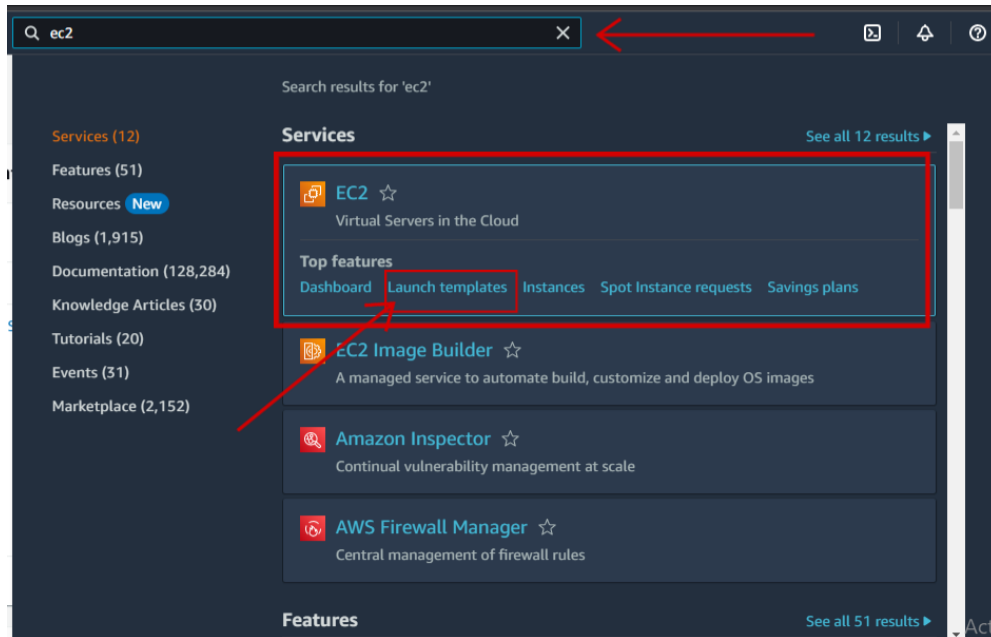


## Assignment 7

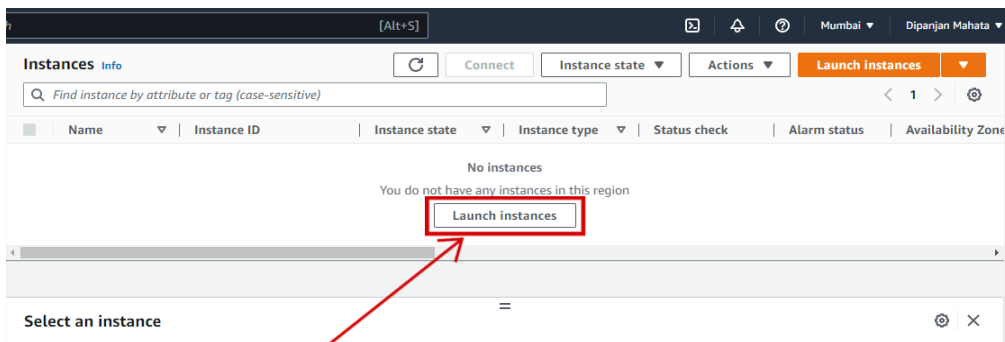
### Upload a static website in EC2 server.

#### Steps for creating an AWS account:

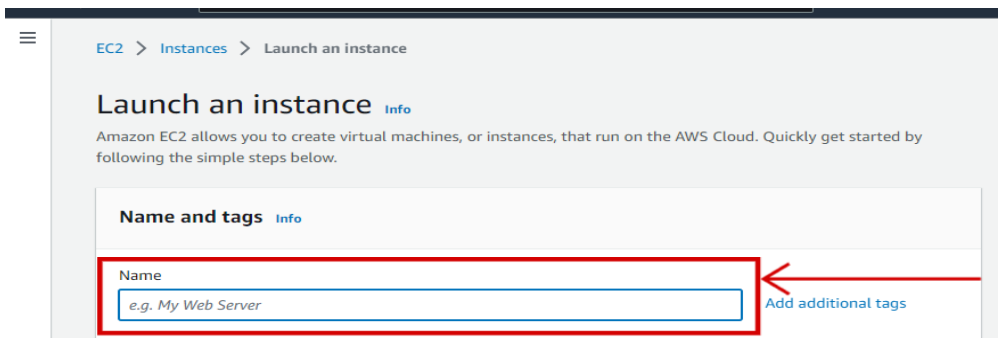
1. **Sign in.** Sign in as a root user. Provide username and password when prompted.
2. Search on the search bar **EC2**. After that click on **Launch template** in the **EC2**.



3. Now click on **Launch instances** and create a public bucket



4. Enter the your server name.



## 5. Click on **Ubuntu**.

**Quick Start**

Amazon Linux macOS **Ubuntu** Windows Red Hat S

aws Mac ubuntu® Microsoft Red Hat

**Amazon Machine Image (AMI)**

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type  
ami-0f8ca728008ff5af4 (64-bit (x86)) / ami-08795883c7b47140 (64-bit (Arm))  
Virtualization: hvm ENA enabled: true Root device type: eb  
Free tier eligible

**Description**

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2023-02-08

**Architecture** 64-bit (x86) **AMI ID** ami-0f8ca728008ff5af4 **Verified provider**

## 6. Click on **Create new key pair**.

**Instance type**

t2.micro Family: t2 1 vCPU 1 GiB Memory  
On-Demand Linux pricing: 0.0124 USD per Hour  
On-Demand Windows pricing: 0.017 USD per Hour  
On-Demand RHEL pricing: 0.0724 USD per Hour  
On-Demand SUSE pricing: 0.0124 USD per Hour  
Free tier eligible Compare instance types

**Key pair (login)** Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required Select Create new key pair

**Network settings** Info Edit

Network Info  
vpc-091fcb910c01454b7

**Summary**

Number of instances Info  
1

Software Image (AMI)  
Canonical, Ubuntu, 22.04 LTS, ...read more  
ami-0f8ca728008ff5af4

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 X

Cancel Launch instance  
Activate Windows  
Go to Settings to activate Windows

## 7. Enter the **Key Per Name**.

Key pairs allow you to connect to your instance securely.

Enter the name of the key pair below. When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** [Learn more](#)

Key pair name  
key12  
The name can include upto 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

☒ RSA  
RSA encrypted private and public key pair

☐ ED25519  
ED25519 encrypted private and public key pair (Not supported for Windows instances)

Private key file format

☒ .pem  
For use with OpenSSH

☐ .ppk  
For use with PuTTY

Cancel Create key pair

8. Click all the **check box**.

**Firewall (security groups)** [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group ☐ Select existing security group

We'll create a new security group called 'launch-wizard-1' with the following rules:

- ☒ Allow SSH traffic from Anywhere  
0.0.0.0/0  
Helps you connect to your instance
- ☒ Allow HTTPS traffic from the internet  
To set up an endpoint, for example when creating a web server
- ☒ Allow HTTP traffic from the internet  
To set up an endpoint, for example when creating a web server

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

9. Click on **Launch Instances**.

**Configure storage** [Info](#) [Advanced](#)

1x  GiB  Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GiB of EBS General Purpose (SSD) or Magnetic storage

[Add new volume](#)

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

0 x File systems [Edit](#)

[Advanced details](#) [Info](#)

Number of instances [Info](#)

**Software Image (AMI)**  
Canonical, Ubuntu, 22.04 LTS, ...[read more](#)  
ami-0f8ca728008ff5af4

**Virtual server type (instance type)**  
t2.micro

**Firewall (security group)**  
New security group

**Storage (volumes)**  
1 volume(s) - 8 GiB

[Cancel](#) [Launch instance](#)

10. Click on **View all instances**.

**Next Steps**

**Create billing and free tier usage alerts**

To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.

[Create billing alerts](#)

**Connect to your instance**

Once your instance is running, log into it from your local computer.

[Connect to instance](#)

[Learn more](#)

**Connect an RDS database**

Configure the connection between an EC2 instance and a database to allow traffic flow between them.

[Connect an RDS database](#)

[Create a new RDS database](#)

[Learn more](#)

[View all instances](#)

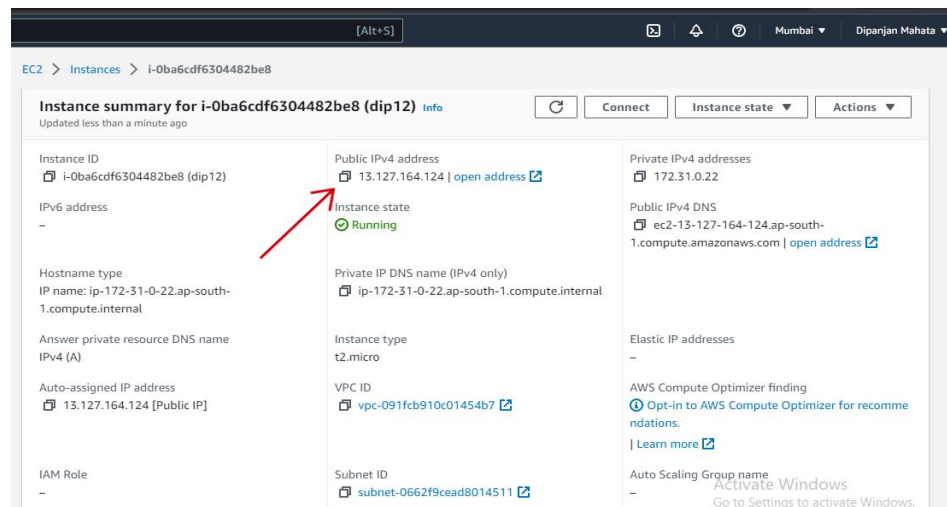
11. Now click on the **Instance ID** number.

ch [Alt+S] Mumbai Dipanjan Mahata

**Instances (1)** [Info](#) [Refresh](#) [Connect](#) [Instance state](#) [Actions](#) [Launch Instances](#)

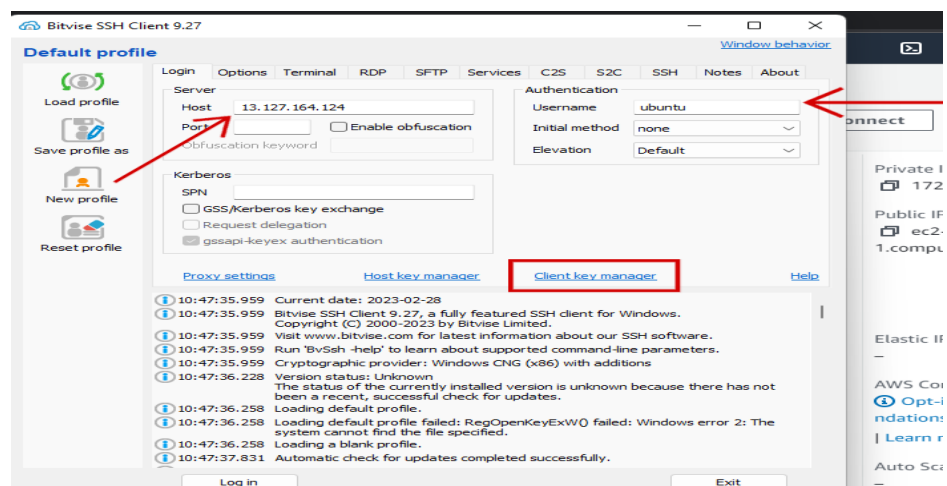
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	dip12	<a href="#">i-0ba6cdf6304482be8</a>	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1b

12. Now click on the **Public IPv4 address** to copy the IP.

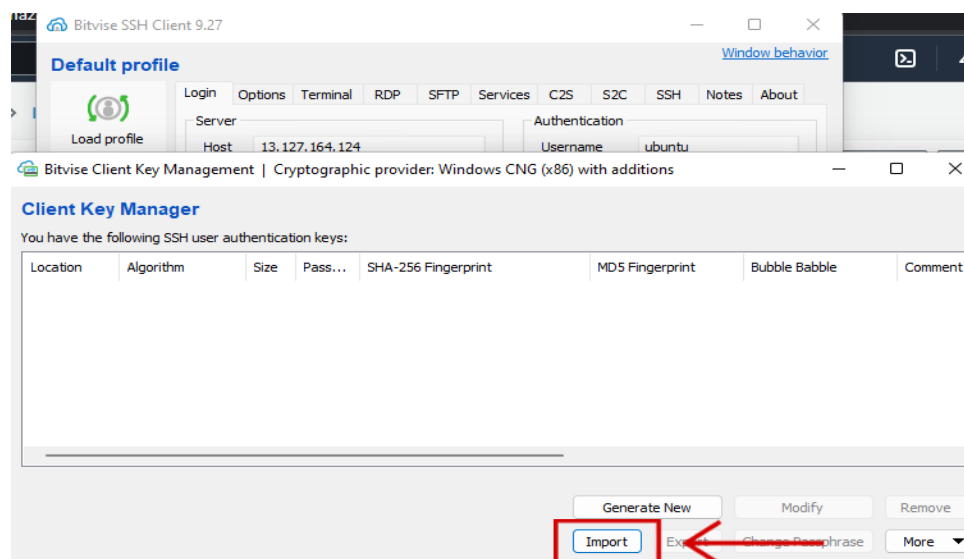


13. Now install the **Bitvise SSH client** on your pc.

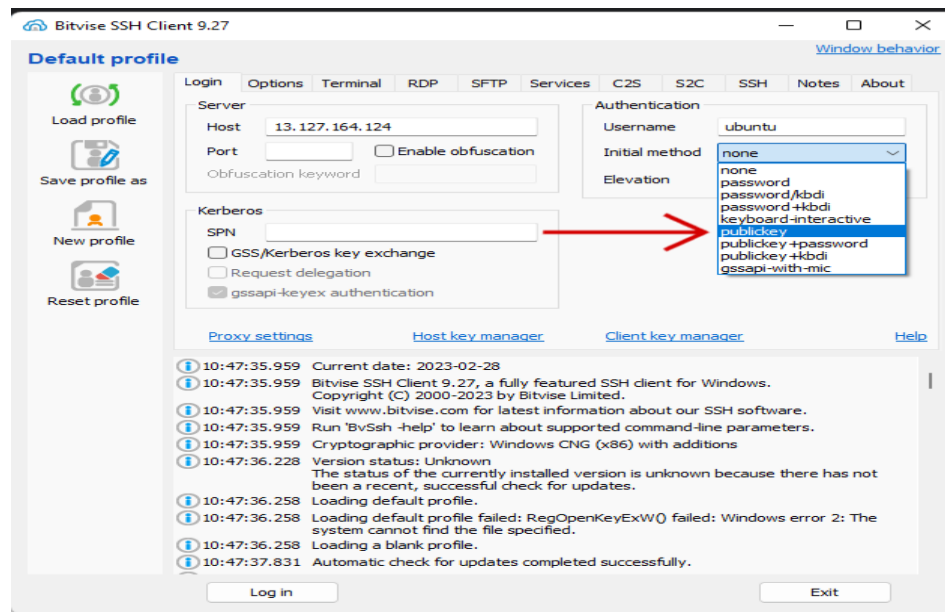
14. **Paste the IP in the Host section** and enter a **user name**. After that click on **client key manager**.



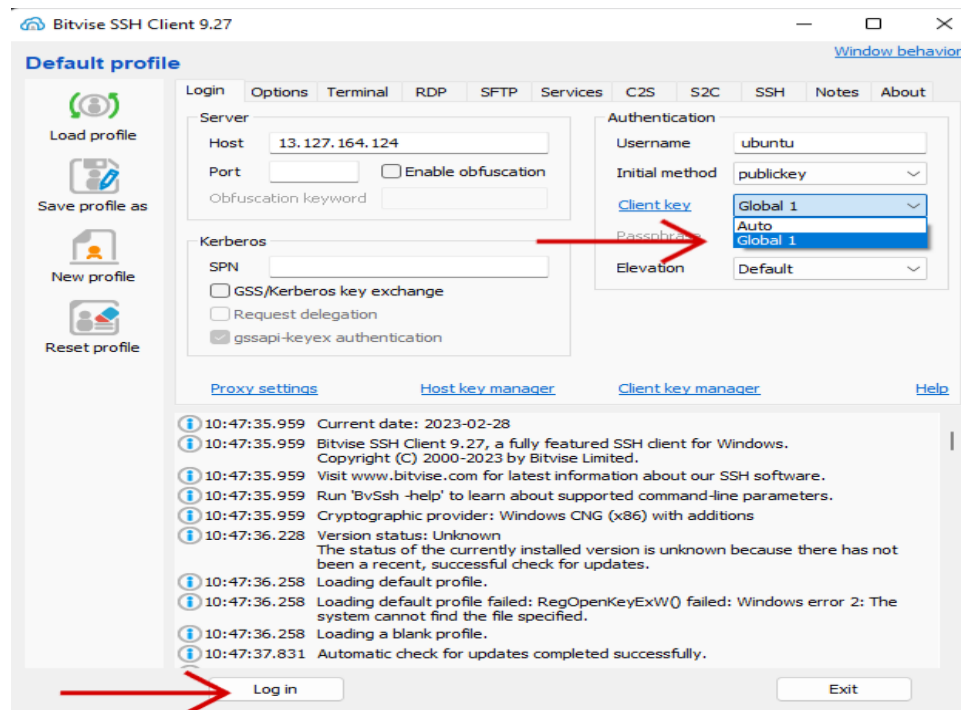
15. Now click on **import** to import the **key pair** that you have created the **step 7**.



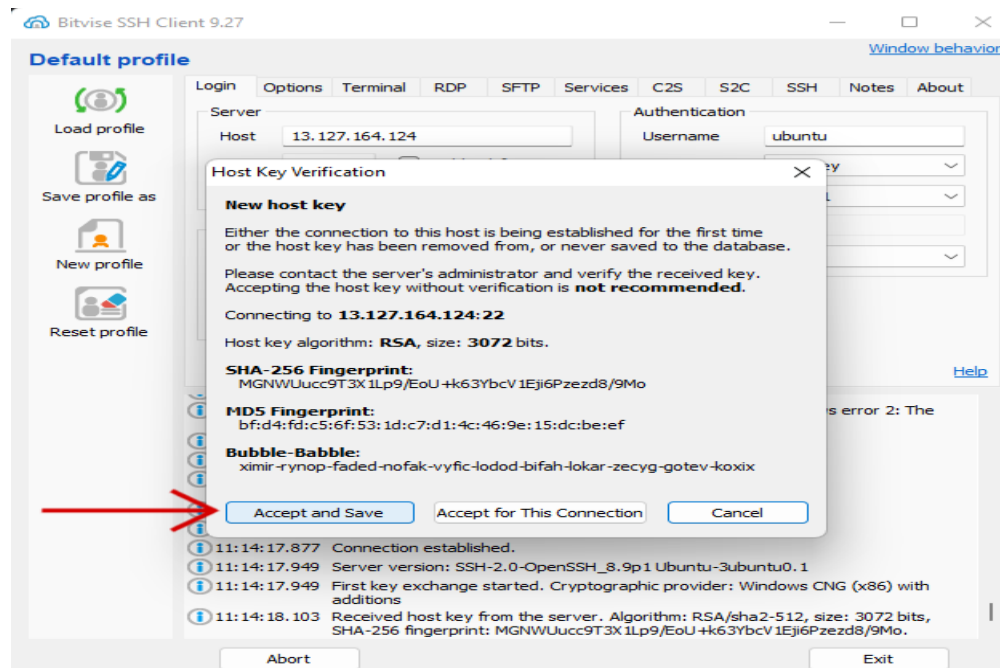
16. Set the **Initial method** as **publickey**.



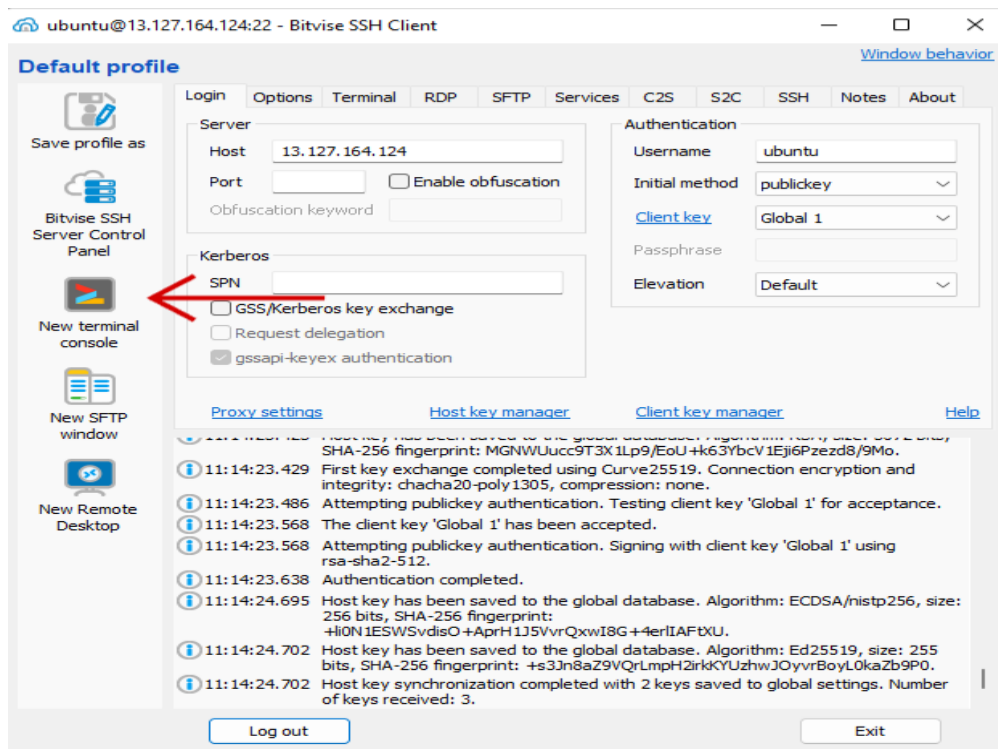
17. Set the **client key** as **Global 1**. After that click **Log in**.



18. Click on **Accept and Save**.

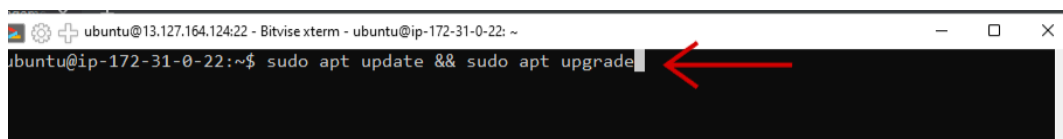


19. Click on **New terminal consol**.



20. Enter these command on the consol.

1) Enter **sudo apt update && sudo apt upgrade** to update the Ubuntu version.



II) Enter **sudo apt install nginx** to install the nginx.

```
ubuntu@ip-172-31-0-22:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-0-22:~$ sudo apt install nginx
```

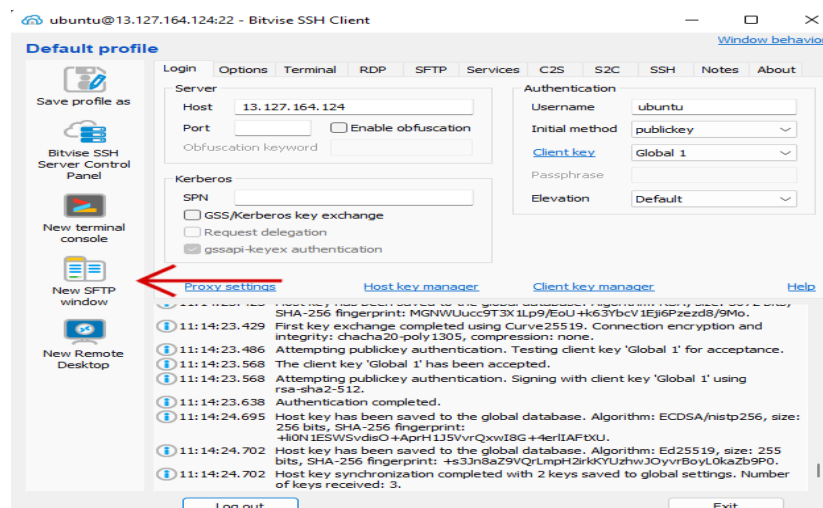
III) Enter the following command to change the directory from **/home /ubuntu** to **/var/www**.

```
ubuntu@ip-172-31-0-22:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-0-22:~$ cd ..
ubuntu@ip-172-31-0-22:/home$ cd ..
ubuntu@ip-172-31-0-22:/$ pwd
/
ubuntu@ip-172-31-0-22:/$ cd var
ubuntu@ip-172-31-0-22:/var$ cd www
```

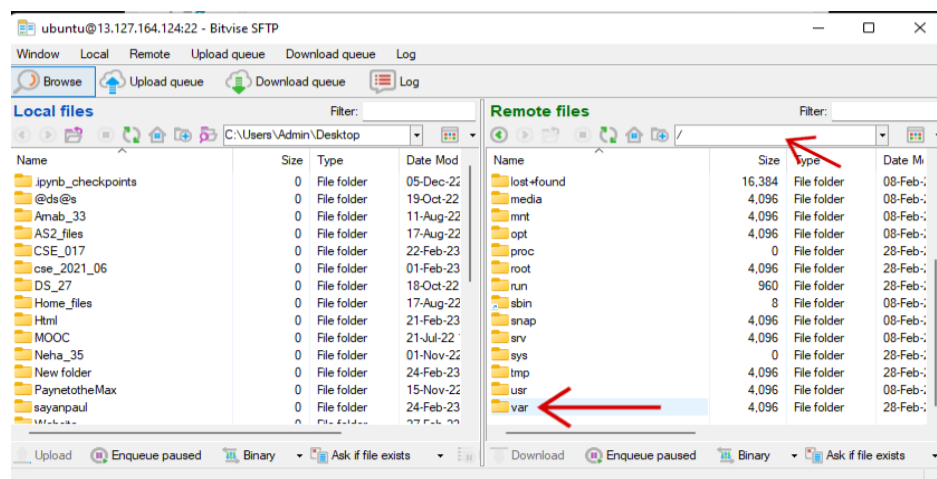
IV) Enter **sudo chmod 777 html** to give all the permission.

```
sudo: chmod: command not found
ubuntu@ip-172-31-0-22:/var/www$ sudo chmod 777 html
```

21. Click on **New SFTP window**.

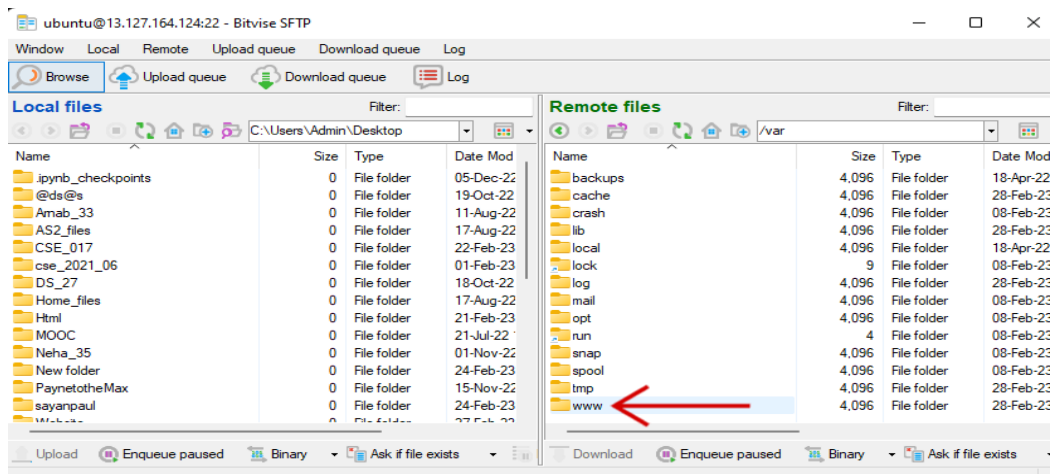


22. First enter **/** in the right bar to get in the root. After that click on **var**.

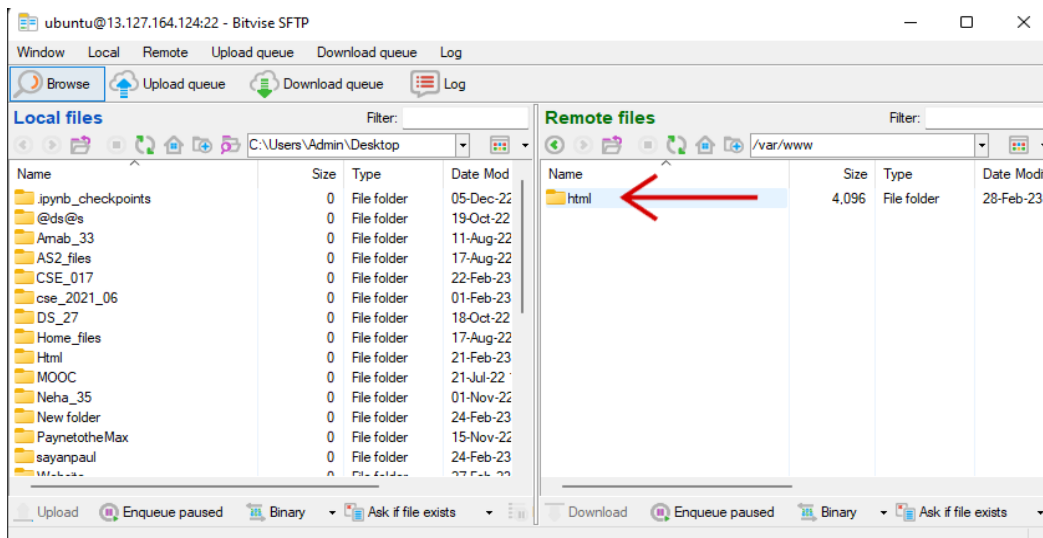




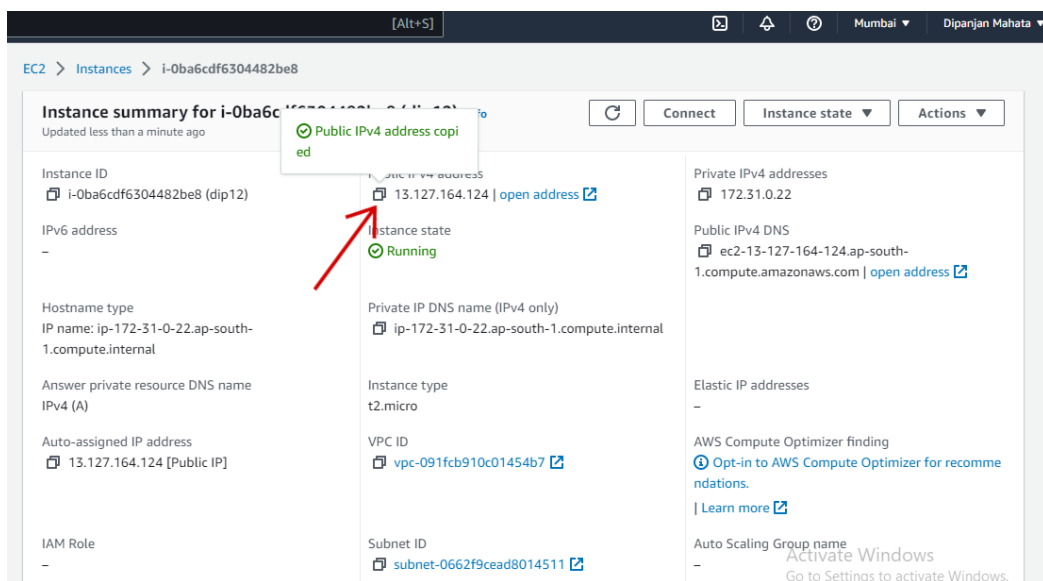
23. Now click on **www**.



24. Now click on **html**. After that copy the **static website file form local file** (which in on the local pc) and **paste it in the html file in the remote file section**.



25. Now come back to AWS and copy the same IP.





26. Now paste the IP in a new tab. Now your static website is live on the EC2 server.

