with 'origin/main'.

  Changes not staged for commit:
    (use "git add <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
          modified:   test.html

  no changes added to commit (use "git add" and/or "git commit -a")
○ [ec2-user@ip-10-0-1-234 front_end_website]$
```

- On the Commit button, choose *more actions* which is located to the right.
- Choose *Commit & Push*, then choose Yes.
- This will push the code and trigger the code pipeline which deploys new version of the file to website.

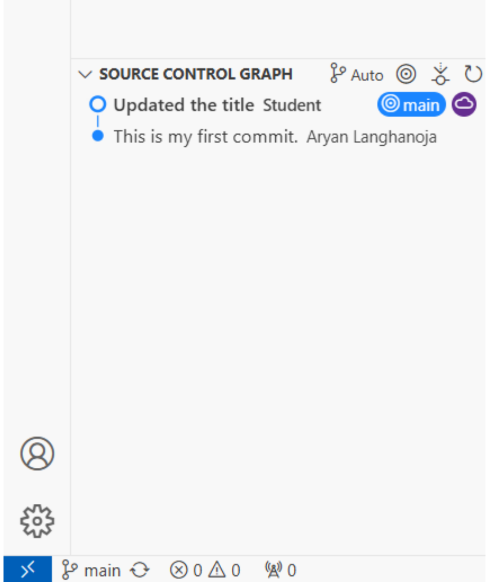


```
<!DOCTYPE html>
<html>
    <head>
    <title>Test page</title>
    </head>
    <body>
        <h1>
            Best test page ever.
        </h1>
    </body>
</html>
```

There are no staged changes to commit.

Would you like to stage all your changes and commit them directly?

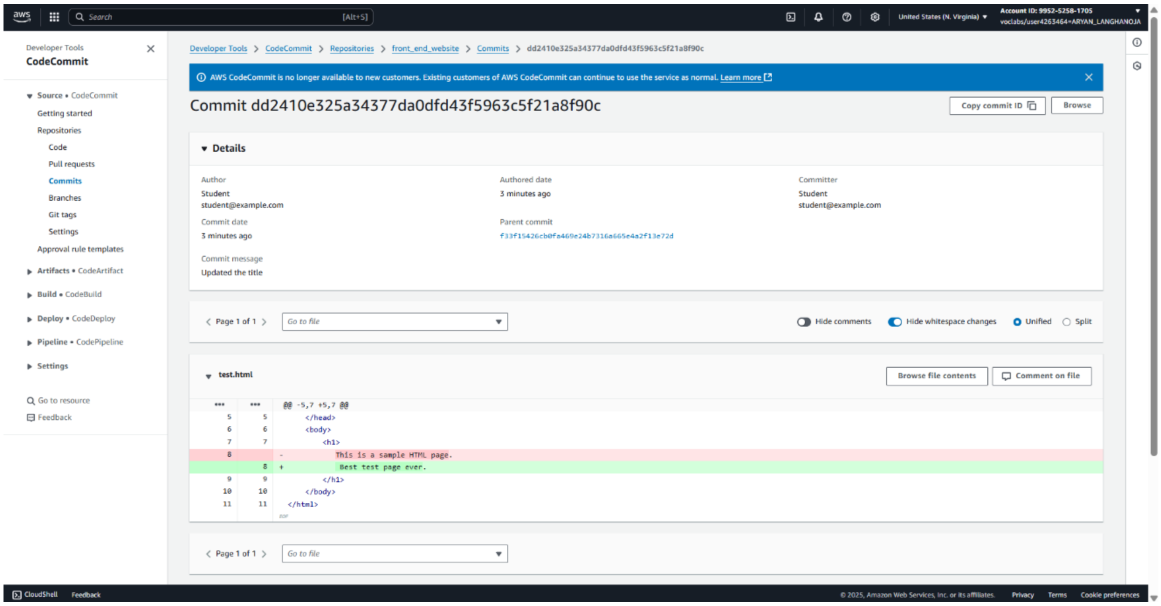
```
remote: Counting objects: 3, done.
Unpacking objects: 100% (3/3), 314 bytes | 314.00 KiB/s, done.
[ec2-user@ip-10-0-1-234 environment]$ cd ~/environment/front_end_website
[ec2-user@ip-10-0-1-234 front_end_website]$ git  status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   test.html

no changes added to commit (use "git add" and/or "git commit -a")
[ec2-user@ip-10-0-1-234 front_end_website]$
```



SOURCE CONTROL GRAPH
Auto
Updated the title Student
main
This is my first commit. Aryan Langhanoja
main

22. Review the changes in CodeCommit.

- Navigate to the CodeCommit console.
- Choose the **front_end_website** repository link.
- In the navigation pane, under **Repositories**, choose **Commits**.
- Choose the link for the commit ID with the most recent commit date.
- Go to the **test.html** section, and notice the highlighted lines, which are shown in the following image.



AWS CodeCommit is no longer available to new customers. Existing customers of AWS CodeCommit can continue to use the service as normal. Learn more

Commit dd2410e325a34377da0dfd43f5963c5f21a8f90c

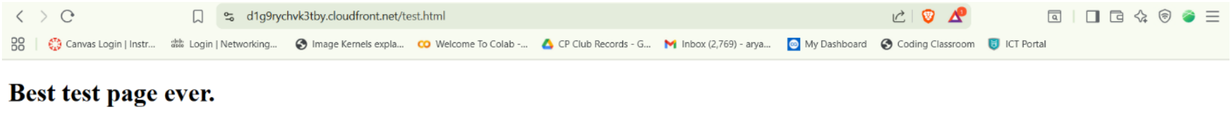
Author	Authored date	Committer
Student student@example.com	3 minutes ago	Student student@example.com
Commit date	Parent commit	
3 minutes ago	f33f15426cb0fa469e24b7316a665e4a2f13e72d	

Commit message
Updated the title

```
@@ -5,7 +5,7 @@
 5   5        </head>
 6   6        <body>
 7   7            <h1>
 8      -             This is a sample HTML page.
     8  +             Best test page ever.
 9   9            </h1>
10  10        </body>
11  11    </html>
```

23. Now, return to the tab that contains the café website and refresh the page.

- Notice that the browser tab title has changed, as shown in the following image. This proves that the pipeline deployed the changes that you committed from your local repository.

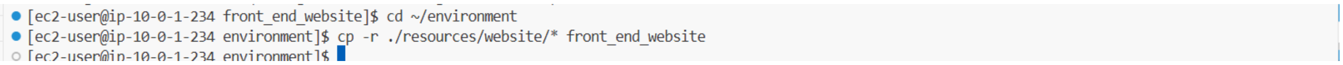


Best test page ever.

24. Return to the VS Code IDE tab.
25. On the explorer, Under *Environment* folder, expand the front_end_website folder and delete the test.html file, as shown in the following image.
26. In the VS Code IDE bash terminal, run the following commands.

The second command will copy all of the contents under the *website* folder to the *front_end_website* folder.

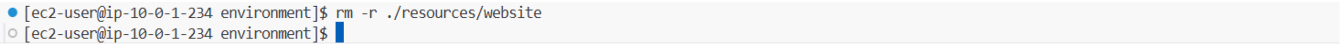
```
cd ~/environment
cp -r ./resources/website/* front_end_website
```



```
● [ec2-user@ip-10-0-1-234 front_end_website]$ cd ~/environment
● [ec2-user@ip-10-0-1-234 environment]$ cp -r ./resources/website/* front_end_website
○ [ec2-user@ip-10-0-1-234 environment]$ 
```

27. To delete the *website* folder, so that there is one source of truth, run the following command.

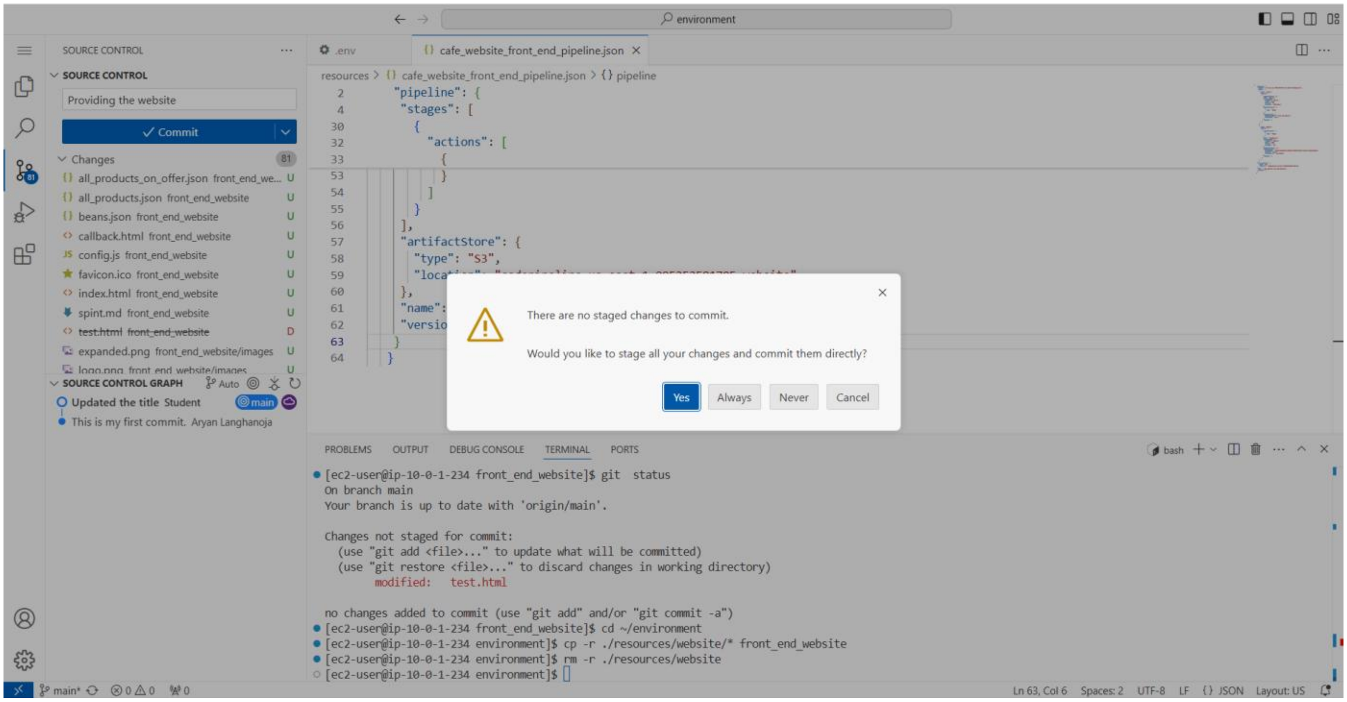
```
rm -r ./resources/website
```



```
● [ec2-user@ip-10-0-1-234 environment]$ rm -r ./resources/website
○ [ec2-user@ip-10-0-1-234 environment]$ 
```

28. Commit your changes.

- Choose the *Branch* icon (now it is showing number of new files to be pushed).
- Enter the following commit message: Providing the website
- On the **Commit** button, choose *more actions* which is located to the right.
- Choose *Commit & Push*, then choose **Yes**.
- To the right of **front_end_website**, choose the options icon and choose **Commit**.
- The **Source control Graph** in the lower pane shows history of changes done to the repository.



environment
SOURCE CONTROL
SOURCE CONTROL
Providing the website
✓ Commit
Changes 81
all_products_on_offer.json front_end_we... U
all_products.json front_end_website U
beans.json front_end_website U
callback.html front_end_website U
config.js front_end_website U
favicon.ico front_end_website U
index.html front_end_website U
spint.md front_end_website U
~~test.html front_end_website~~ D
expanded.png front_end_website/images U
logo.png front_end_website/images U
SOURCE CONTROL GRAPH Auto
Updated the title Student main
This is my first commit. Aryan Langhanoja
.env
cafe_website_front_end_pipeline.json
resources > cafe_website_front_end_pipeline.json > pipeline
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
bash
Ln 63, Col 6 Spaces: 2 UTF-8 LF JSON Layout: US
main* 0 0 0

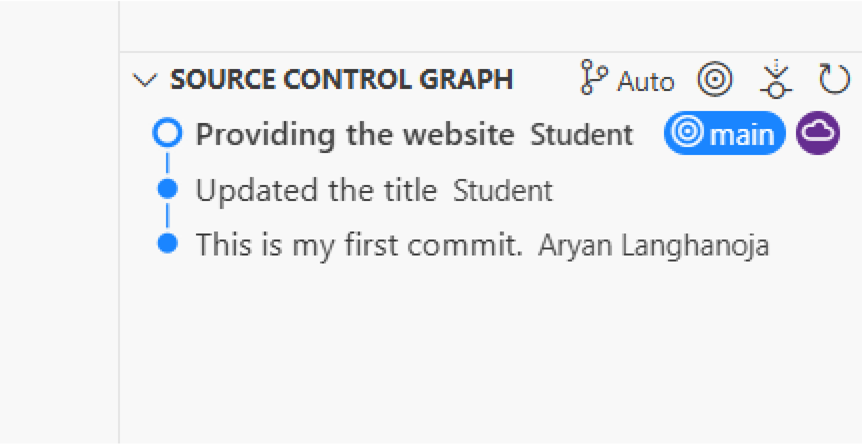
```
  "pipeline": {
    "stages": [
      {
        "actions": [
          {
          }
        ]
      }
    ],
    "artifactStore": {
      "type": "S3",
      "loca
    },
    "name":
    "versio
  }
}
```

There are no staged changes to commit.

Would you like to stage all your changes and commit them directly?

Yes Always Never Cancel

```
● [ec2-user@ip-10-0-1-234 front_end_website]$ git  status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   test.html

no changes added to commit (use "git add" and/or "git commit -a")
● [ec2-user@ip-10-0-1-234 front_end_website]$ cd ~/environment
● [ec2-user@ip-10-0-1-234 environment]$ cp -r ./resources/website/* front_end_website
● [ec2-user@ip-10-0-1-234 environment]$ rm -r ./resources/website
○ [ec2-user@ip-10-0-1-234 environment]$ 
```



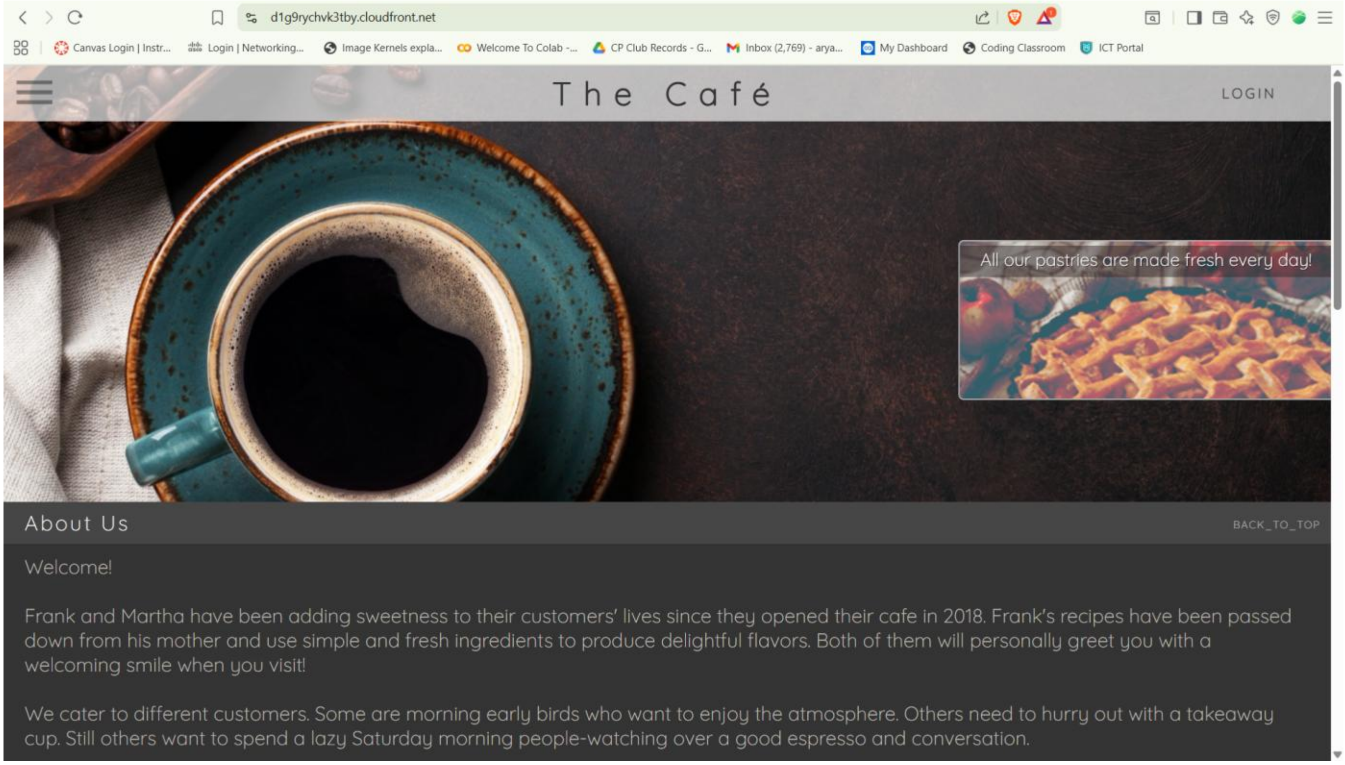
Providing the website Student

Updated the title Student

This is my first commit. Aryan Langhanoja

29. Update the URL by removing test.html from the address, and then submit the updated URL.

- The updated address is similar to the following example: `https://d1g9rychvk3tby.cloudfront.net`
- The browser now displays the café website. Well done!



About Us

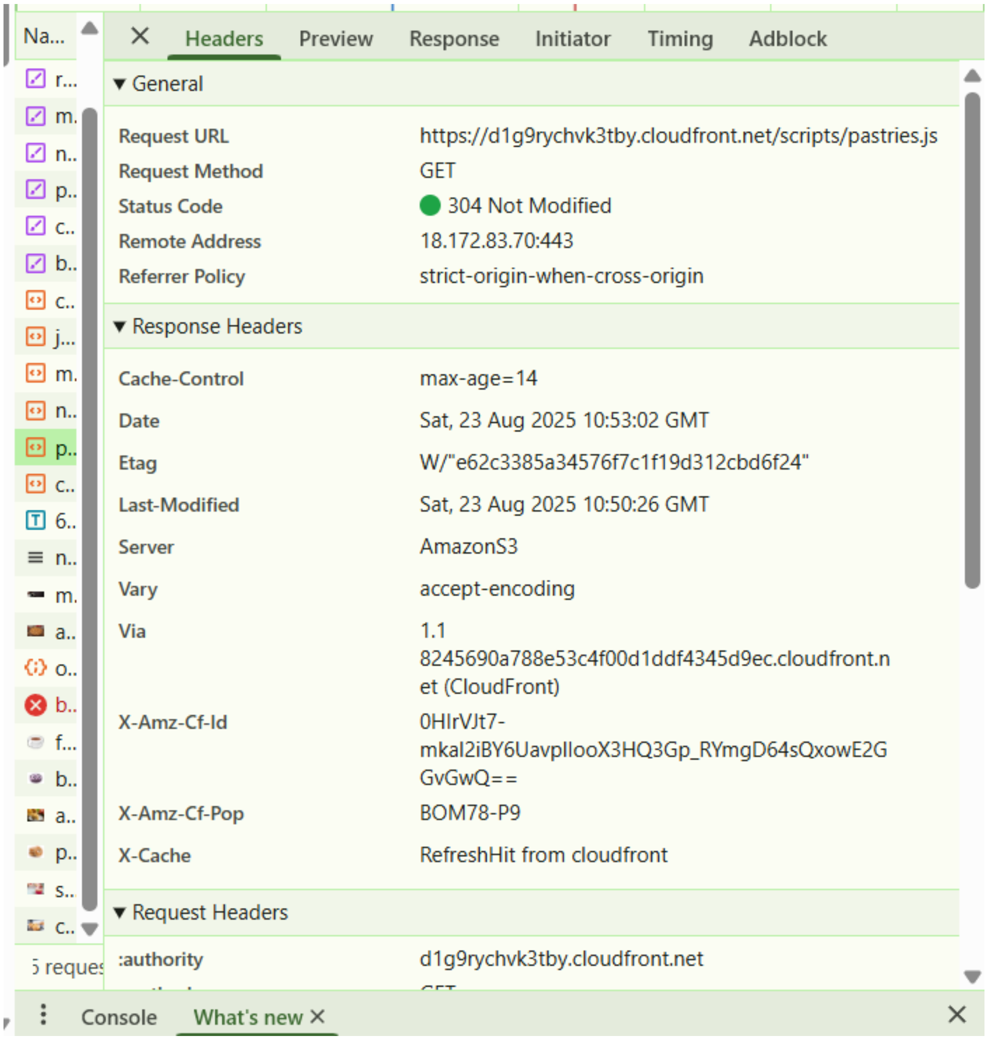
Welcome!

Frank and Martha have been adding sweetness to their customers' lives since they opened their cafe in 2018. Frank's recipes have been passed down from his mother and use simple and fresh ingredients to produce delightful flavors. Both of them will personally greet you with a welcoming smile when you visit!

We cater to different customers. Some are morning early birds who want to enjoy the atmosphere. Others need to hurry out with a takeaway cup. Still others want to spend a lazy Saturday morning people-watching over a good espresso and conversation.

30. Verify that your pipeline applied the cache-control setting, which was configured in Task 3.

- Remain on the café website page, and open your browser's developer tools.
    - **Note:** To open the developer tools, open your browser's context menu (right-click) and choose **Inspect**.
- Choose the **Network** tab, and then refresh the webpage.
- Choose **pastries.js**.
- Choose the **Headers** tab, and locate the **Response Headers** section, as shown in the following image.



▼ General

Request URL	https://d1g9rychvk3tby.cloudfront.net/scripts/pastries.js
Request Method	GET
Status Code	● 304 Not Modified
Remote Address	18.172.83.70:443
Referrer Policy	strict-origin-when-cross-origin

▼ Response Headers

Cache-Control	max-age=14
Date	Sat, 23 Aug 2025 10:53:02 GMT
Etag	W/"e62c3385a34576f7c1f19d312cbd6f24"
Last-Modified	Sat, 23 Aug 2025 10:50:26 GMT
Server	AmazonS3
Vary	accept-encoding
Via	1.1 8245690a788e53c4f00d1ddf4345d9ec.cloudfront.net (CloudFront)
X-Amz-Cf-Id	0HIrVJt7-mkal2iBY6UavpllooX3HQ3Gp_RYmgD64sQxowE2GGvGwQ==
X-Amz-Cf-Pop	BOM78-P9
X-Cache	RefreshHit from cloudfront

▼ Request Headers

:authority	d1g9rychvk3tby.cloudfront.net

Conclusion:-

- Connected to VS Code IDE using given LabIDE URL and password.
- Downloaded required lab files using:

```
wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-
ACCDEV-2-91558/13-lab-ci-cd/code.zip -P /home/ec2-user/environment
```

- Extracted files with:

```
unzip code.zip
```

- Upgraded AWS CLI and configured setup using:

```
chmod +x ./resources/setup.sh && ./resources/setup.sh
```

- Verified installation with:

```
aws --version
pip3 show boto3
```

- Created a CodeCommit repository named front_end_website and committed a test file (test.html).
- Added and reviewed comments in the commit history.
- Updated the pipeline configuration file (cafe_website_front_end_pipeline.json) with AWS Account ID and S3 bucket.
- Created the pipeline with:

```
cd ~/environment/resources
aws codepipeline create-pipeline --cli-input-json
file://cafe_website_front_end_pipeline.json
```

- Verified automated deployment by retrieving CloudFront domain:

  ```
  aws cloudfront list-distributions --query
  DistributionList.Items[0].DomainName --output text
  ```

- Cloned the repository using:

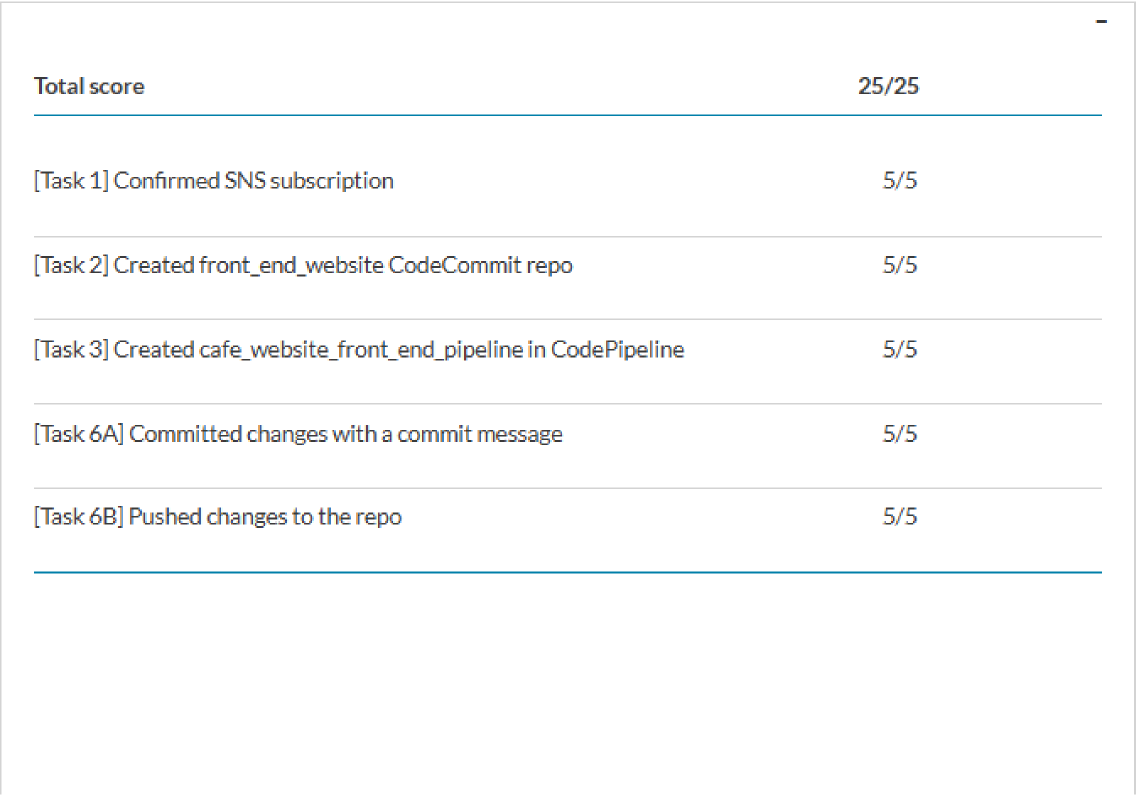
  ```
  git clone <<Clone URL>>
  ```

- Edited and committed changes (test.html) using Git integration in VS Code IDE.
- Verified updates reflected on the website via CloudFront URL.
- Replaced old website files with updated ones using:

```
cd ~/environment
cp -r ./resources/website/* front_end_website
rm -r ./resources/website
```

- Committed and pushed full website content with message “Providing the website”.
- Verified final deployment by accessing the CloudFront URL (without test.html).
- Checked cache-control settings using browser developer tools under Network → Headers.

Result :-



Total score	25/25
[Task 1] Confirmed SNS subscription	5/5
[Task 2] Created front_end_website CodeCommit repo	5/5
[Task 3] Created cafe_website_front_end_pipeline in CodePipeline	5/5
[Task 6A] Committed changes with a commit message	5/5
[Task 6B] Pushed changes to the repo	5/5