Name:Shejal Nilesh Tiwari Div/Roll No: D15C/57

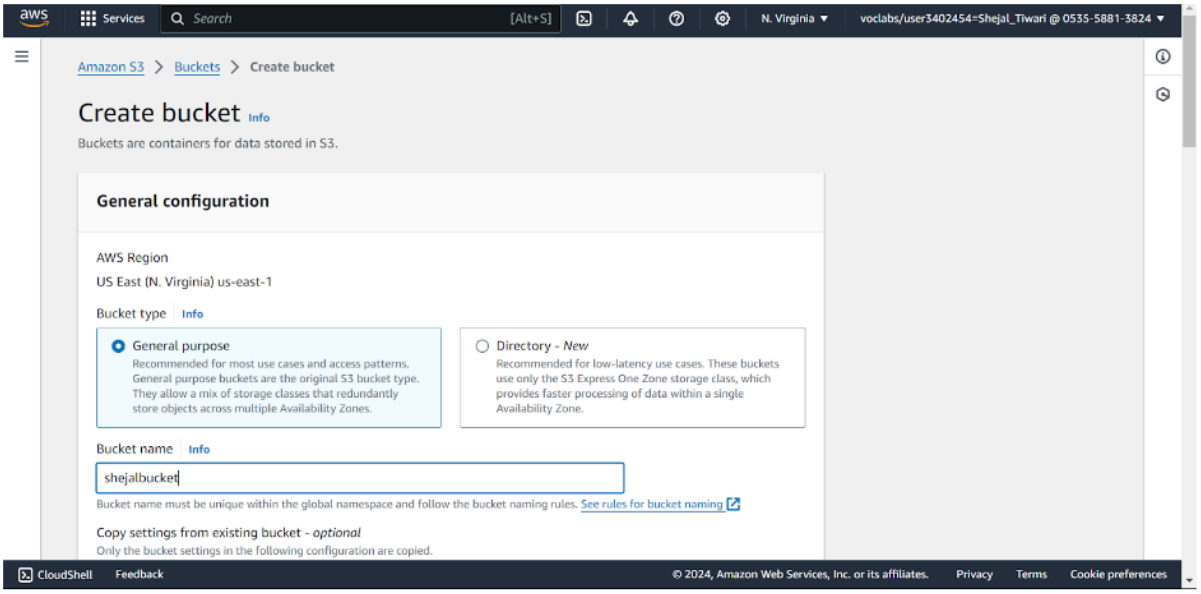
EXPERIMENT 1A

Step 1: Log in to AWS Management Console
Go to AWS Management Console and log in with your AWS account credentials.

Step 2: Navigate to S3 Service
In the AWS Management Console, search for "S3" in the search bar at the top, and click on "S3" under Services.

Step 3: Create a New Bucket
Click on the "Create bucket" button.
Fill in the details:

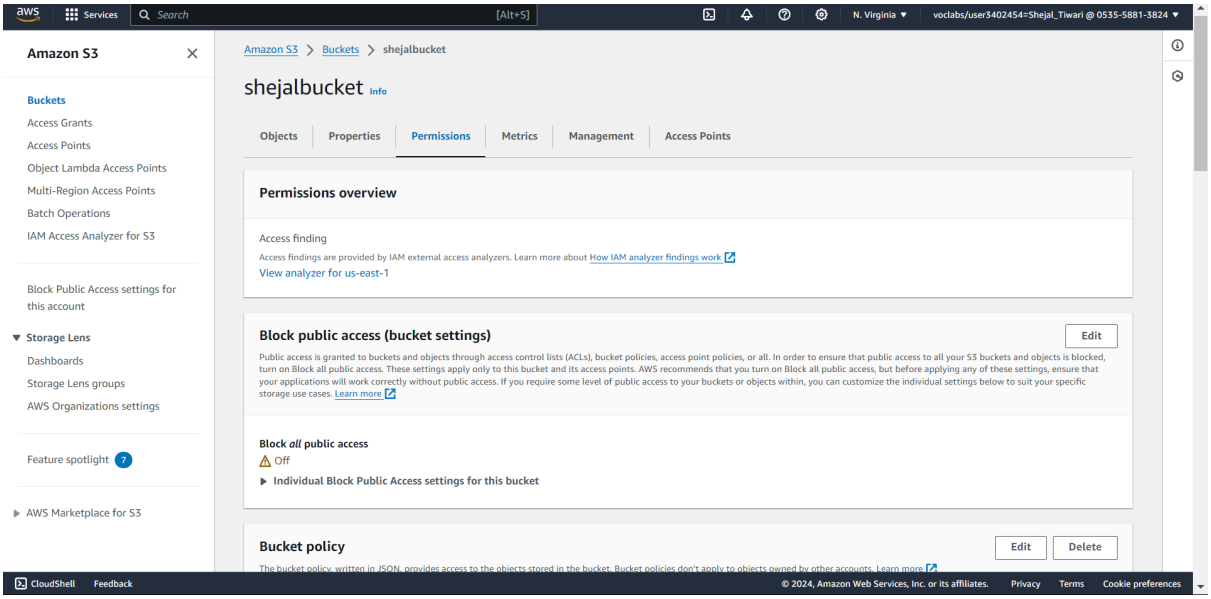
- Bucket Name: Enter a globally unique name for your bucket. The name must be DNS-compliant.
- AWS Region: Select the region where you want to create the bucket. Choose a region closest to you or your users for better performance.



Step 4: Configure Bucket Options

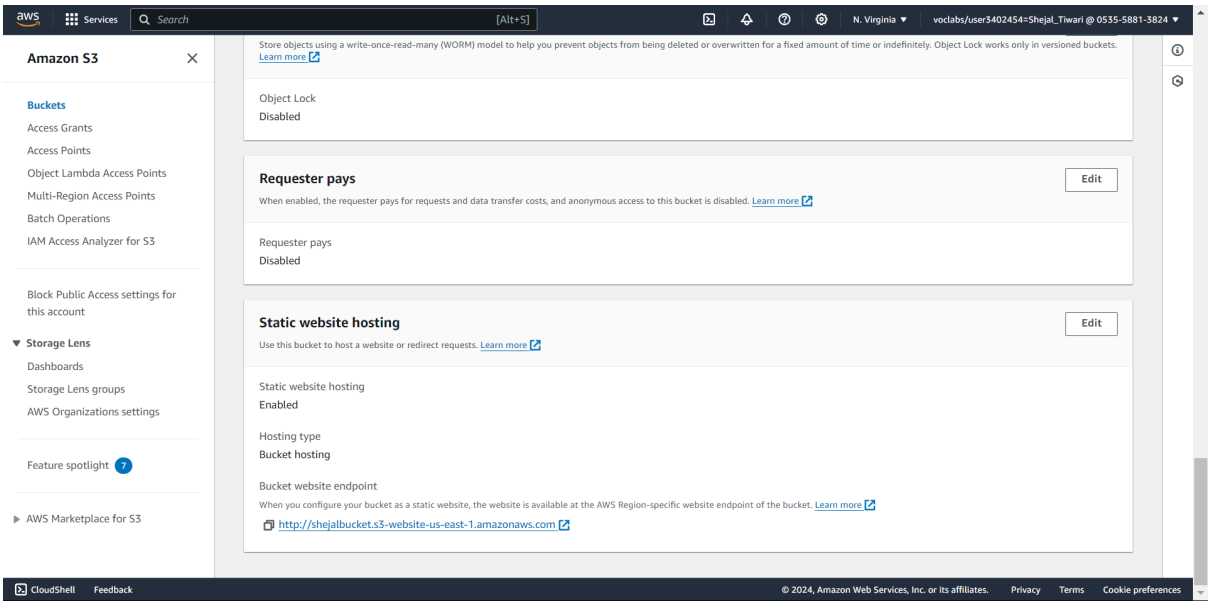
Object Ownership: Decide if you want to disable or enable object ownership. By default, it's set to "Bucket owner preferred."

Block Public Access Settings: You can choose to block all public access to your bucket or configure specific rules. AWS recommends blocking public access unless you specifically need it to be public.



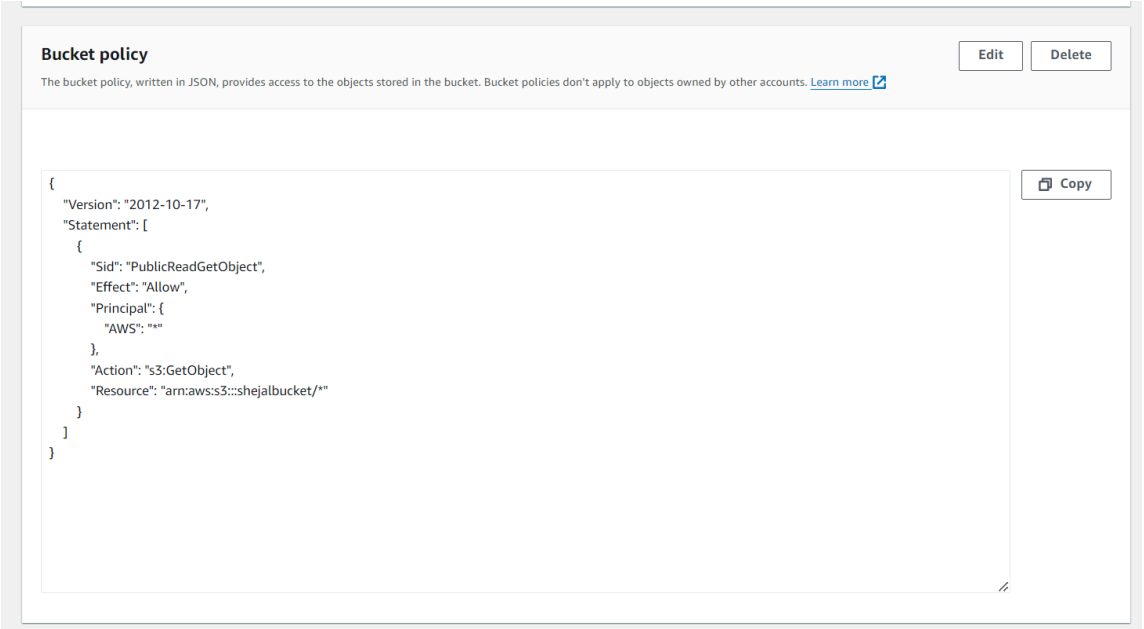
Bucket Versioning: Enable versioning if you want to keep multiple versions of objects in your bucket. This is useful for backup and recovery.

Tags: You can add tags to your bucket to organize and track its costs.



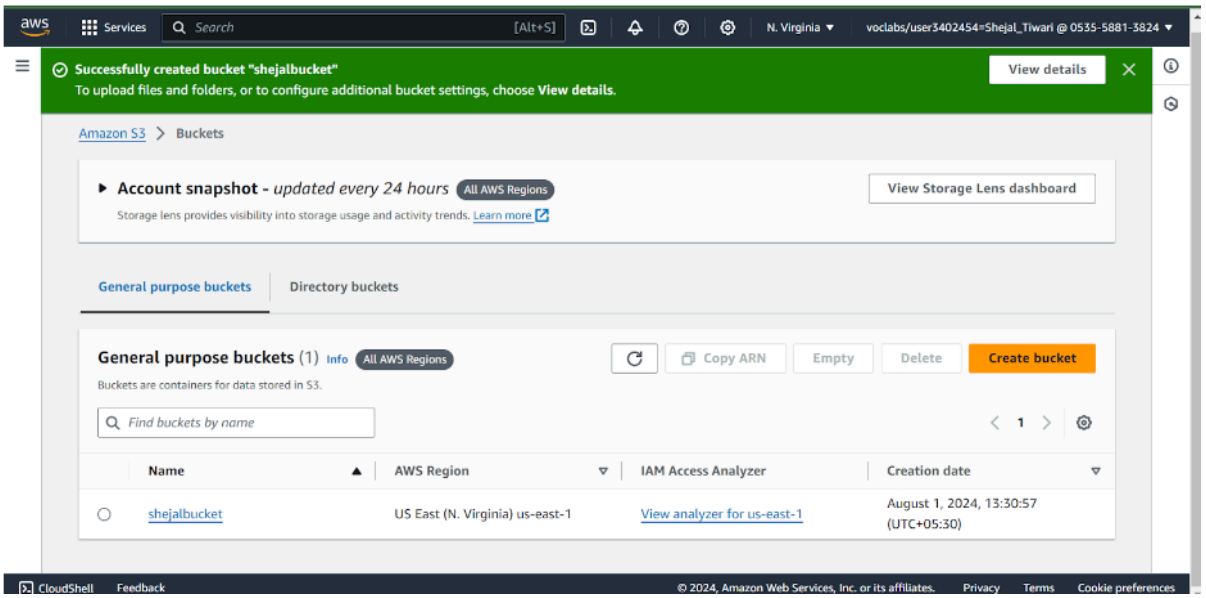
Step 5: Set Policy and Permissions

1. Set permissions based on your requirements. You can keep the default settings for private access or modify them for public or specific user access.



Step 6: Review and Create

Review all the settings you've configured.
If everything looks good, click the "Create bucket" button.



Step 1: Install XAMPP

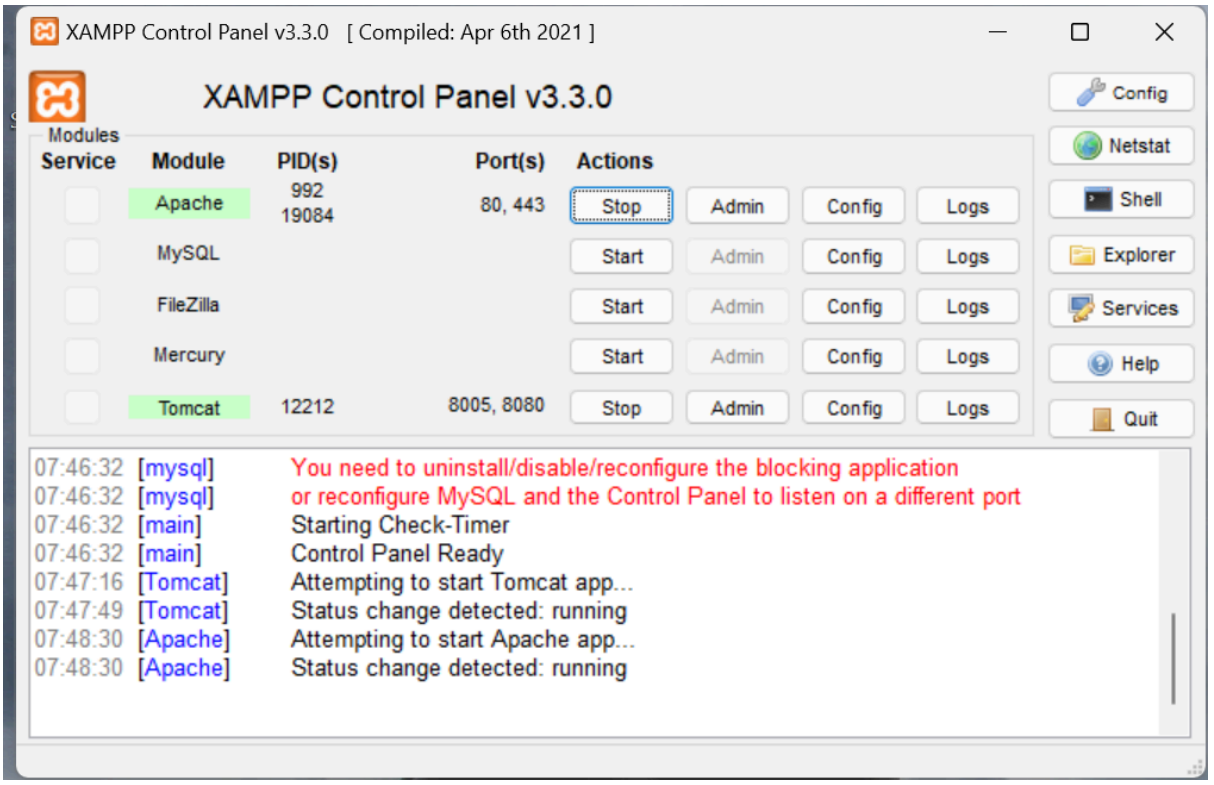
1. **Download XAMPP**: If you haven't already installed XAMPP, you can download it from the official website.
2. **Install XAMPP**: Run the installer and follow the instructions to install XAMPP on your machine.

To host a PHP file on the XAMPP server, follow these steps:

Step 1: Install XAMPP

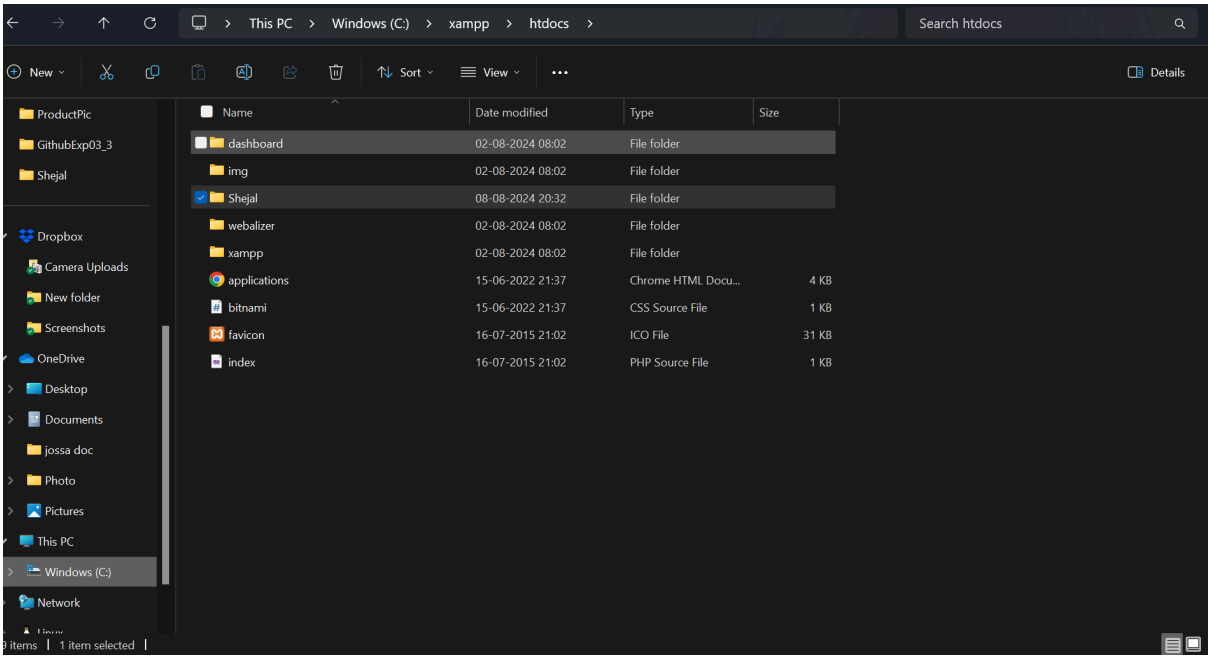
1. **Download XAMPP**: If you haven't already installed XAMPP, you can download it from the official website.
2. **Install XAMPP**: Run the installer and follow the instructions to install XAMPP on your machine.

Step 2: Start Apache Server

1. **Open XAMPP Control Panel**: After installation, open the XAMPP Control Panel.
2. **Start Apache**: Click the "Start" button next to Apache. The status should turn green, indicating that the Apache server is running.
3. 

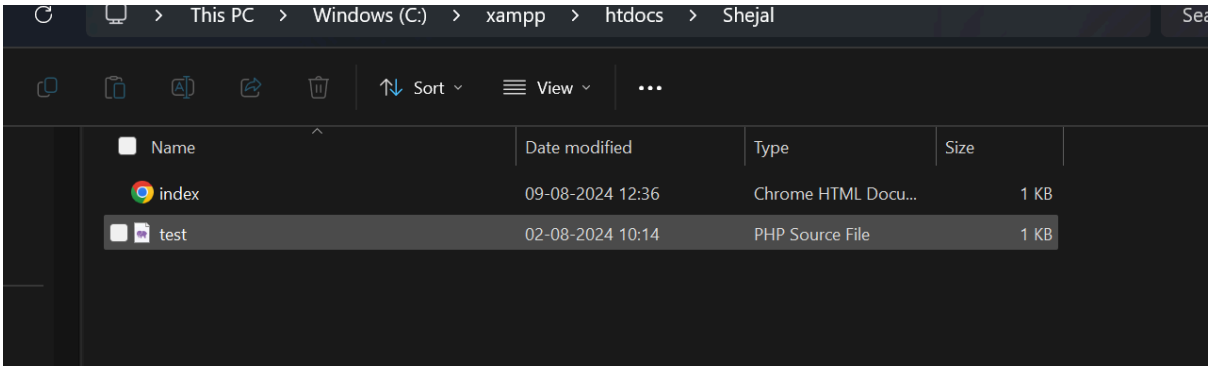
Step 3: Place PHP Files in the htdocs Folder

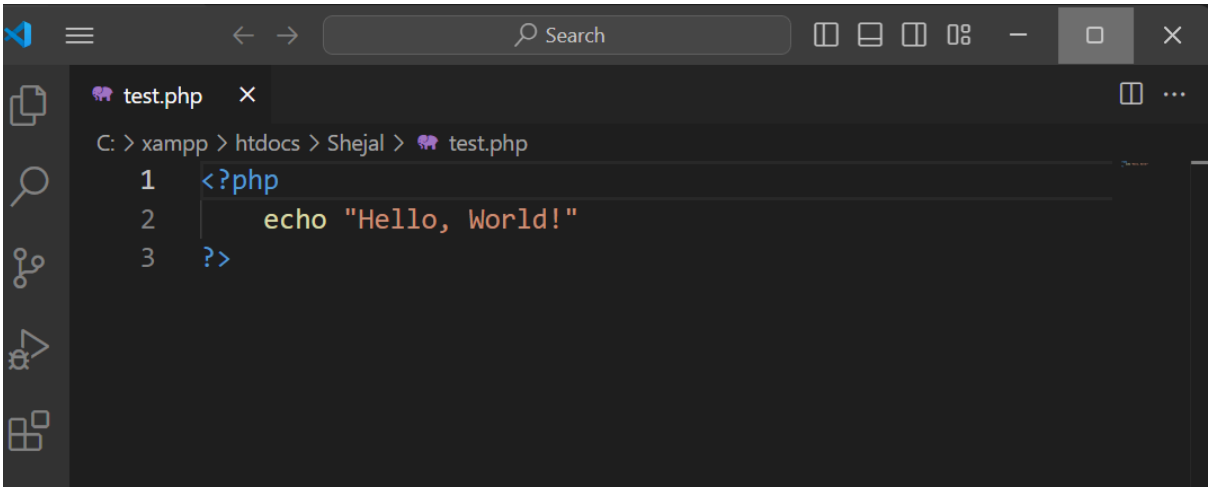
1. **Navigate to XAMPP Directory**: Go to the directory where you installed XAMPP (e.g., C:\xampp on Windows).
2. **Open htdocs Folder**: Inside the XAMPP folder, you'll find a folder named htdocs. This is the root directory where you should place your PHP files.
3. **Create a New Folder (Optional)**: You can create a new folder inside htdocs to organize your project (e.g., C:\xampp\htdocs\myproject).
4. **Place PHP File**: Copy your PHP file (e.g., index.php) into the htdocs directory or your project folder.



Step 4: Access the PHP File via Browser

1. **Open Web Browser**: Open any web browser.
2. **Access the File**: In the address bar, type the following URL:
    - If you placed the file directly in htdocs, use: http://localhost/filename.php
    - If you placed the file inside a folder, use: http://localhost/foldername/filename.php
    - For example, if your file is index.php inside myproject folder, the URL would be: http://localhost/myproject/index.php





Step 5: View the Output

- The PHP file will be executed on the server, and the output will be displayed in your browser.

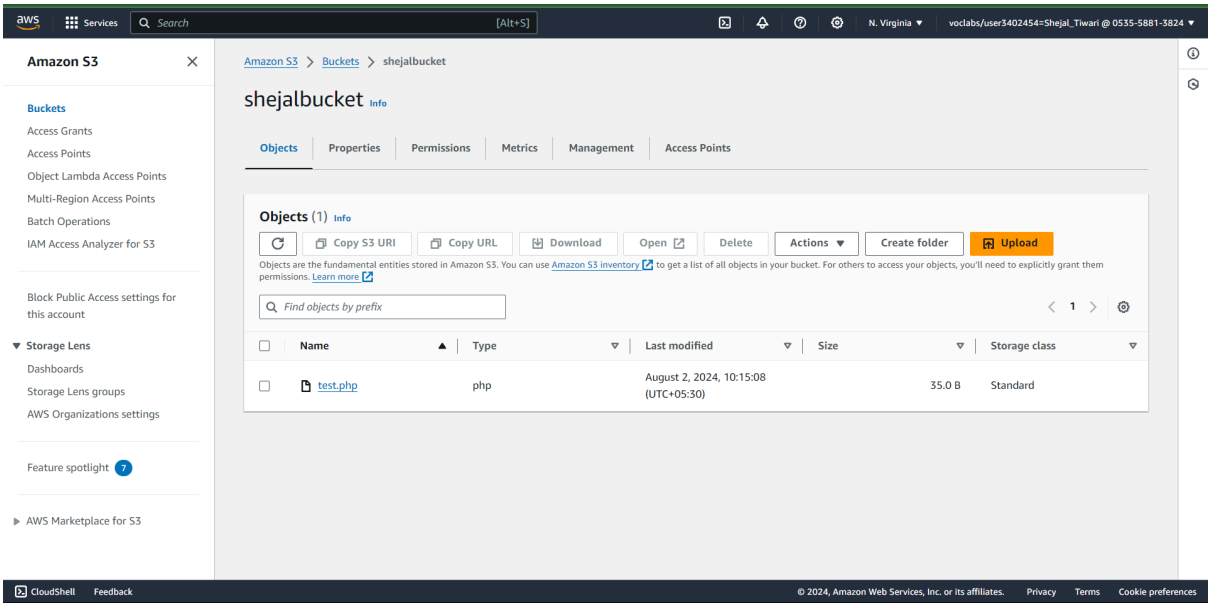
Additional Tips:

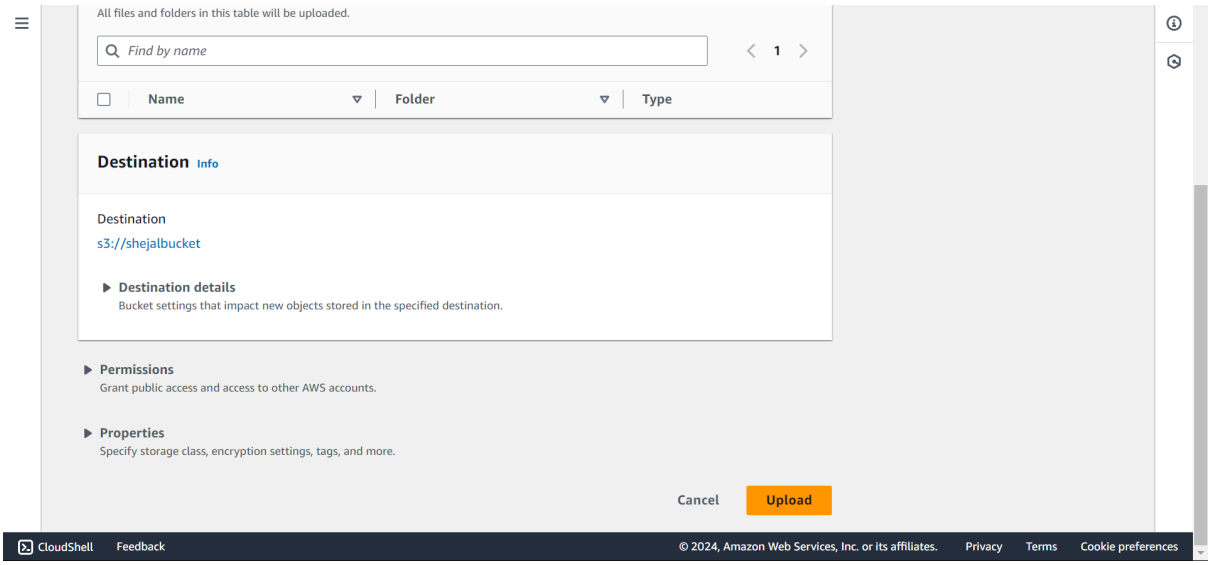
- **PHP Errors**: If there are errors in your PHP code, they will be displayed in the browser. You can also check the error logs in the XAMPP Control Panel under Apache > Logs > PHP Error Log.

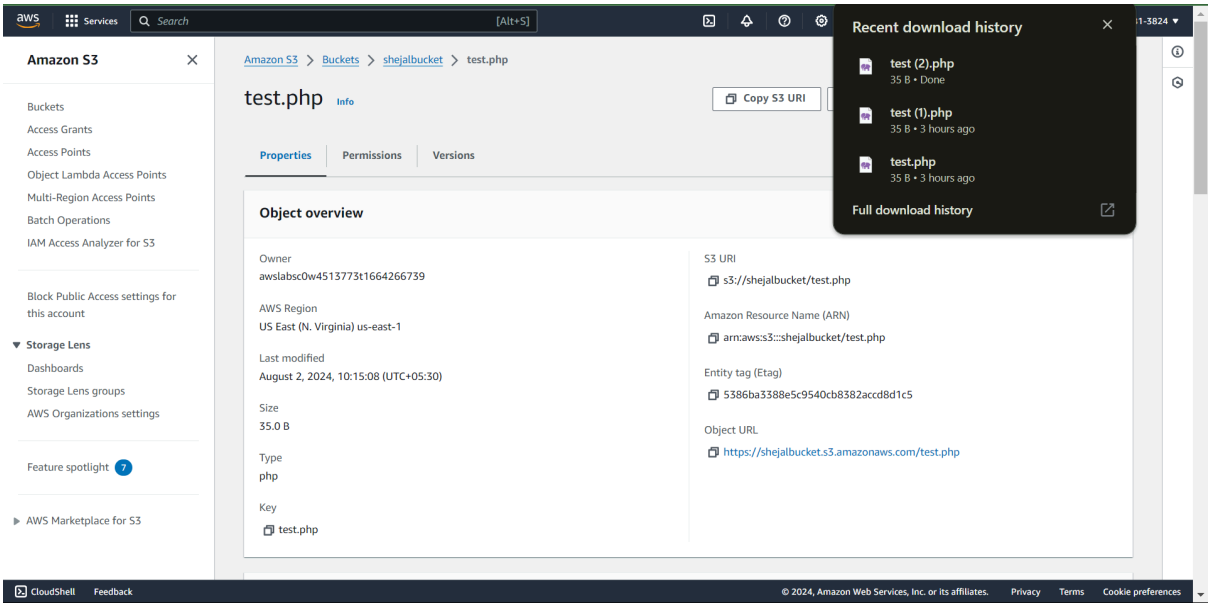
Your PHP file is now hosted on your local XAMPP server!

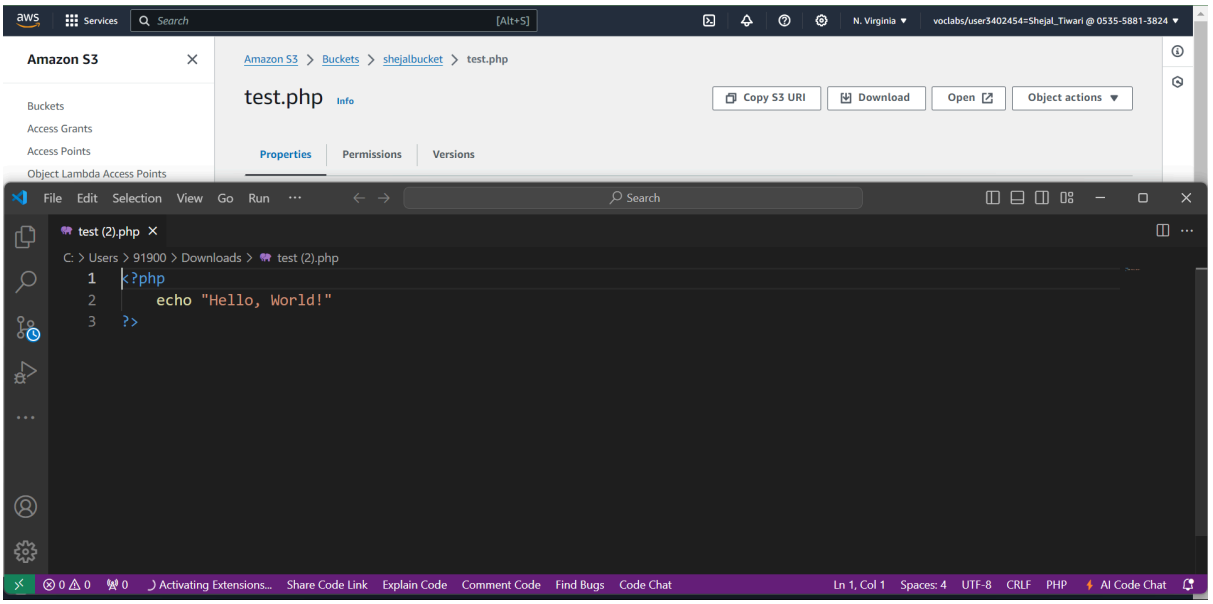
Step 7: Upload Objects (Optional)

1. Once the bucket is created, you can start uploading objects to it by clicking on the bucket name and using the "Upload" button.

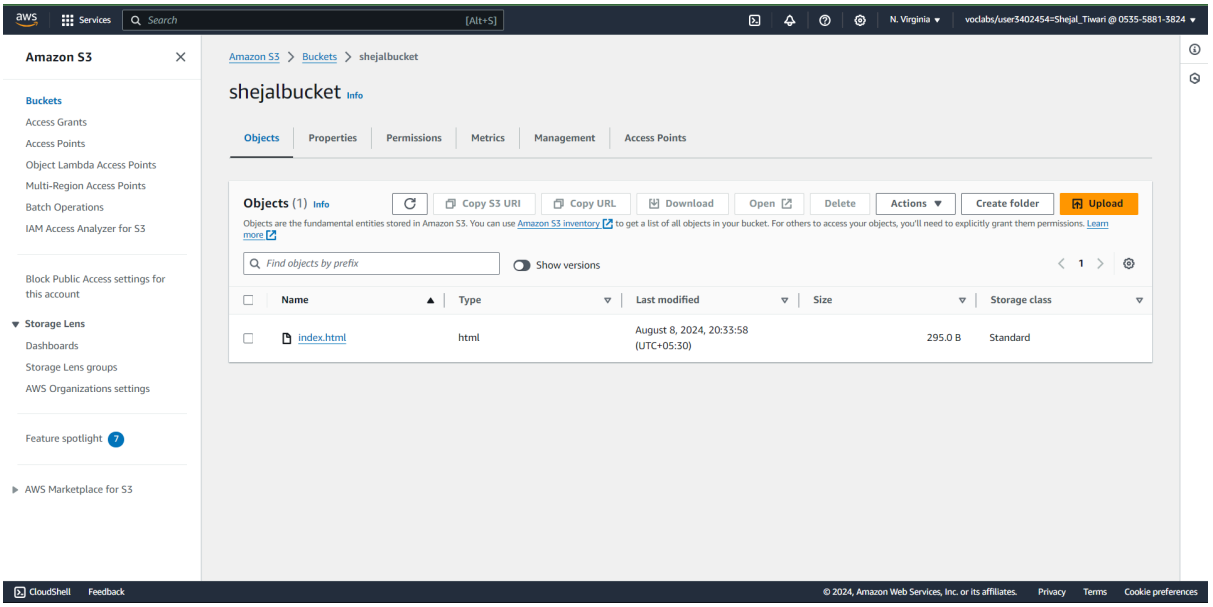




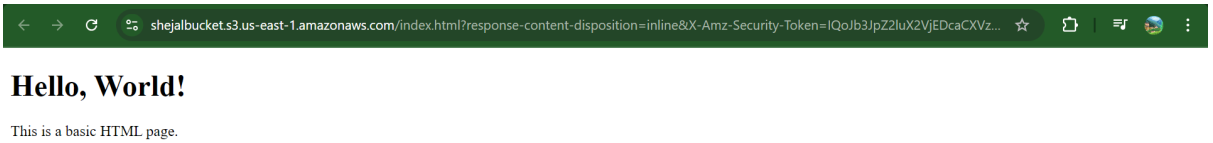

Recent download history
test (2).php
35 B • Done
test (1).php
35 B • 3 hours ago
test.php
35 B • 3 hours ago
Full download history
Amazon S3
Buckets
Access Grants
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings
Feature spotlight
AWS Marketplace for S3
Amazon S3 > Buckets > shejalbucket > test.php
test.php Info
Copy S3 URI
Properties
Permissions
Versions
Object overview
Owner
awslabsc0w4513773t1664266739
AWS Region
US East (N. Virginia) us-east-1
Last modified
August 2, 2024, 10:15:08 (UTC+05:30)
Size
35.0 B
Type
php
Key
test.php
S3 URI
s3://shejalbucket/test.php
Amazon Resource Name (ARN)
arn:aws:s3:::shejalbucket/test.php
Entity tag (Etag)
5386ba3388e5c9540cb8382accd8d1c5
Object URL
https://shejalbucket.s3.amazonaws.com/test.php
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Amazon S3
Buckets
Access Grants
Access Points
Object Lambda Access Points
Amazon S3 > Buckets > shejalbucket > test.php
test.php Info
Copy S3 URI
Download
Open
Object actions
Properties
Permissions
Versions
File Edit Selection View Go Run
test (2).php
C: > Users > 91900 > Downloads > test (2).php
1 <?php
2 echo "Hello, World!"
3 ?>
Activating Extensions...
Share Code Link
Explain Code
Comment Code
Find Bugs
Code Chat
Ln 1, Col 1
Spaces: 4
UTF-8
CRLF
PHP
AI Code Chat

In similar way it is done for index.html file



Click on the URL and your website is hosted on cloud .



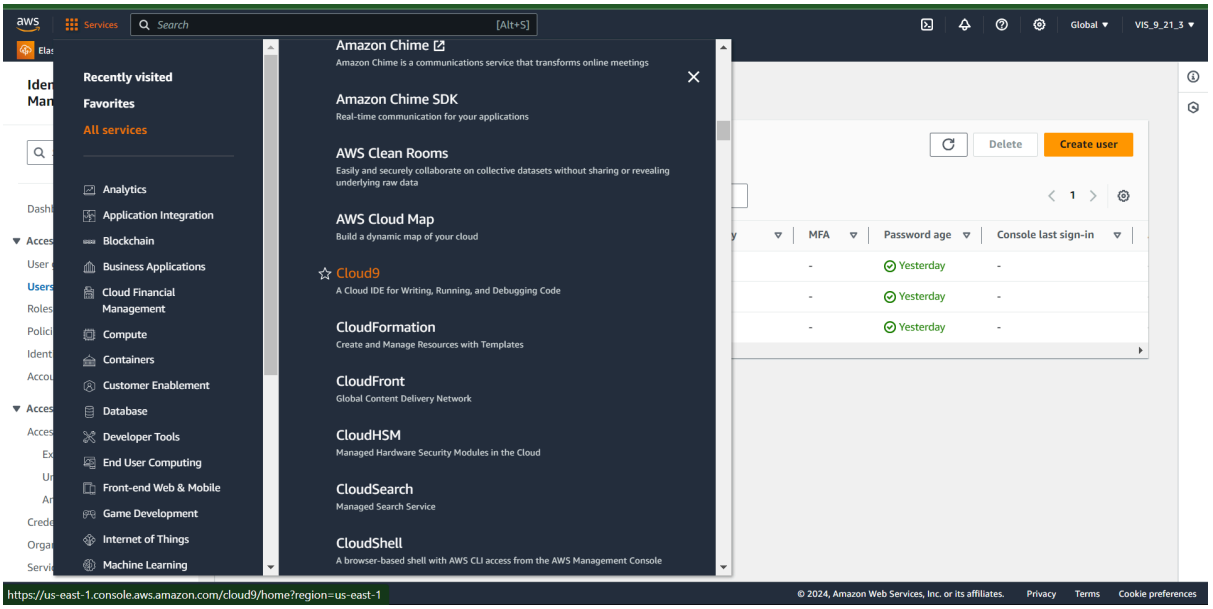
EXPERIMENT 1B

Step 1: Sign in to AWS Management Console

1. Go to the AWS Management Console.
2. Sign in with your AWS credentials.

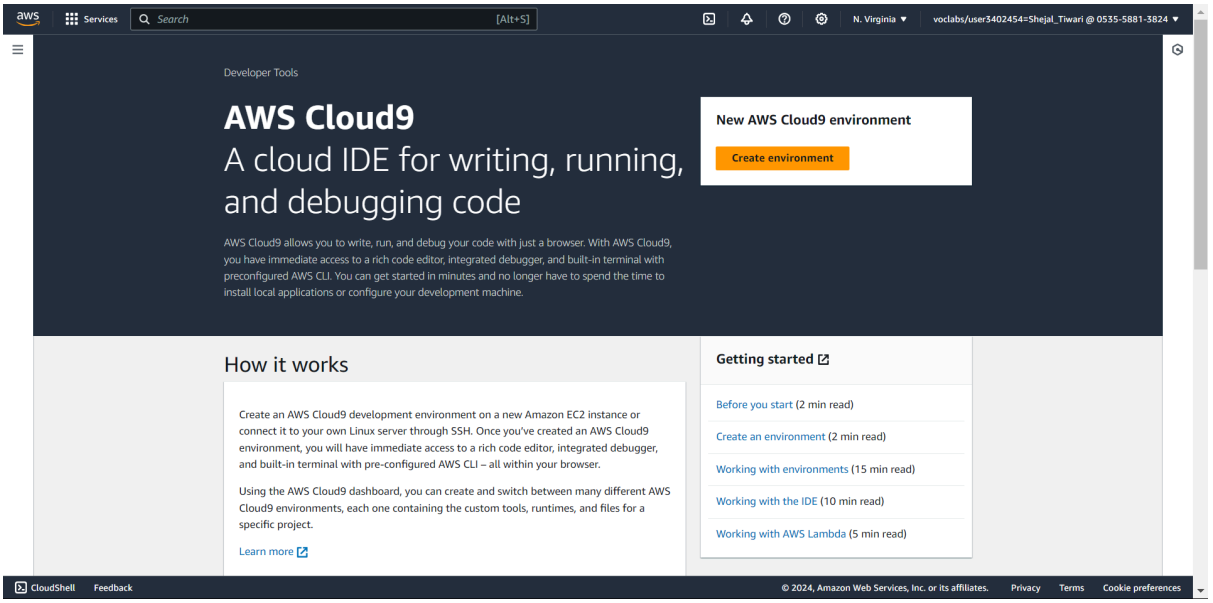
Step 2: Navigate to Cloud9

1. In the AWS Management Console, search for "Cloud9" in the search bar at the top.
2. Click on "Cloud9" under Services.



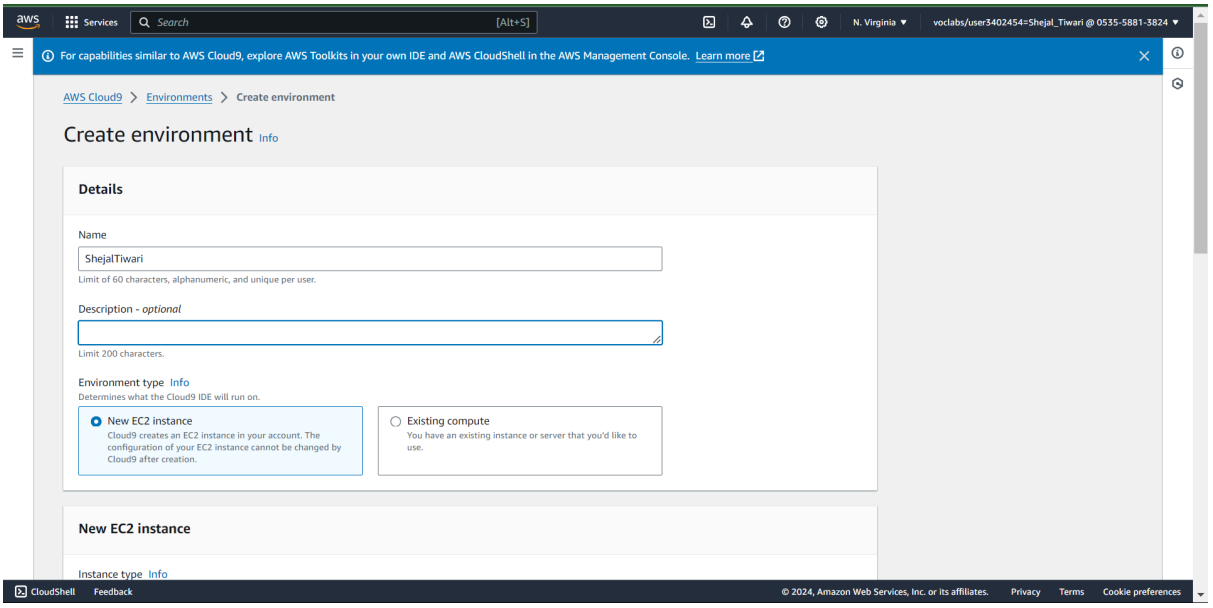
Step 3: Create a New Environment

1. On the Cloud9 dashboard, click on the "Create environment" button.

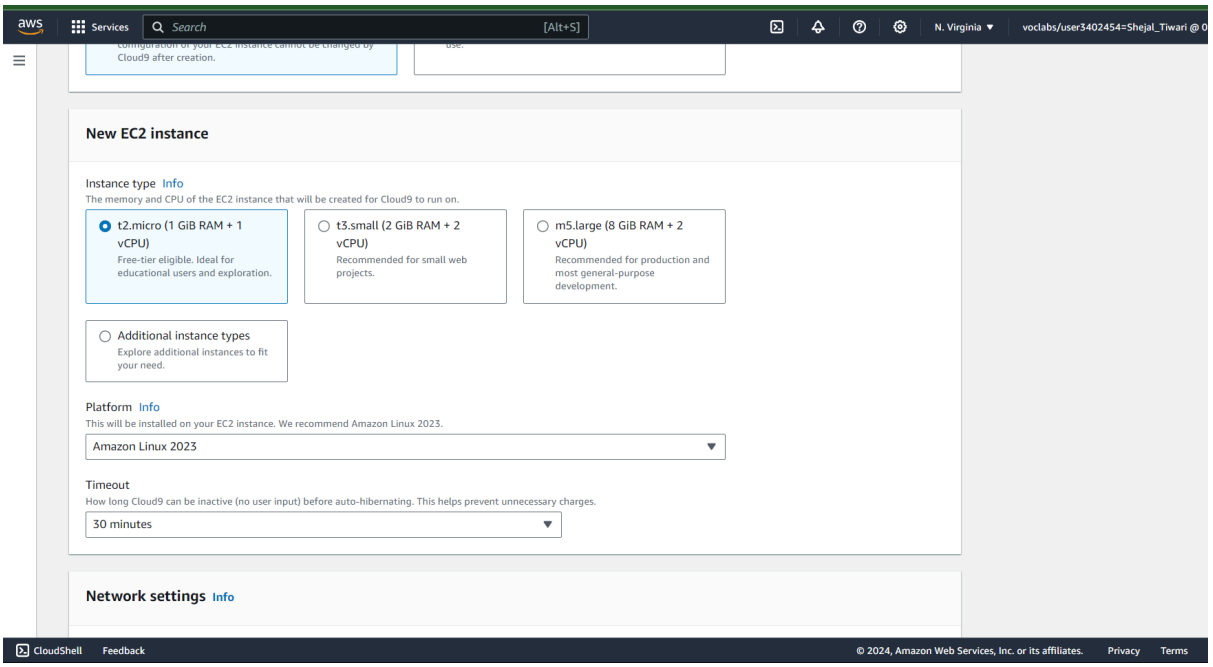


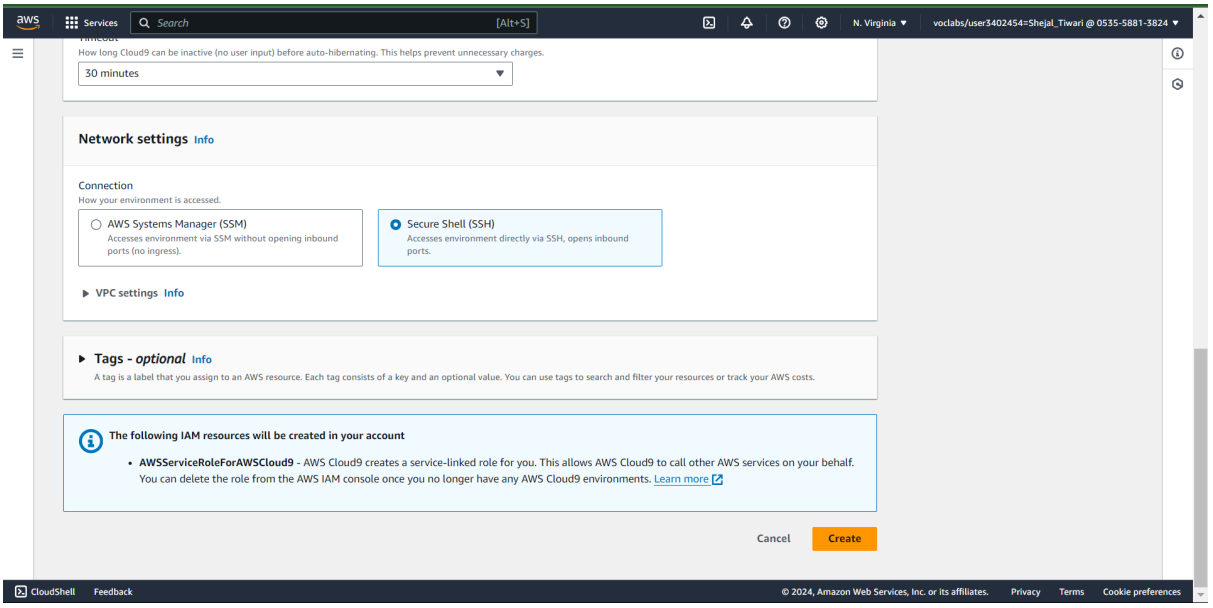
Step 4: Configure Environment Settings

1. **Name and Description**:
    - **Name**: Enter a name for your environment.
    - **Description**: Optionally, provide a description of the environment.
2. **Environment Settings**:
    - **Environment type**: Choose one of the following:
        - **Create a new EC2 instance for environment**: AWS will automatically create an EC2 instance for you. Choose this if you want to run the environment on a new EC2 instance.
        - **Connect to an existing EC2 instance**: If you have an existing EC2 instance, you can select this option to connect Cloud9 to it.
        - **Connect to your own server via SSH**: Use this if you want to connect Cloud9 to an on-premises server or another cloud provider's server.
    - **Instance type**: Select the instance type (e.g., t2.micro for free tier).
    - **Platform**: Choose the platform that suits your development needs (e.g., Amazon Linux, Ubuntu, etc.).
    - **Cost-saving setting**: Set an automatic timeout to stop the EC2 instance when it's not in use, saving costs.



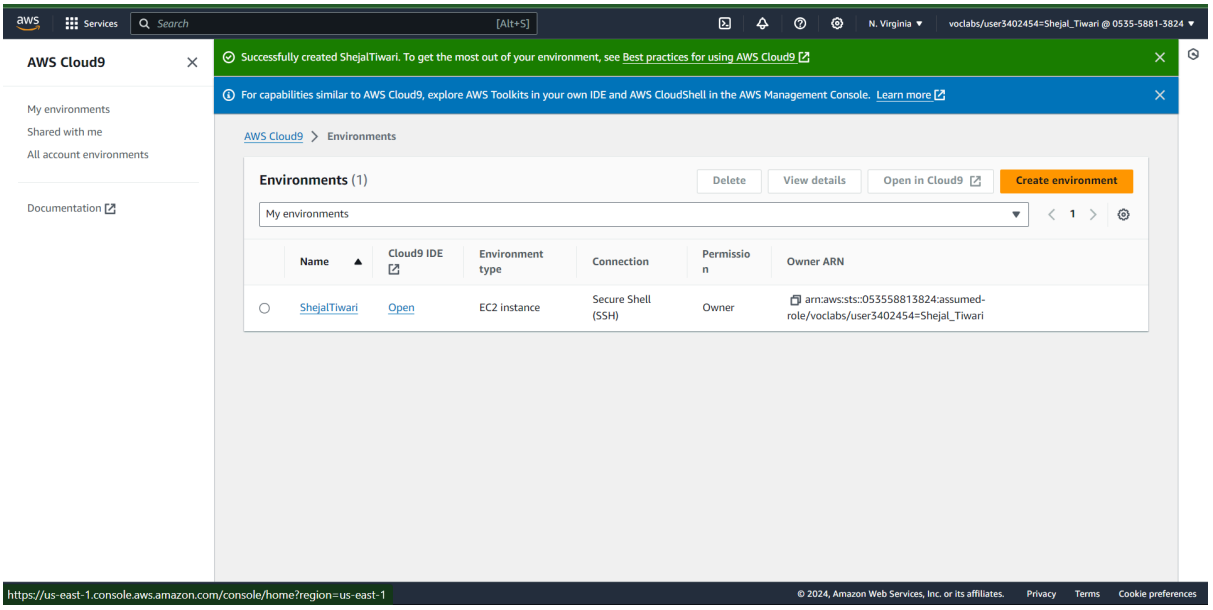
3. **Network Settings**:
    - If you selected "Create a new EC2 instance for environment," you'll need to choose a VPC (Virtual Private Cloud) and a subnet. Usually, the default options will work, but you can customize them if needed.





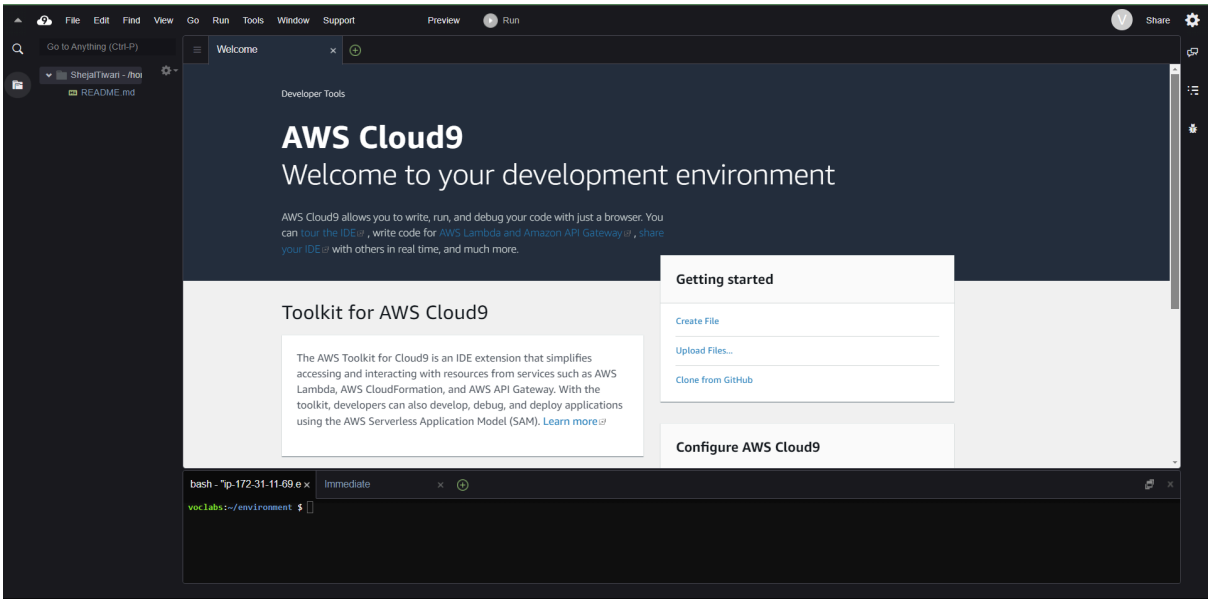
Step 5: Review and Create

1. Review all the settings you configured.
2. If everything looks good, click on the "Create environment" button.



Step 6: Access Your Cloud9 Environment

1. After a few moments, your Cloud9 environment will be ready, and you'll be automatically redirected to the Cloud9 IDE.
2. You can start coding immediately, using the terminal to install packages, run scripts, or manage files as you would on a local machine.



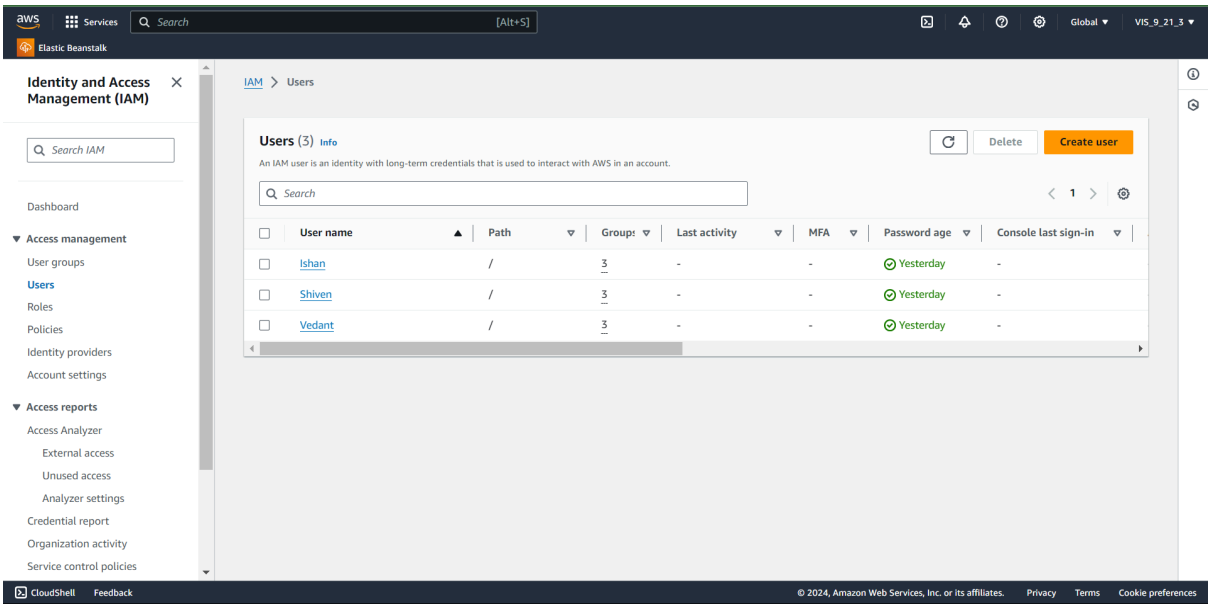
Additional Configuration (Optional)

- **Install additional tools**: You can install additional tools and software using the terminal within Cloud9, just as you would on a regular Linux server.
- **Clone repositories**: Use Git within Cloud9 to clone repositories, manage version control, and collaborate on code.

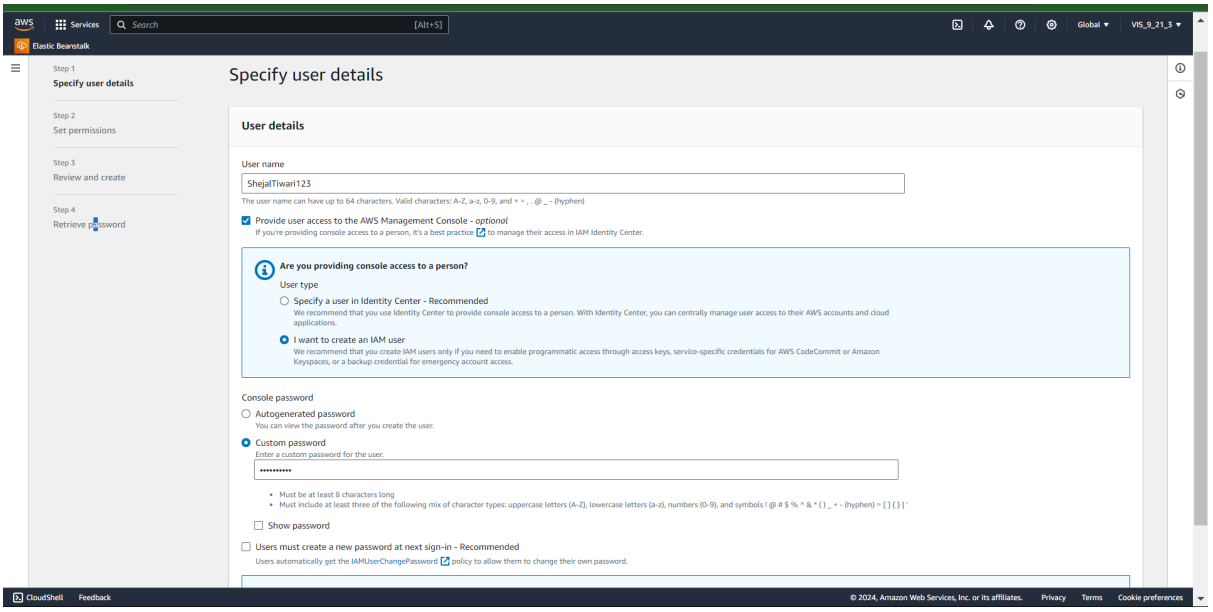
2. Creating an IAM User in AWS

To create a new IAM user in AWS:

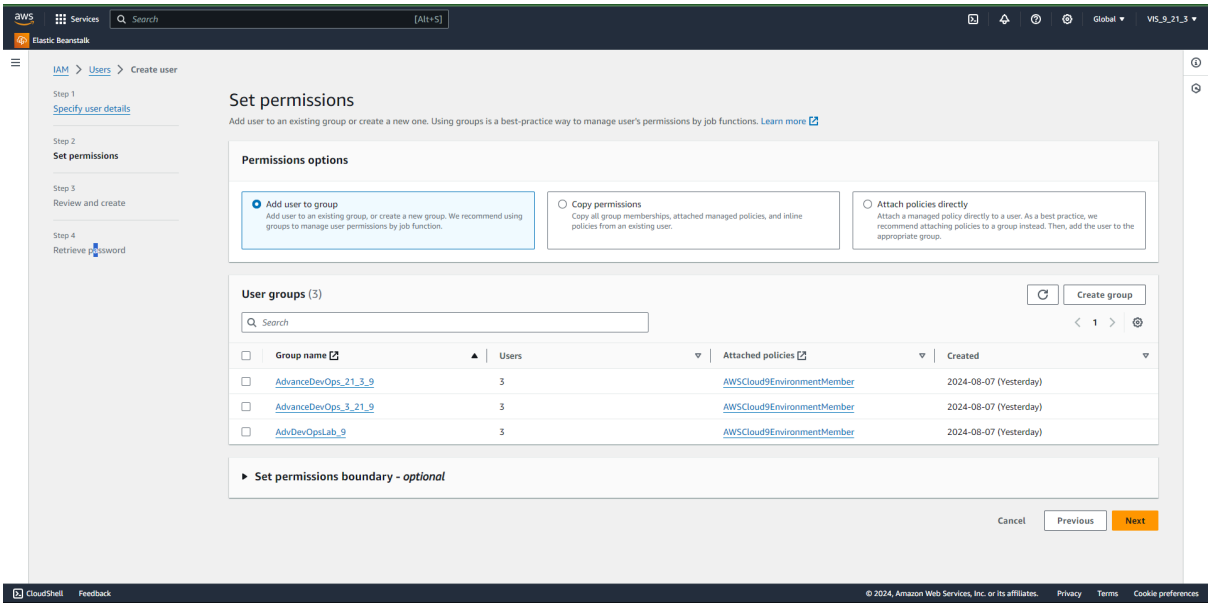
1. **Sign in to AWS Management Console**:
    - Go to the AWS Management Console.
2. **Navigate to IAM**:
    - Search for "IAM" in the services menu and select it.
3. **Create a New User**:
    - Click on "Users" in the left-hand menu.
    - Click on "Add user."



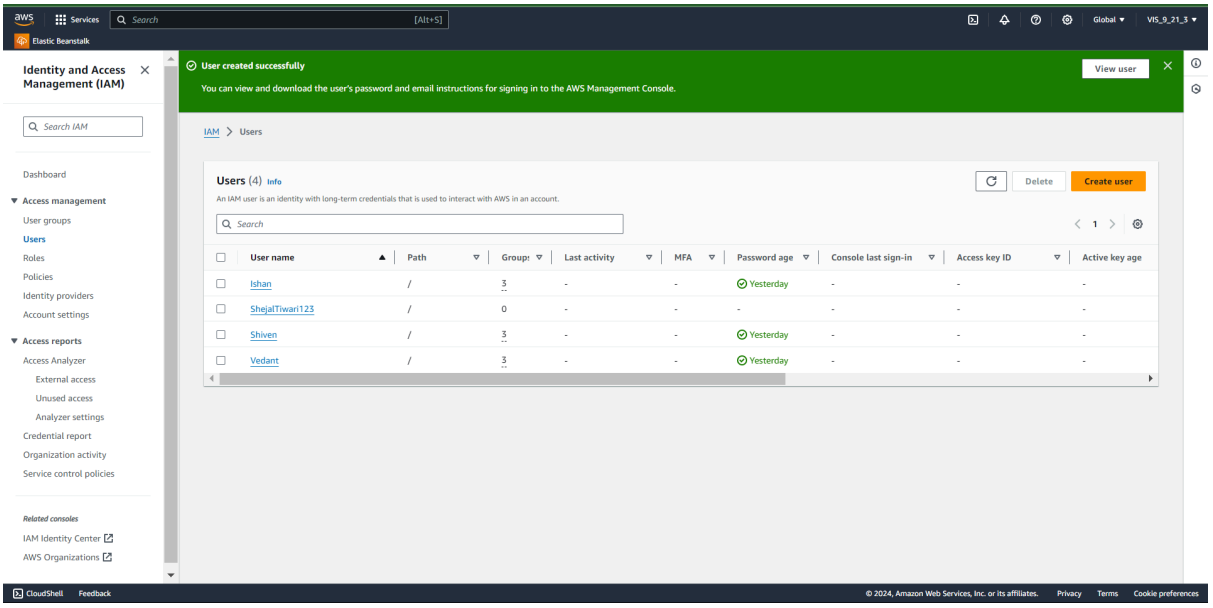
4. **Set User Details**:
    - **User Name**: Enter a name for the new user.
    - **Access Type**:
        - **Programmatic access**: If the user needs access to AWS CLI, SDK, or API.
        - **AWS Management Console access**: If the user needs access to the AWS Management Console.



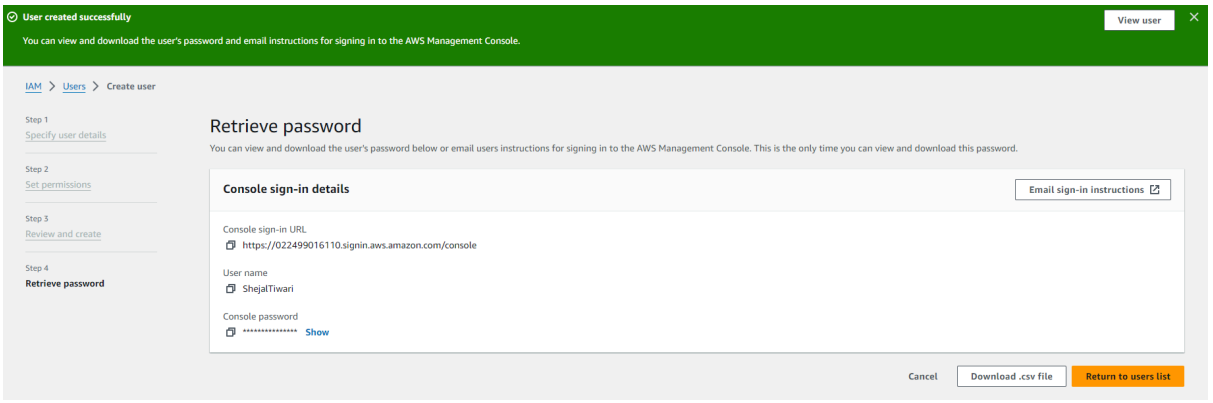
5. **Set Permissions**:
    - You can attach existing policies directly, add the user to a group with predefined permissions, or copy permissions from another user.



6. **Tags (Optional)**:
    - Add tags to the user account for easy identification or organization.
7. **Review and Create**:
    - Review the settings and click "Create user."



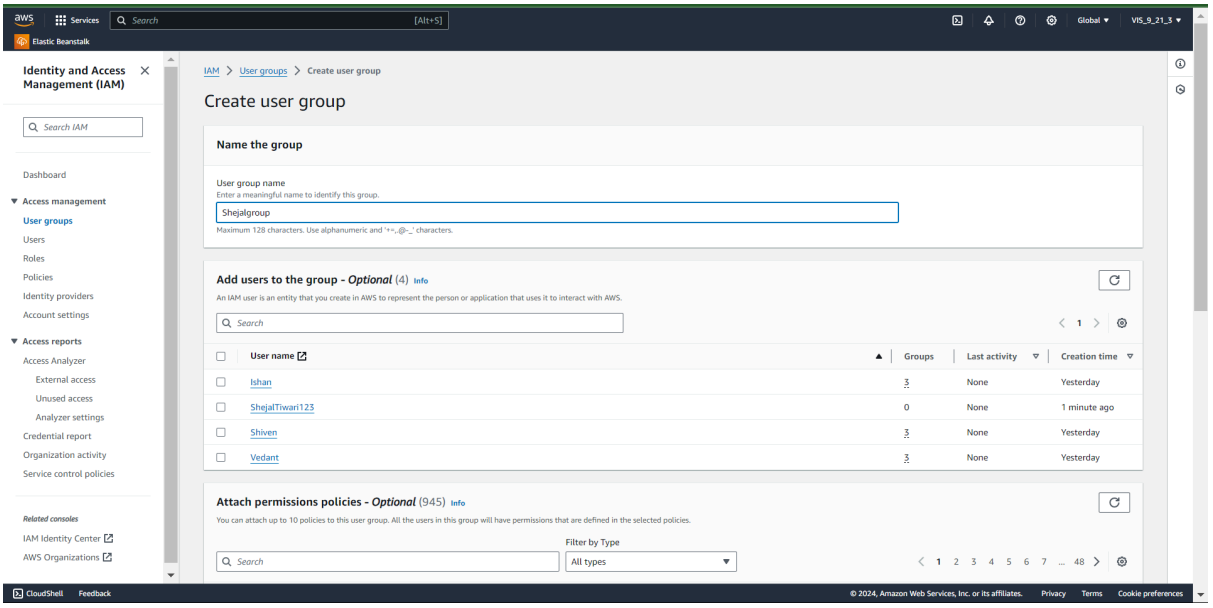
8. **Download Credentials**:
    - On the confirmation page, download the access key and secret key if you've enabled programmatic access. Keep them secure.



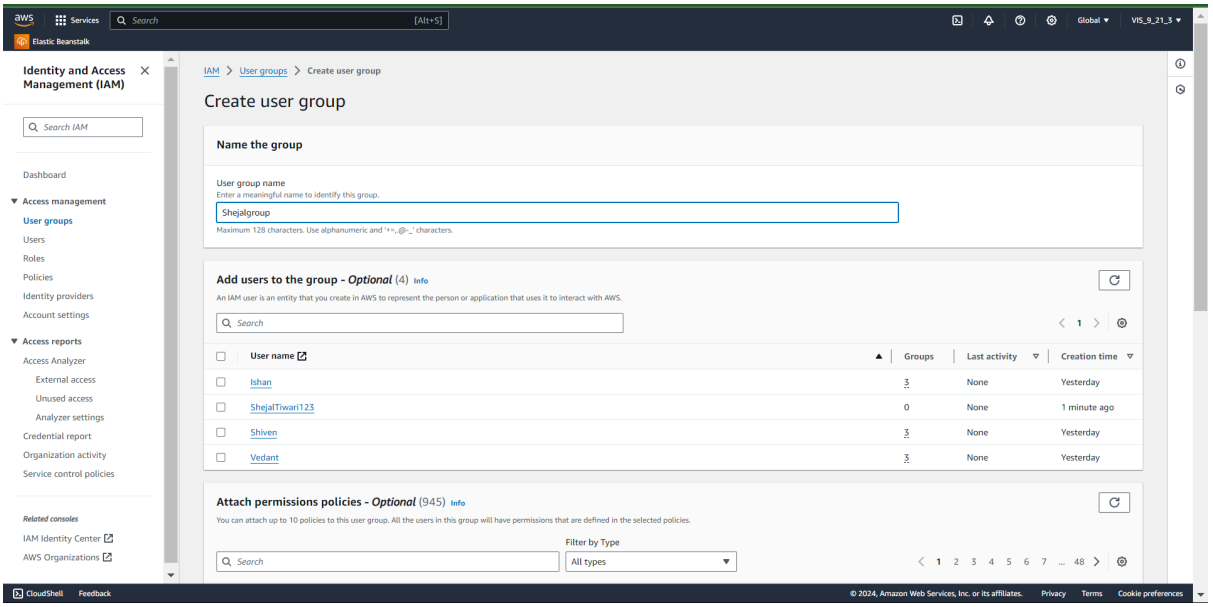
Creating a User Group in AWS IAM

To create a new user group in AWS IAM:

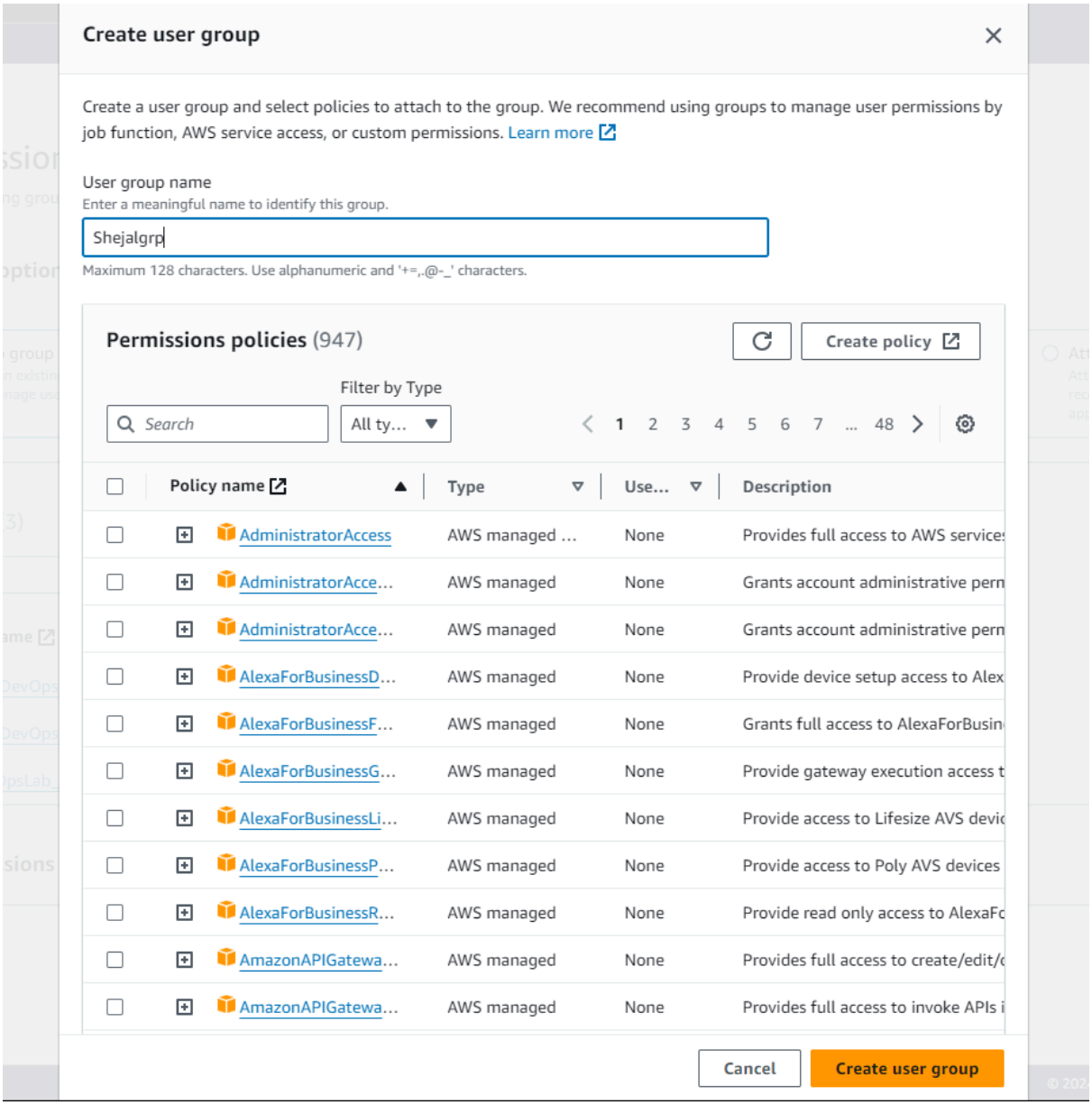
1. **Sign in to AWS Management Console**:
    - Go to the AWS Management Console.
2. **Navigate to IAM**:
    - Search for "IAM" in the services menu and select it.
3. **Create a New Group**:
    - Click on "User groups" in the left-hand menu.
    - Click on "Create group."

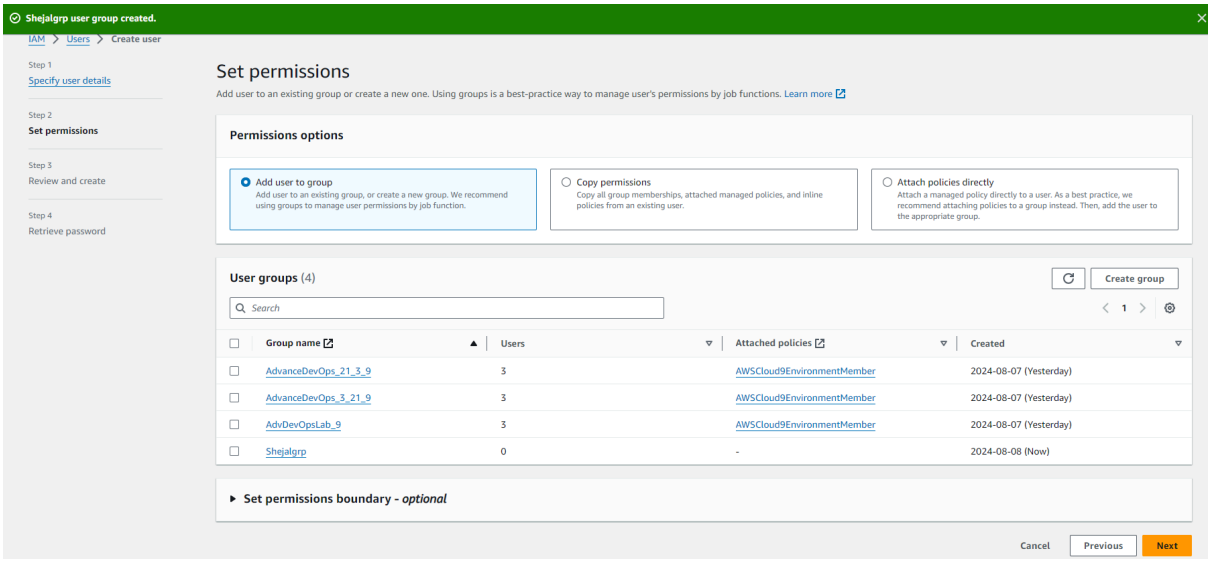


4. **Set Group Name**:
    - Enter a name for your group in the "Group name" field.

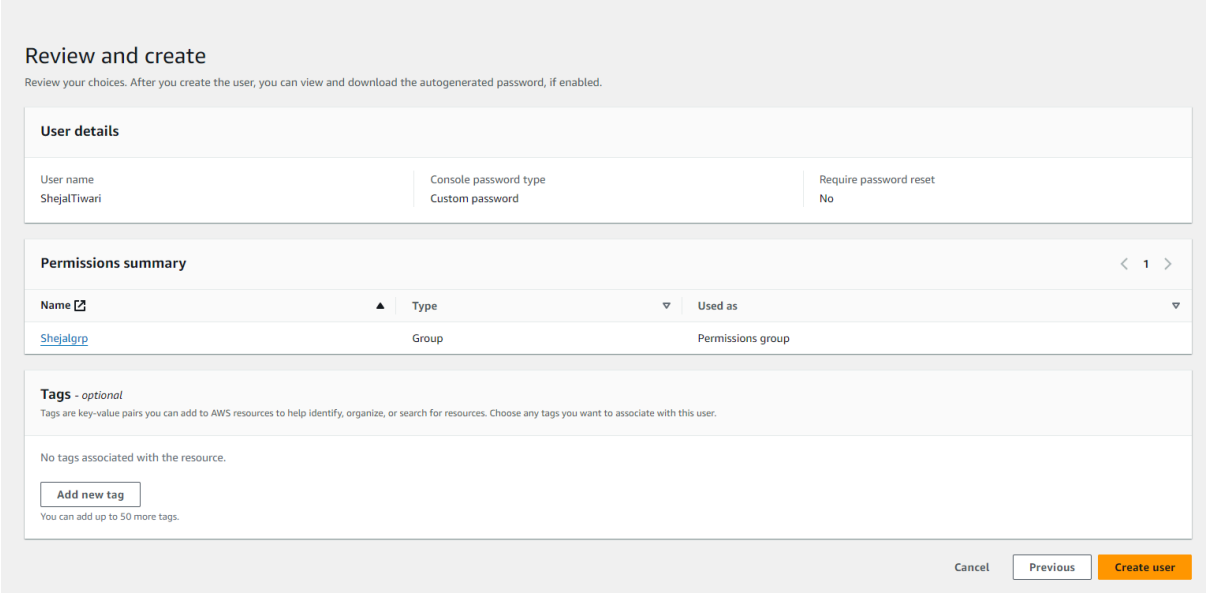


5. **Attach Permissions Policies**:
    - Select the policies you want to attach to the group. These policies define what actions the users in this group can perform.
    - You can attach existing AWS managed policies, or create and attach a custom policy.


Create user group
Create a user group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. Learn more
User group name
Enter a meaningful name to identify this group.
Shejalgrp
Maximum 128 characters. Use alphanumeric and '+=,.@-_' characters.
Permissions policies (947)
Create policy
Filter by Type
Search
All ty...
1 2 3 4 5 6 7 ... 48
Policy name
Type
Use...
Description
AdministratorAccess AWS managed ... None Provides full access to AWS service
AdministratorAcce... AWS managed None Grants account administrative perm
AdministratorAcce... AWS managed None Grants account administrative perm
AlexaForBusinessD... AWS managed None Provide device setup access to Alex
AlexaForBusinessF... AWS managed None Grants full access to AlexaForBusin
AlexaForBusinessG... AWS managed None Provide gateway execution access t
AlexaForBusinessLi... AWS managed None Provide access to Lifesize AVS devic
AlexaForBusinessP... AWS managed None Provide access to Poly AVS devices
AlexaForBusinessR... AWS managed None Provide read only access to AlexaFc
AmazonAPIGatewa... AWS managed None Provides full access to create/edit/
AmazonAPIGatewa... AWS managed None Provides full access to invoke APIs i
Cancel
Create user group


Shejalgrp user group created.
IAM > Users > Create user
Step 1
Specify user details
Step 2
Set permissions
Step 3
Review and create
Step 4
Retrieve password
Set permissions
Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more
Permissions options
Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.
Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.
User groups (4)
Create group
Search
Group name
Users
Attached policies
Created
AdvanceDevOps_21_3_9 3 AWSCloud9EnvironmentMember 2024-08-07 (Yesterday)
AdvanceDevOps_3_21_9 3 AWSCloud9EnvironmentMember 2024-08-07 (Yesterday)
AdvDevOpsLab_9 3 AWSCloud9EnvironmentMember 2024-08-07 (Yesterday)
Shejalgrp 0 - 2024-08-08 (Now)
Set permissions boundary - optional
Cancel
Previous
Next

6. **Review and Create**:
    - Review your settings and click on "Create group."



7. **Add Users to the Group** (Optional):
    - After creating the group, you can add users to it by selecting the group, then going to the "Users" tab and clicking on "Add users to group."