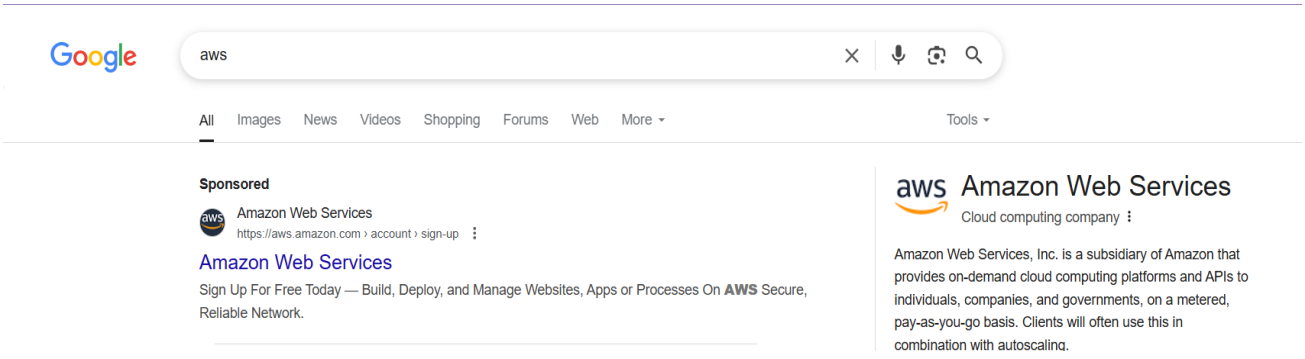
Amazon Web Services (AWS) Account Creation

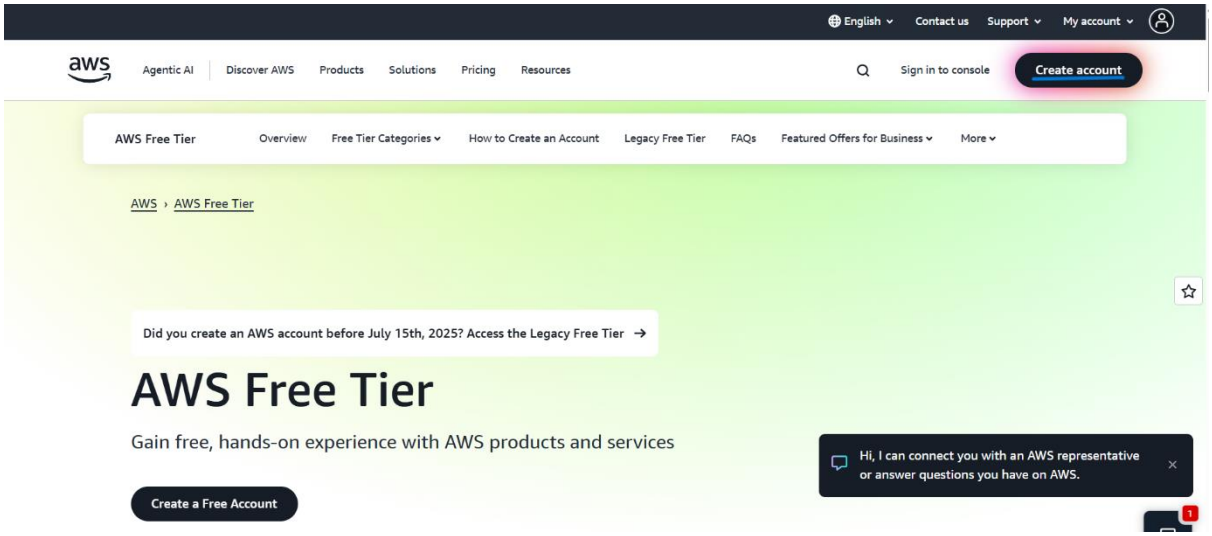
1.Open Google Chrome (or any browser).



- Visit the AWS official website: https://aws.amazon.com/ or search for AWS.
- You will reach the **AWS homepage**.

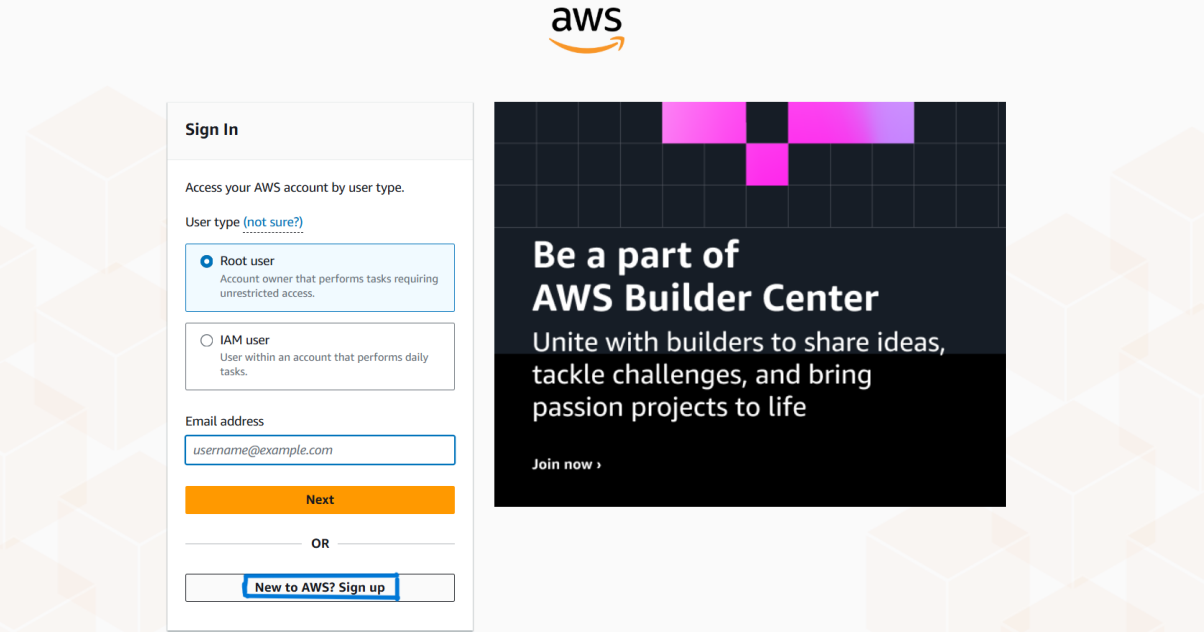
2. Click Create Account

- On the AWS home page, click **Create an AWS Account**.



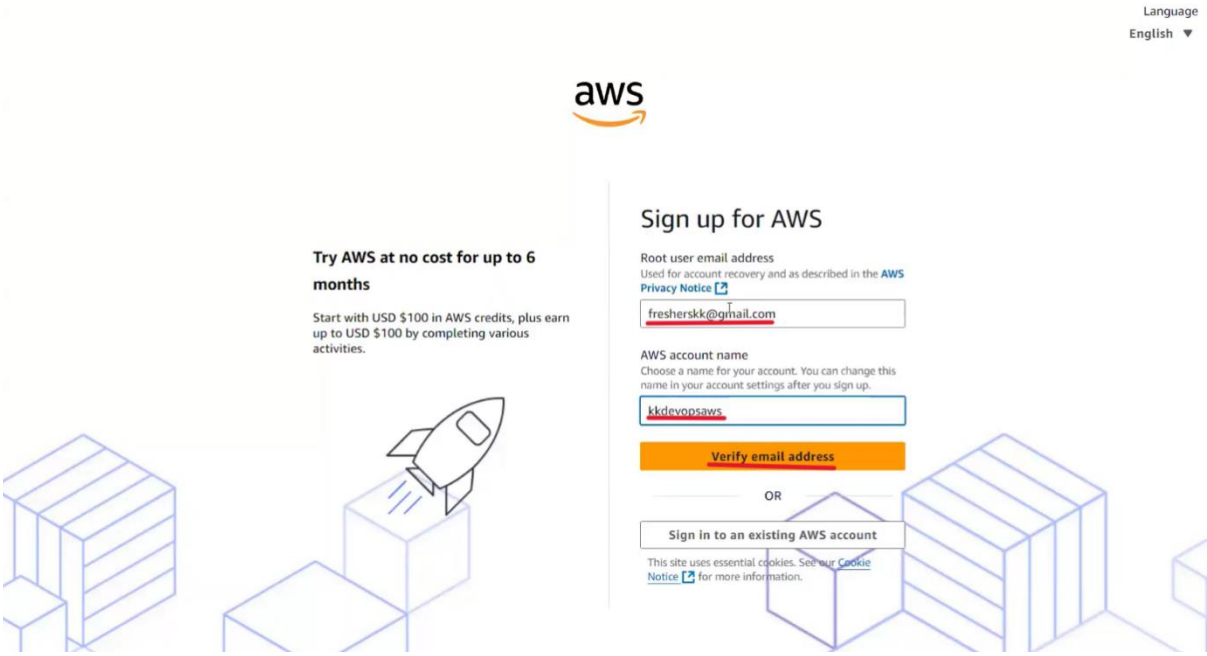
3. Create a New AWS Account

- On the homepage, click **“Create a new AWS account”**. Or, if you see **“New to AWS? Sign up”**, click that.



4. Enter Email & Account Name

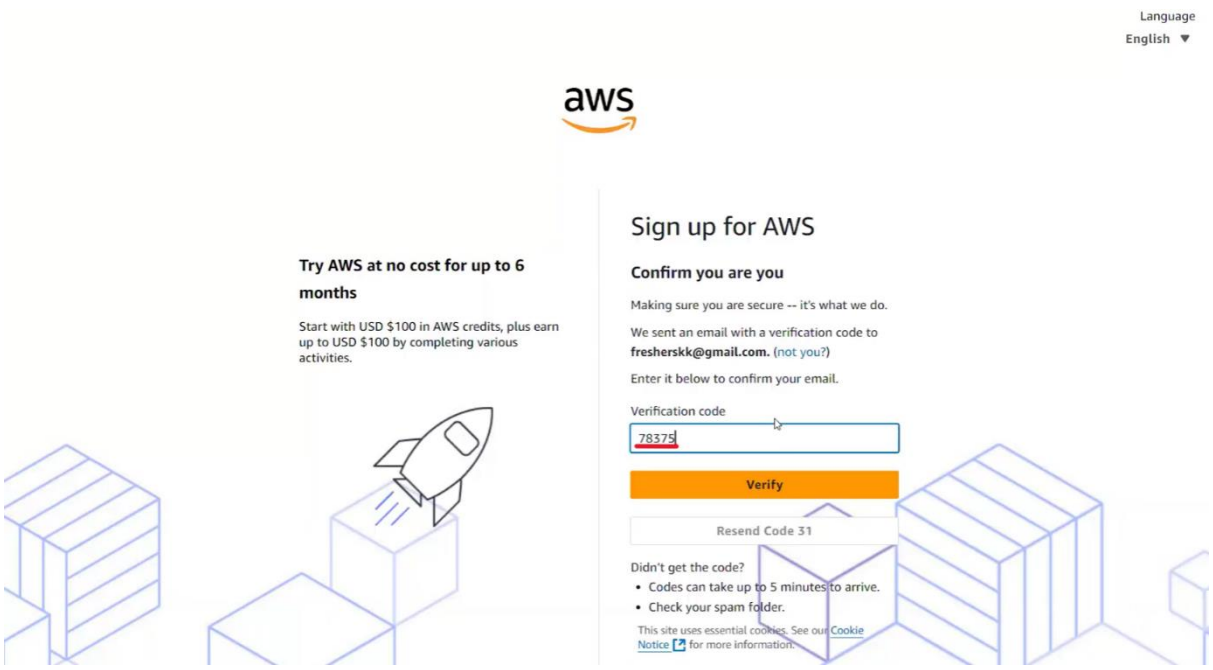
- Provide a valid **Email address** and a unique **Account name** (any name you like)



- Click **“Verify email address”**.
- AWS will send a **verification code** to your email.

5. Email Verification

- Go to your **Gmail inbox**, copy the **verification code**.

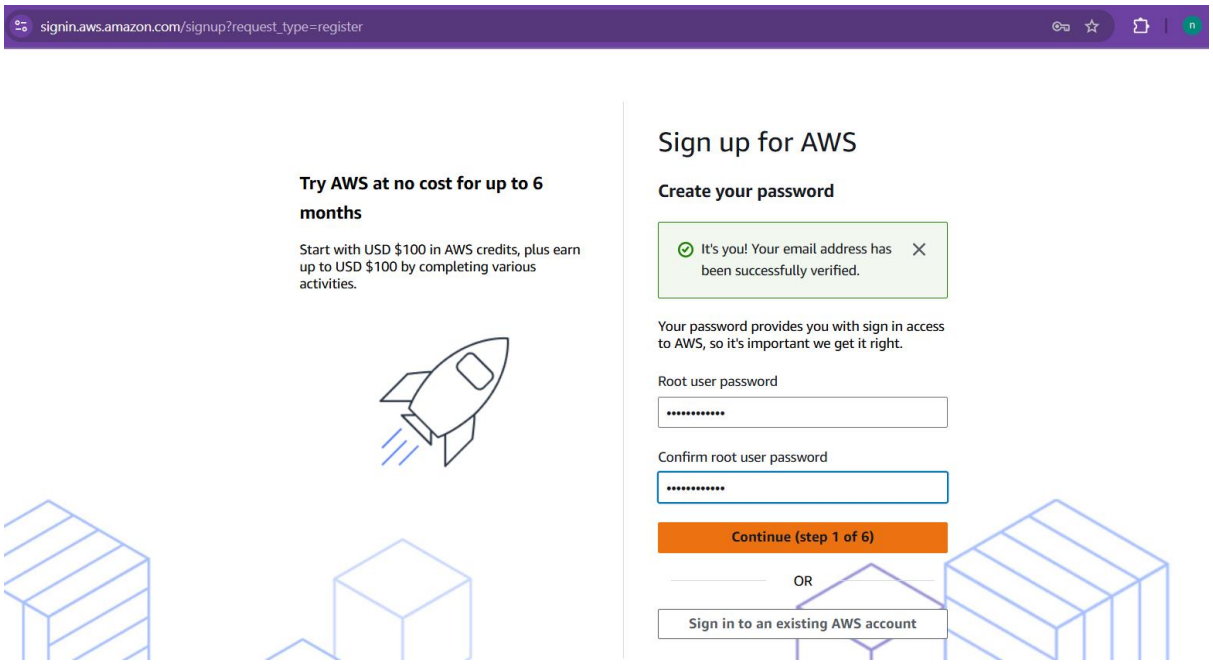


- Paste the code in the box and click **"Verify"**.

6. Set Root User Password

1. Root User Password

- Enter a strong password
- Confirm the password.



Choose Your Account Plan

You have **two options**:

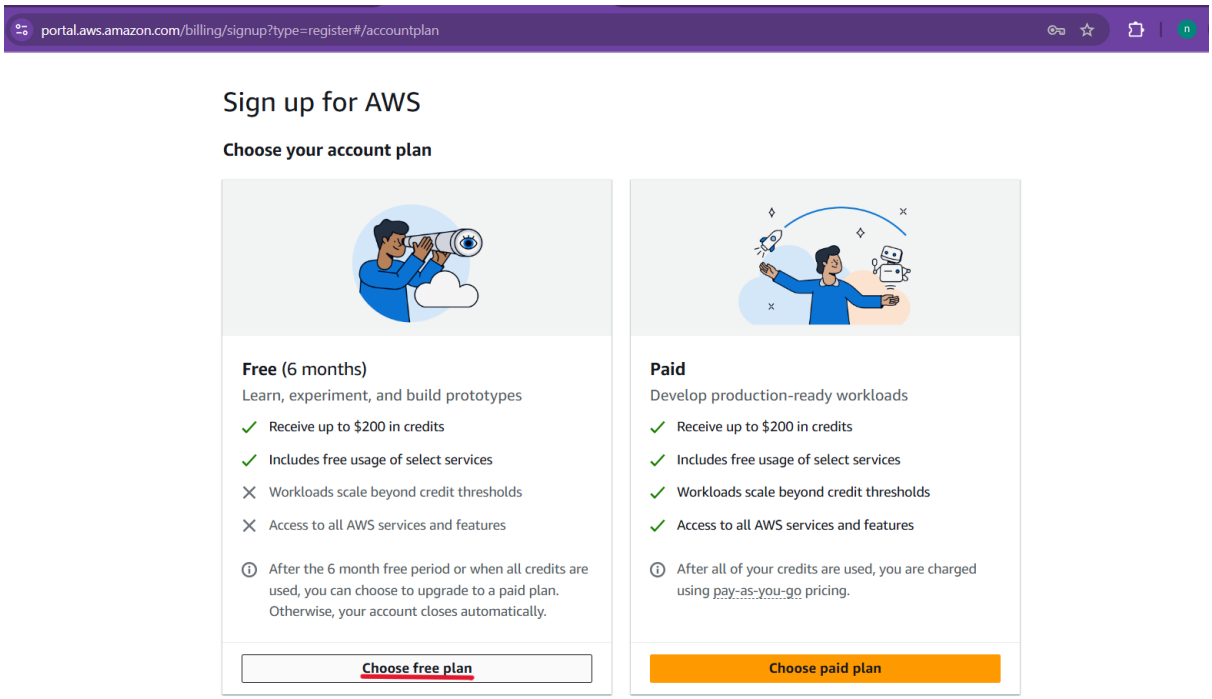
Free (6 months) – Best for learning & practice

This plan is ideal for **students, beginners, and practice purposes**.

- You will receive **up to $200 AWS credits** to use selected AWS services.
- Many AWS services are available **free within specific limits**, such as EC2, S3, and others.
- You can **learn, practice, and build small projects** without paying any money.
- No charges will be applied as long as you stay within the free usage limits and credits.

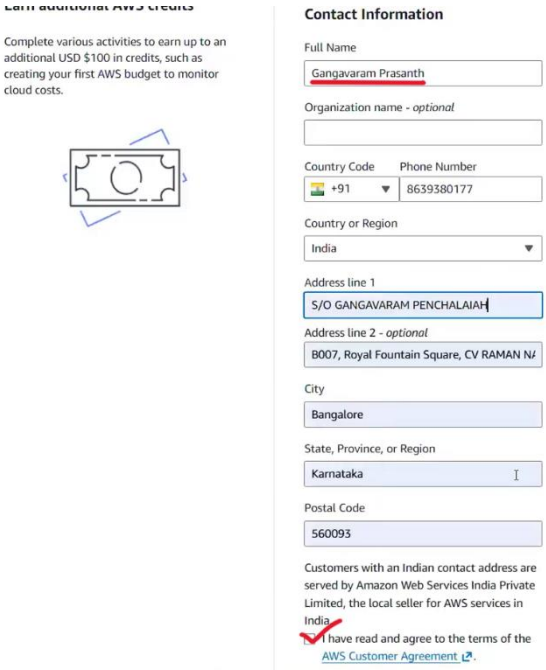
Important Note:

- After **6 months**, or if the **credits are fully used**, the account will **automatically close** unless you upgrade to a paid plan.
- To continue using AWS services after that, you must switch to the **Paid Plan**.



2. Account Information

- Enter your **Full name**.
- Enter your **Phone number**.



- Enter your **Country/Region, Address, City, State, Postal code**.
- Agree to the terms.

3. Payment Information

AWS requires a **payment method** for **identity verification**.
You will **not be charged** if you stay within the **Free Tier limits**.

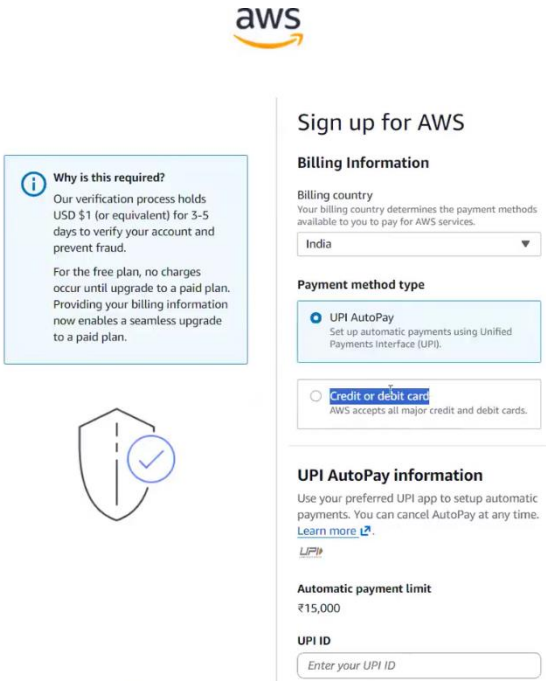
AWS supports:

- **UPI AutoPay (India)**
- **Credit / Debit Card**

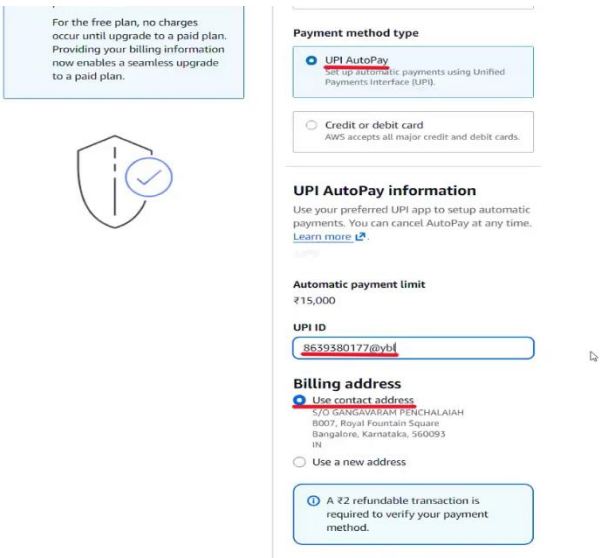
Option 1: Payment Using UPI (Recommended for India)

UPI AutoPay Setup

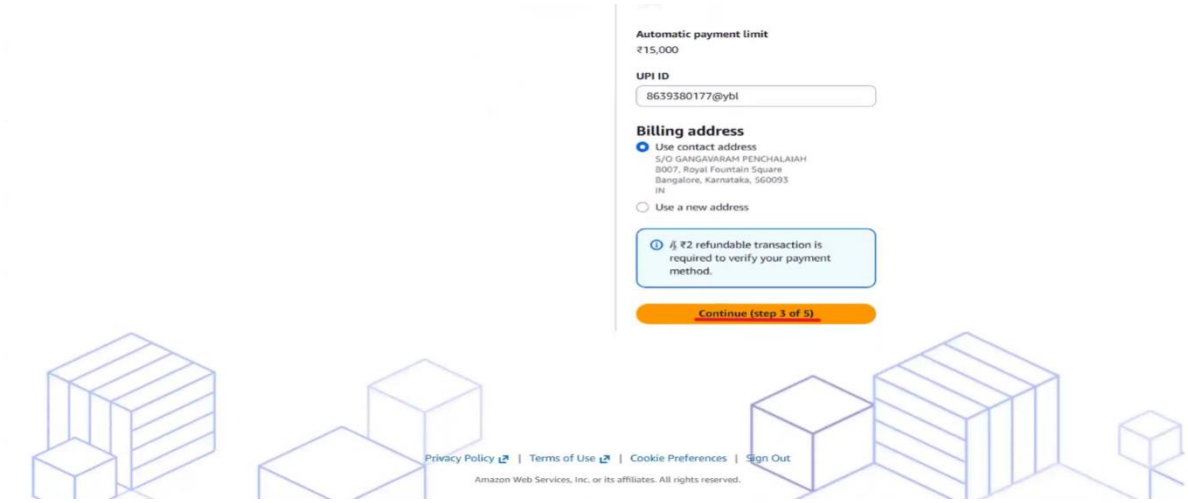
- Select **UPI AutoPay** as the payment method.
- You must **enable AutoPay** during setup.



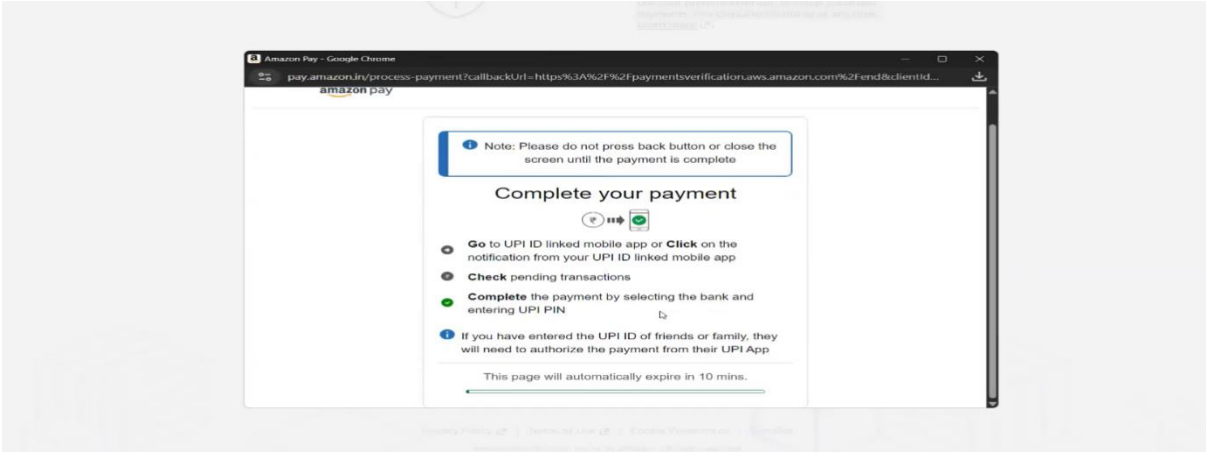
- **Automatic Payment Limit**
- AWS sets an **AutoPay limit of ₹15,000**.
- This **does NOT mean AWS will charge ₹15,000 immediately**.
- This is only the **maximum allowed limit** if you later use paid services.



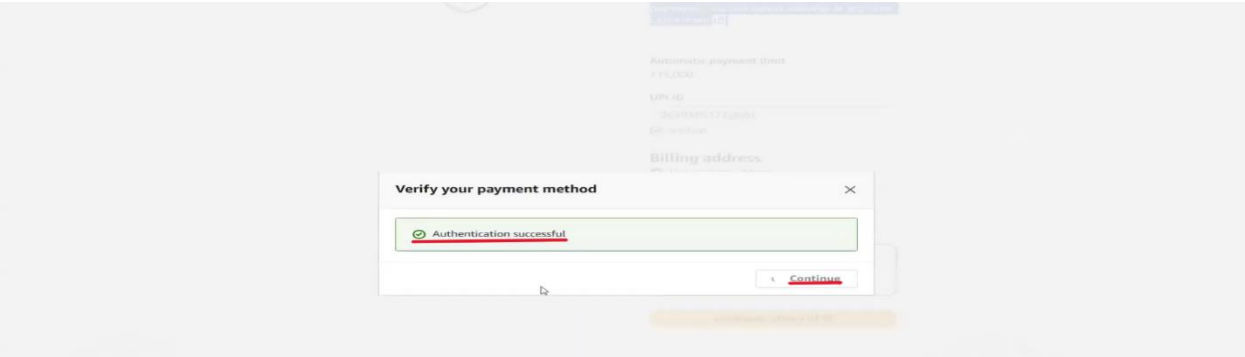
- Select **Use contact address** (recommended).



- A new **Amazon Pay** window will open.
- You will see the message **"Complete your payment"**.

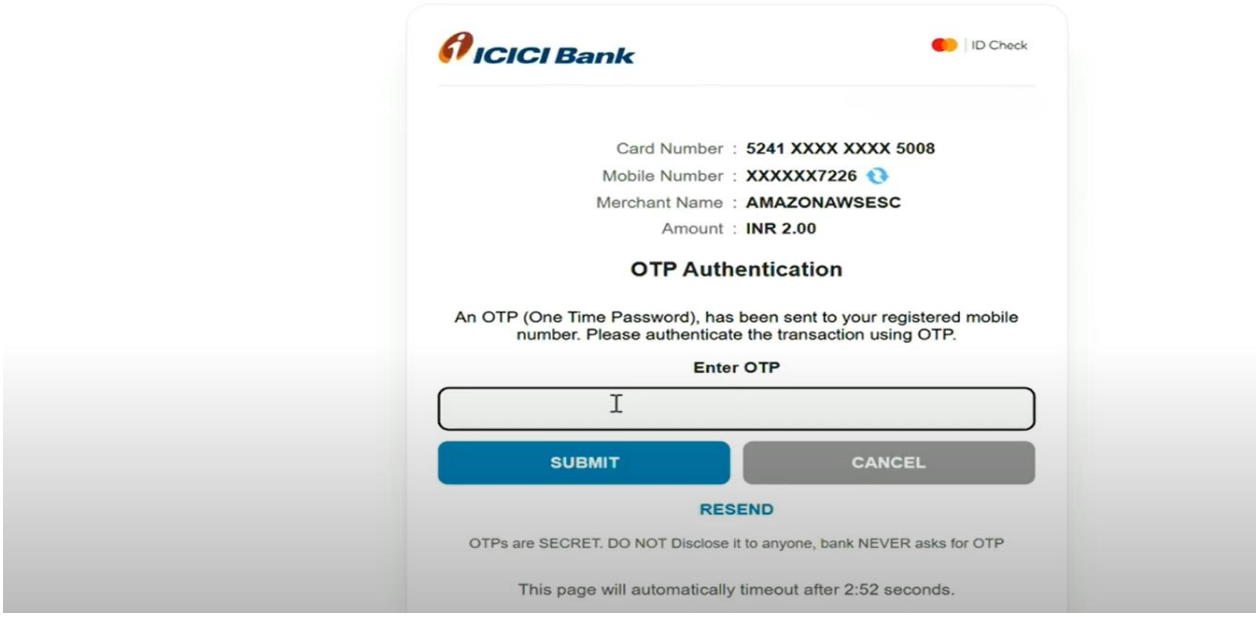


- Open your **UPI app** (Google Pay / PhonePe / Paytm / BHIM).
- Check for a **pending request / notification**.
- Approve the request by:
- Selecting your **bank**
- Entering your **UPI PIN**
- **Do not refresh, go back, or close the browser** until payment completes.

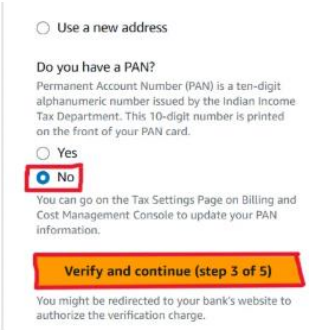


Option 2: Payment Using Credit / Debit Card

Important Note (India Users)



- Make sure **International Transactions are enabled** on your card.
- Otherwise, payment verification may fail.
- **Enter Card Details**
- AWS will place a **small temporary charge ($1 or ₹2 INR)** → this will be **refunded automatically**.
- **About PAN Card (for India users):**
- AWS sometimes asks whether you have a **PAN (Permanent Account Number)**.
- If you don't want to add it now, select **No** and continue. You can add it later.



OTP Verification

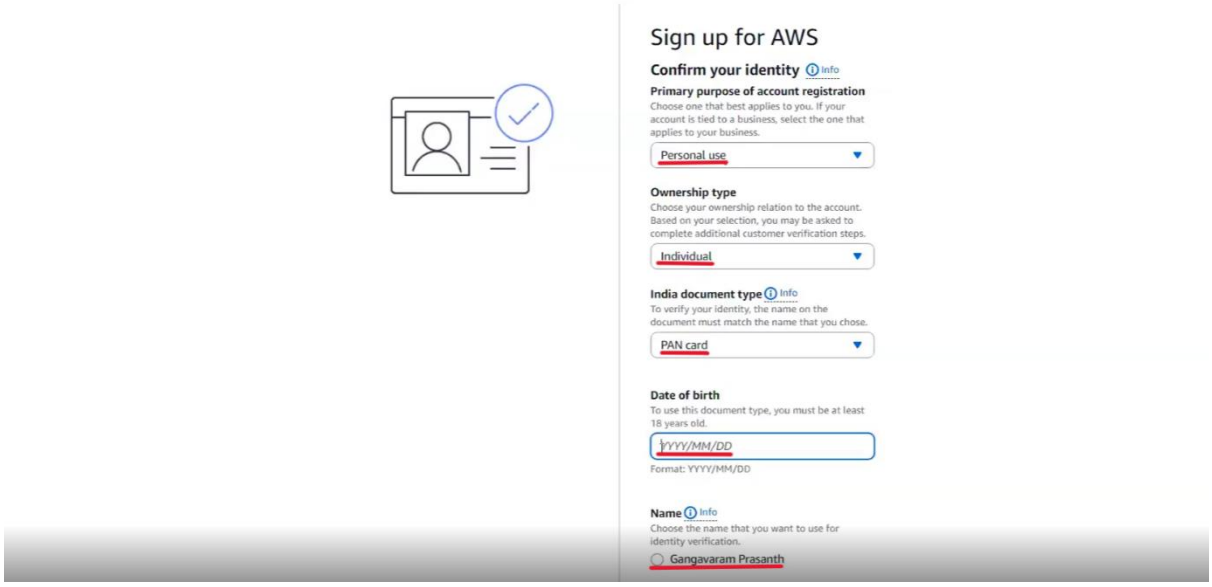
- After entering details, you'll receive a **payment OTP from your bank**.

Enter the OTP and click **“Confirm”**.

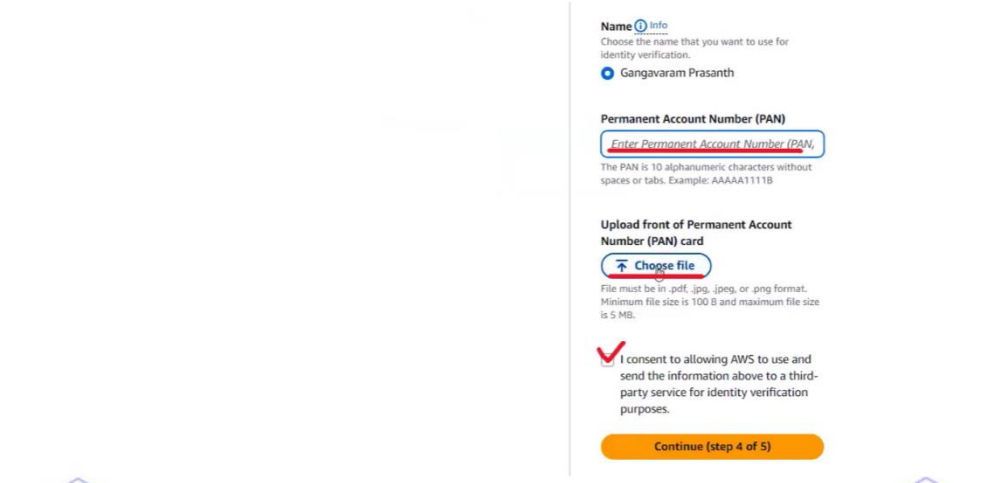
- If successful, you’ll see confirmation that your **payment method is added**.

4. Confirm Your Identity

- **Select purpose of account** → Choose **Personal use**.
- **User type** → Select **Individual**.
- **Identity verification document** → Choose **PAN Card**.
- Enter your **Date of Birth (DOB)** exactly as per PAN card.



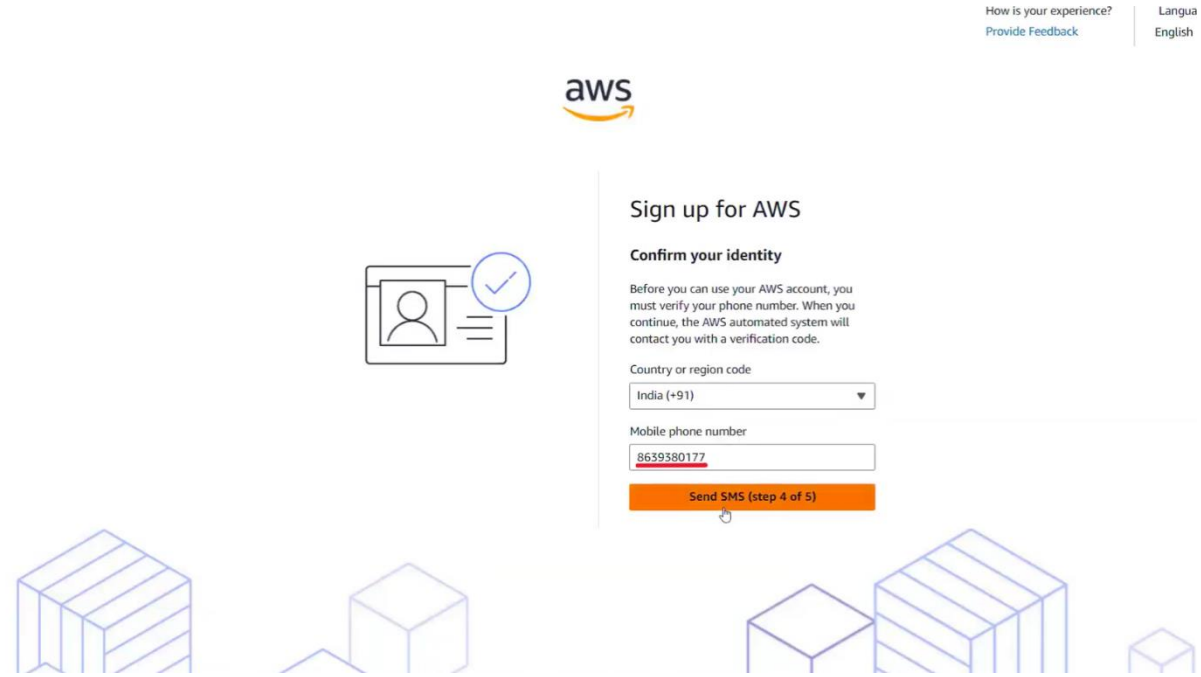
- Enter your **PAN card number**.
- Select the **account identity**
- **Upload a clear image of your PAN card** (front side).



- Make sure it’s sharp and all details are visible.
- Click **Continue**.

Mobile Verification

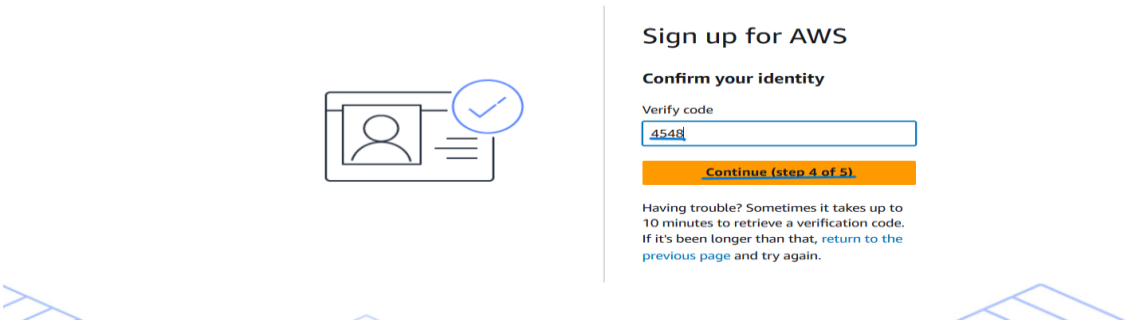
1. Enter your **mobile number**.
2. Click **Send SMS**.



3. **solve CAPTCHA** (enter the characters shown).

3. Enter the **OTP you receive**.

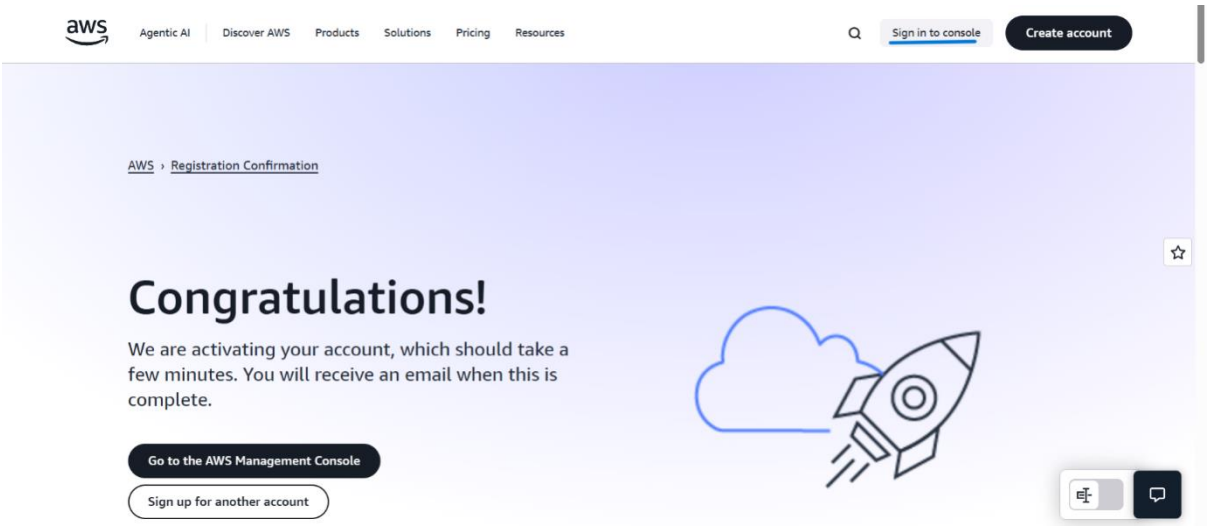
After this, your identity will be successfully verified.



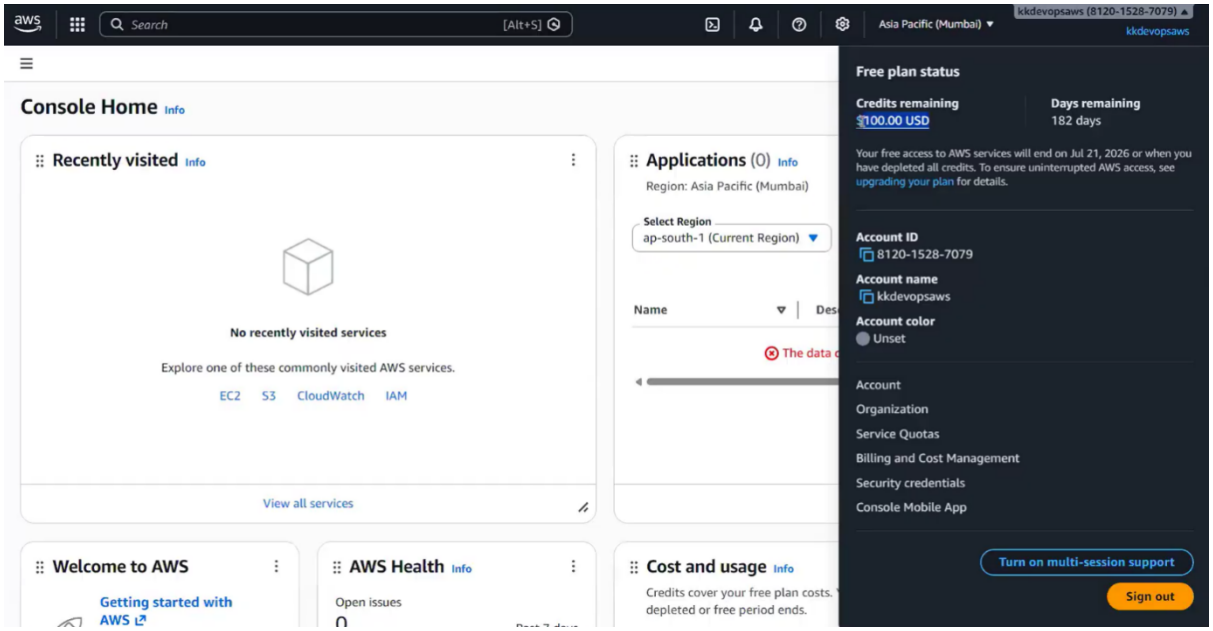
5. AWS Account Setup in Progress

- Your AWS account signup is now complete.

- You'll get a **Welcome Email from AWS**, and you can log in to the **AWS Management Console**.



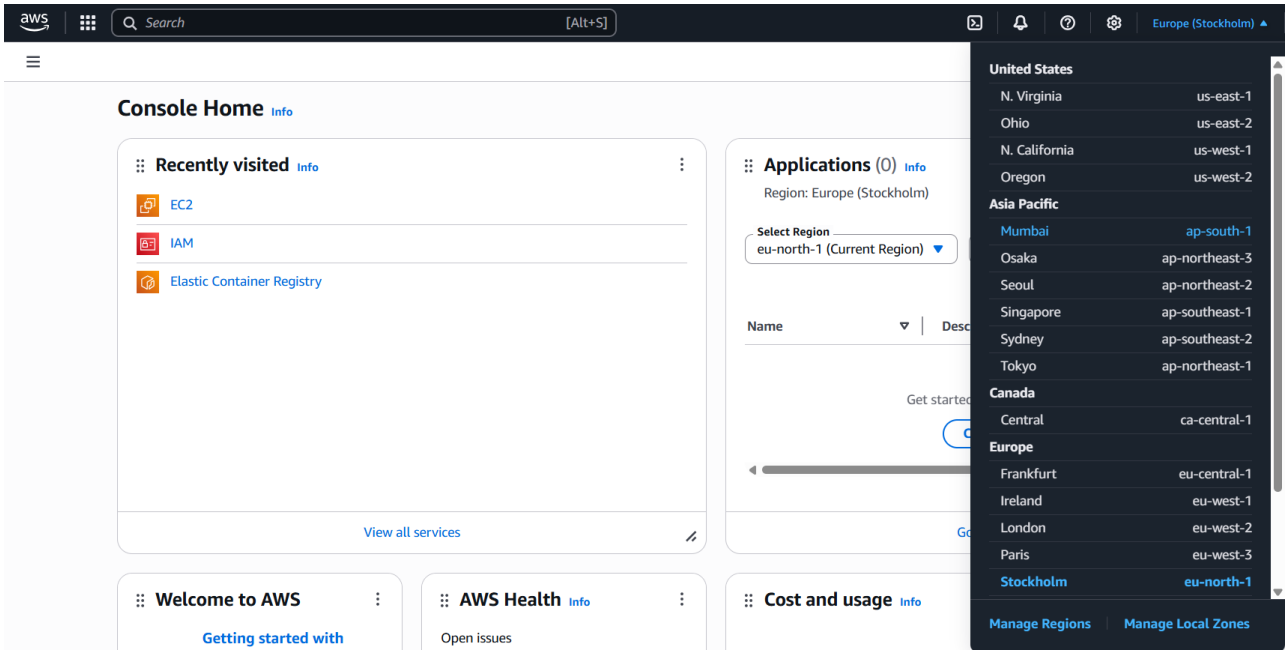
- Click **"Sign in to Console"**.
- Choose **"Root user"**, enter your **registered email**.
- Click **Next**, enter your **password**, then **Sign In**.
- You'll now be logged in to the **AWS Cloud Console** and can start using AWS services



6. Launching an EC2 Instance (AWS Virtual Server)

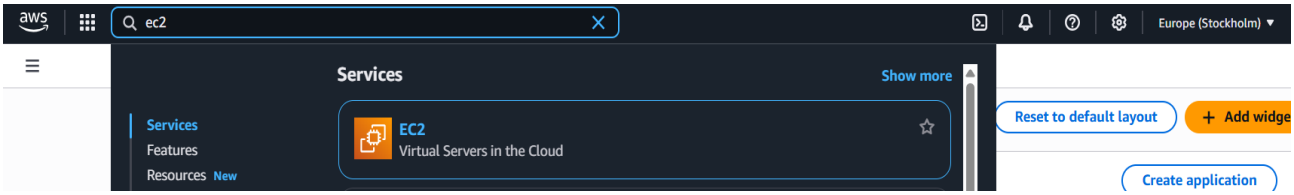
Step 1: Choose Region

- You can create the server in **any AWS region** you have access to.
- Note: Some regions might **not be available** to you due to permission restrictions or quotas.



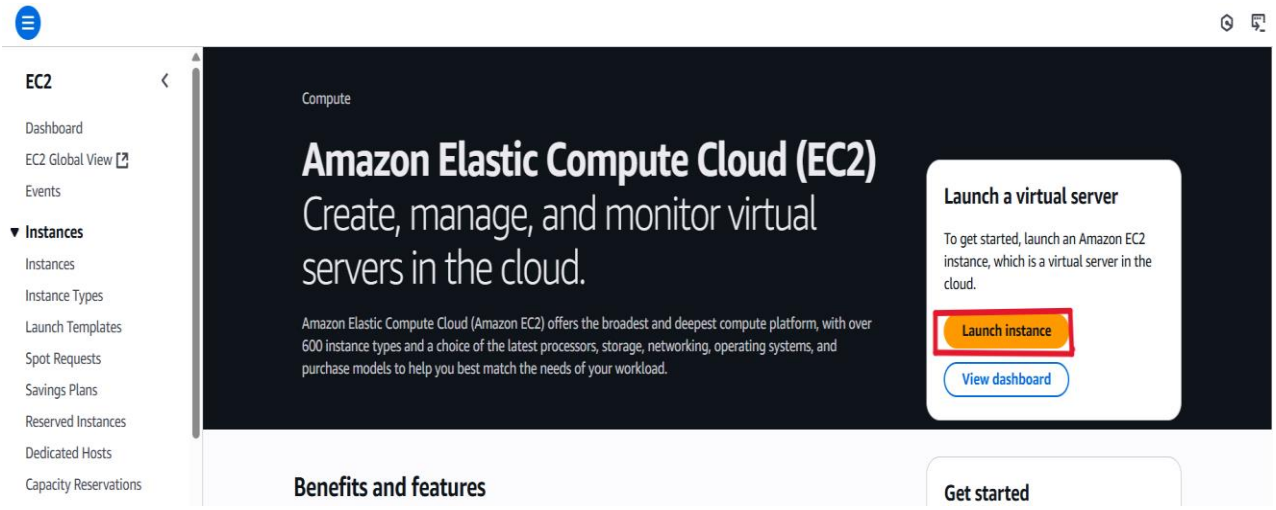
Step 2: Open EC2 Dashboard

- Search “EC2” → Click on EC2 to open the dashboard.



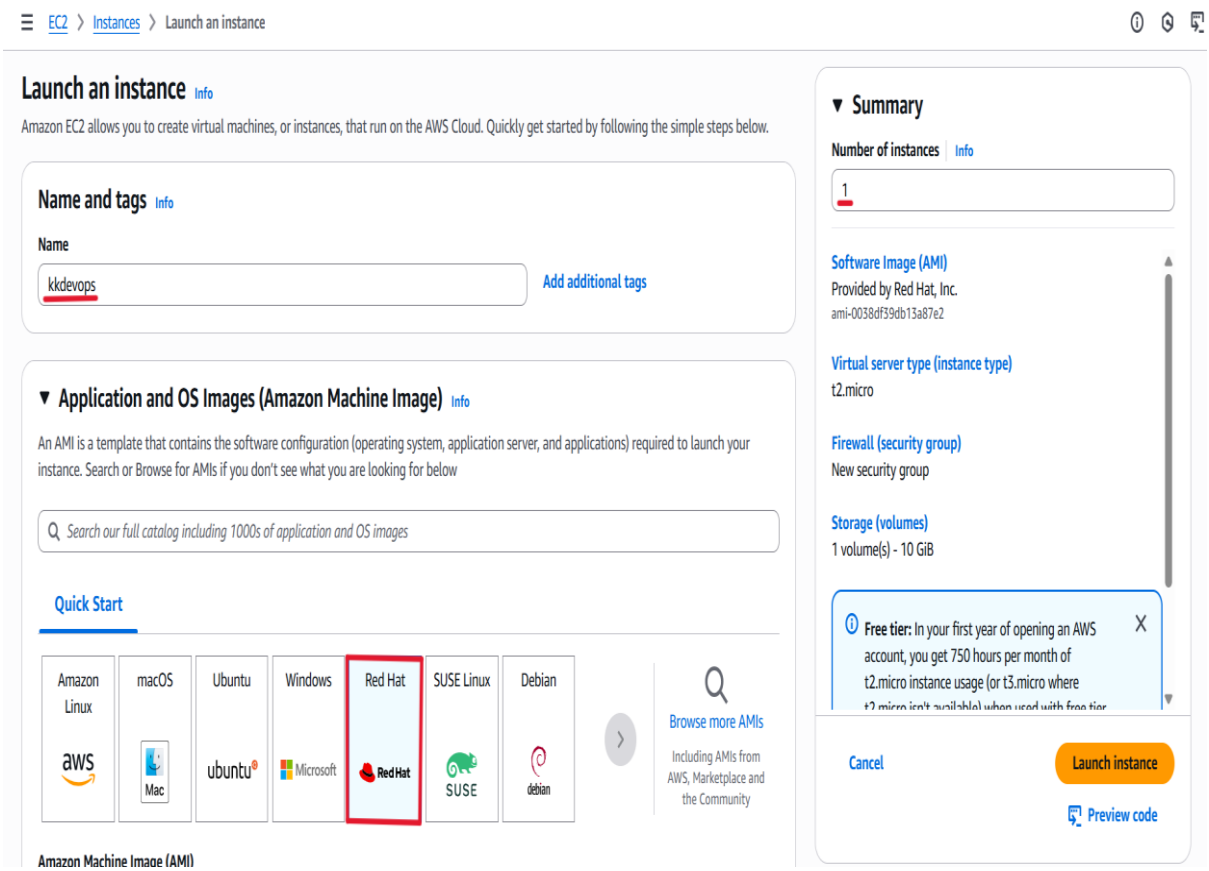
Step 3: Launch New Instance

- **Click** “Launch Instance”
- In AWS Console, **search for "EC2"** and open the EC2 Dashboard.
- Click **"Launch Instance"**.
- **Name your instance**: Example – kkdevops.



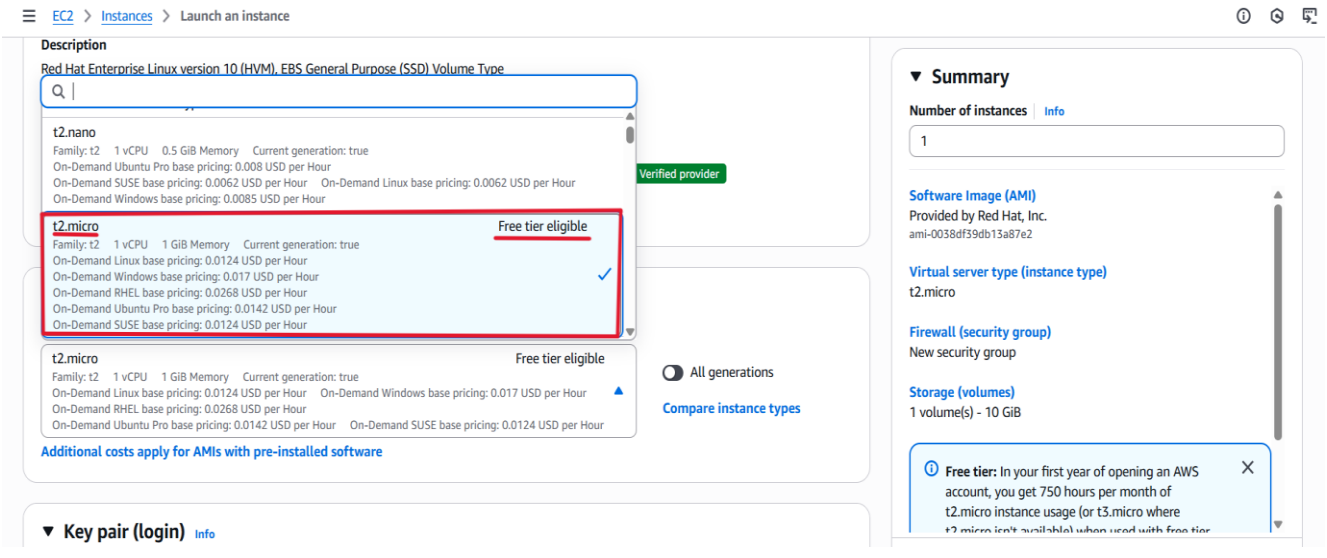
Step 4: Choose OS (AMI / Flavour)

- Choose the **OS (distribution)** you want, like **Red Hat (RHEL)**, Ubuntu, or Amazon Linux.
- These are called **AMI (Amazon Machine Images)**.



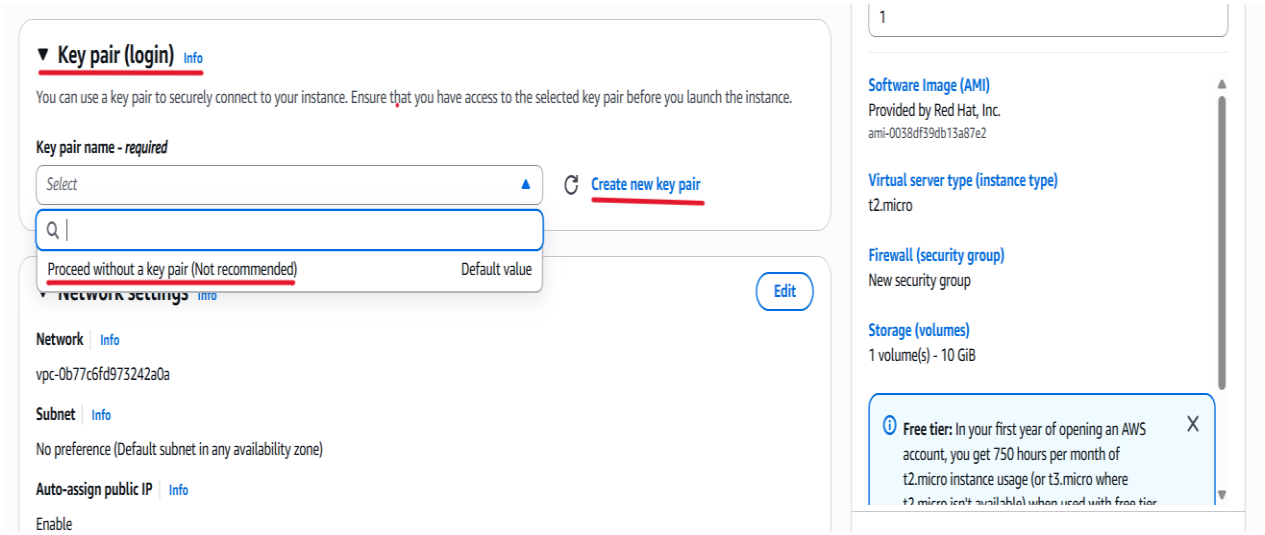
Step 5 : Choose Instance Type

- For basic use or learning, **t2.micro** is enough.
- It is **Free Tier Eligible** and works fine for testing or small projects.



Step 6 : Create or Use a Key Pair

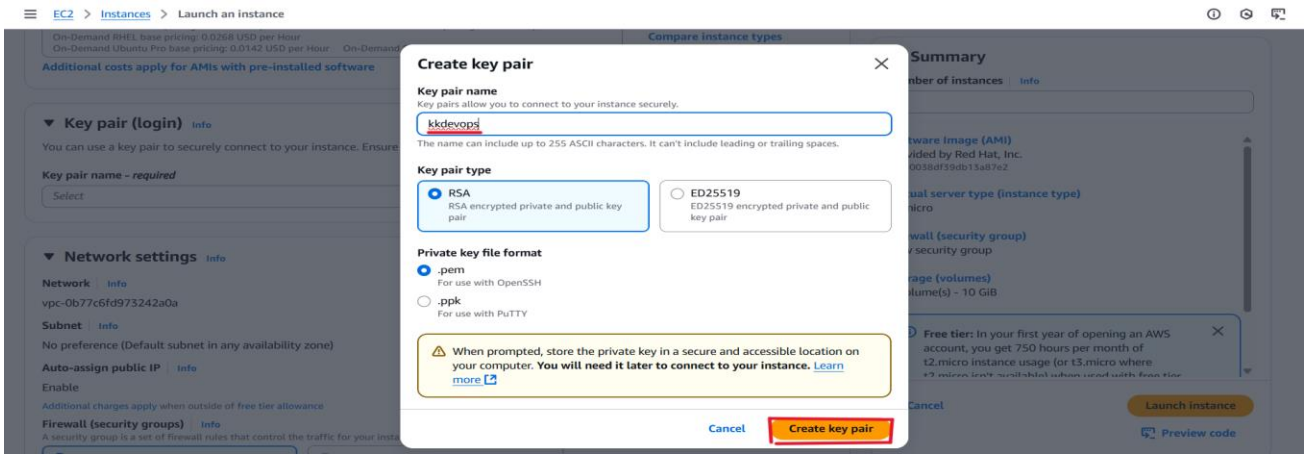
- A **key pair** is needed to **connect securely (via SSH)** to your instance.



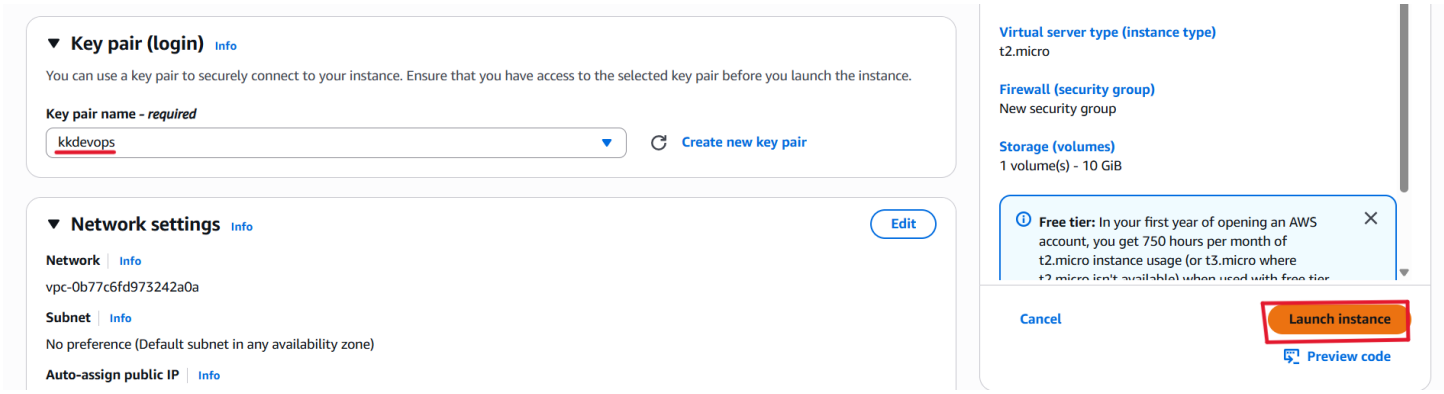
If you **don't have one**, click **Create new key pair**:

- Name: kkdevops
- Type: RSA
- Format: **.pem**
- Now, click on "Create key pair." This will automatically download the .pem file, which is usually saved in your **Downloads** folder.

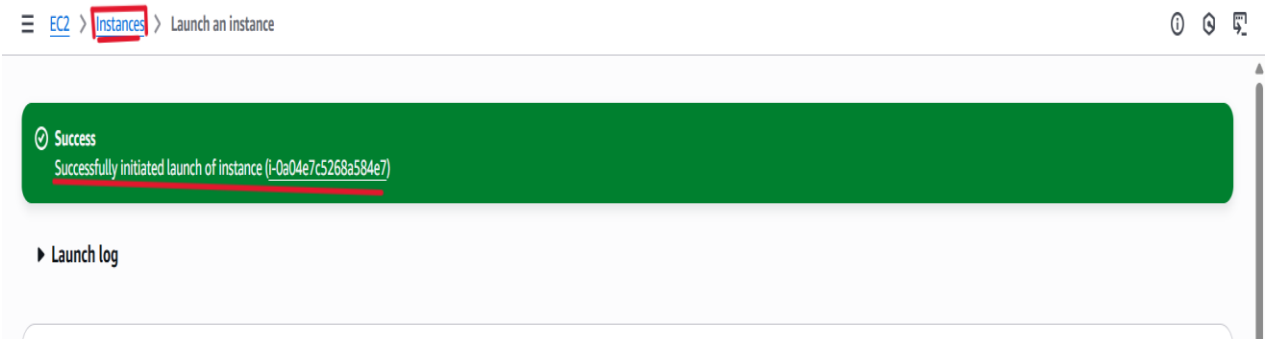
Tip: You can **reuse existing key pairs** for future instances instead of creating a new one each time.



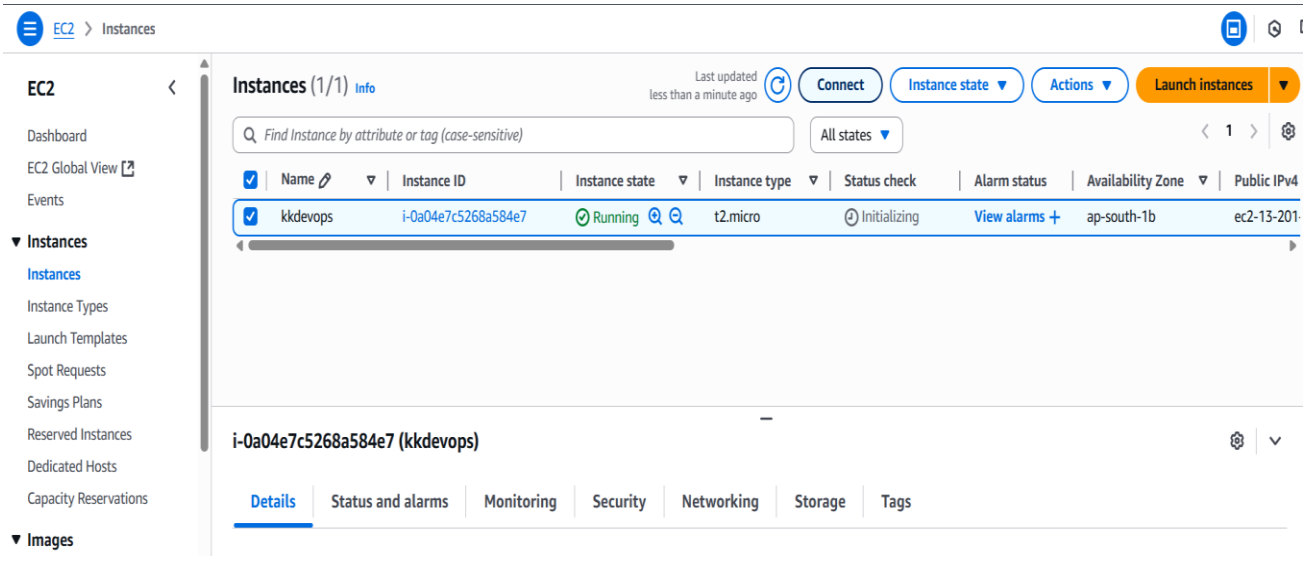
- **Click** "Launch Instance"



- Now you can see the instance launched successfully.
- Select the instance to view the launched instances.



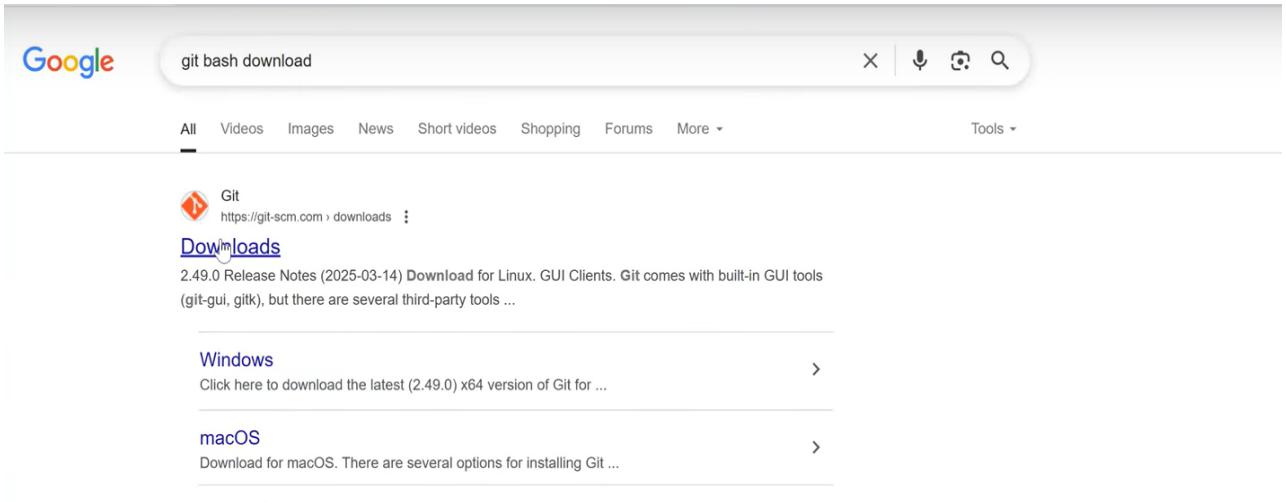
7. After Launching the EC2 Instance: How to Connect Using Git Bash Once the EC2 instance is launched successfully, we need a way to **connect to the server from our local machine**. This is where **Git Bash** comes in.



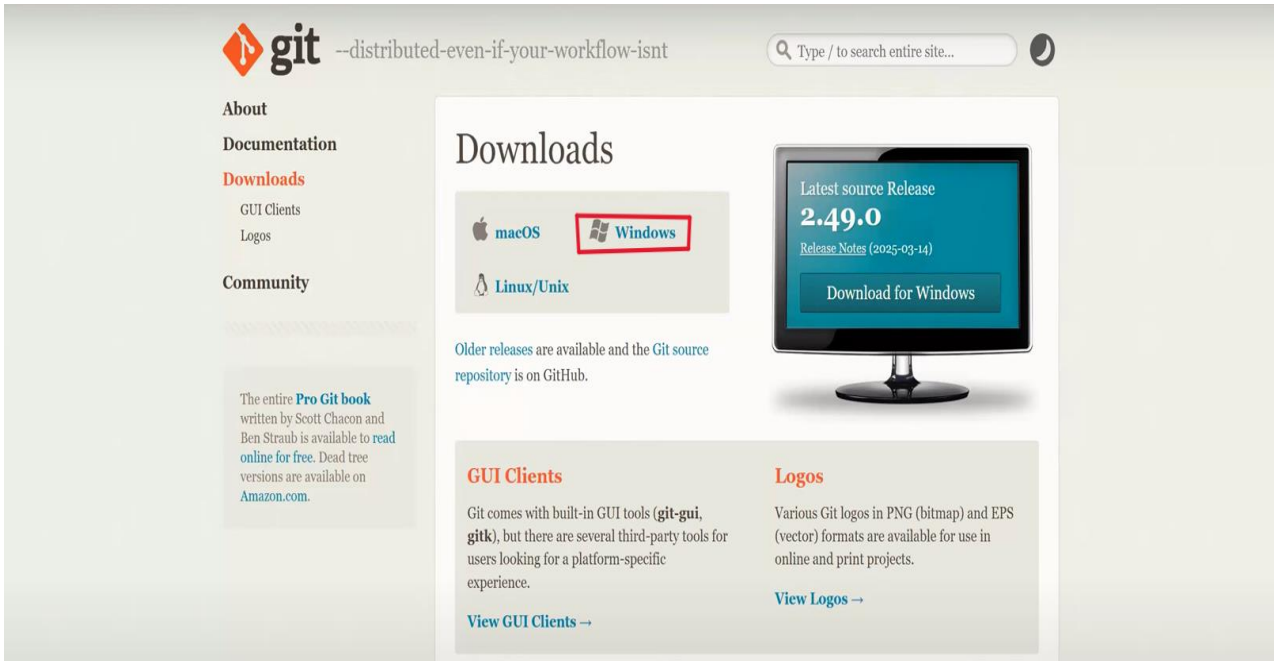
Step 1: Install Git Bash

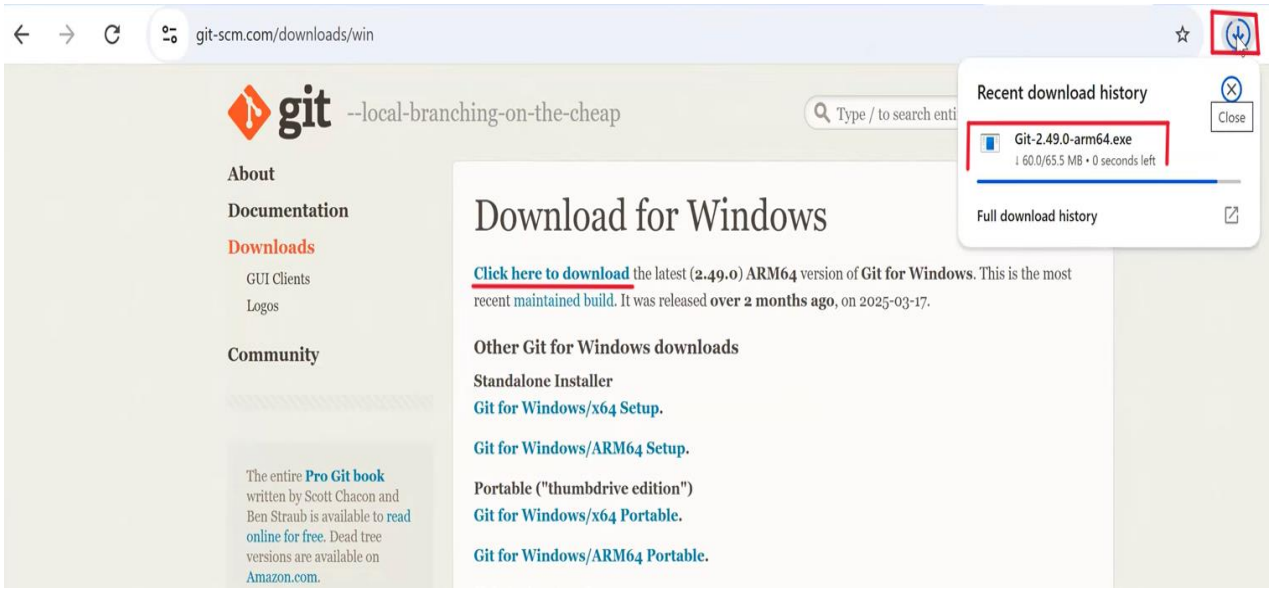
If you don’t have Git Bash on your system:

- **Open Google** and search for: Git Bash download
- Click on the link that goes to **https://git-scm.com/**

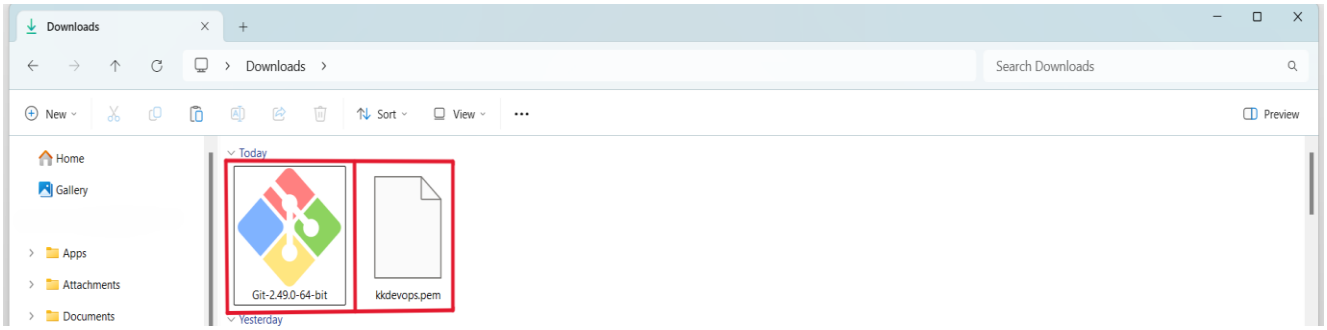


- On the site, click on **Download for Windows**

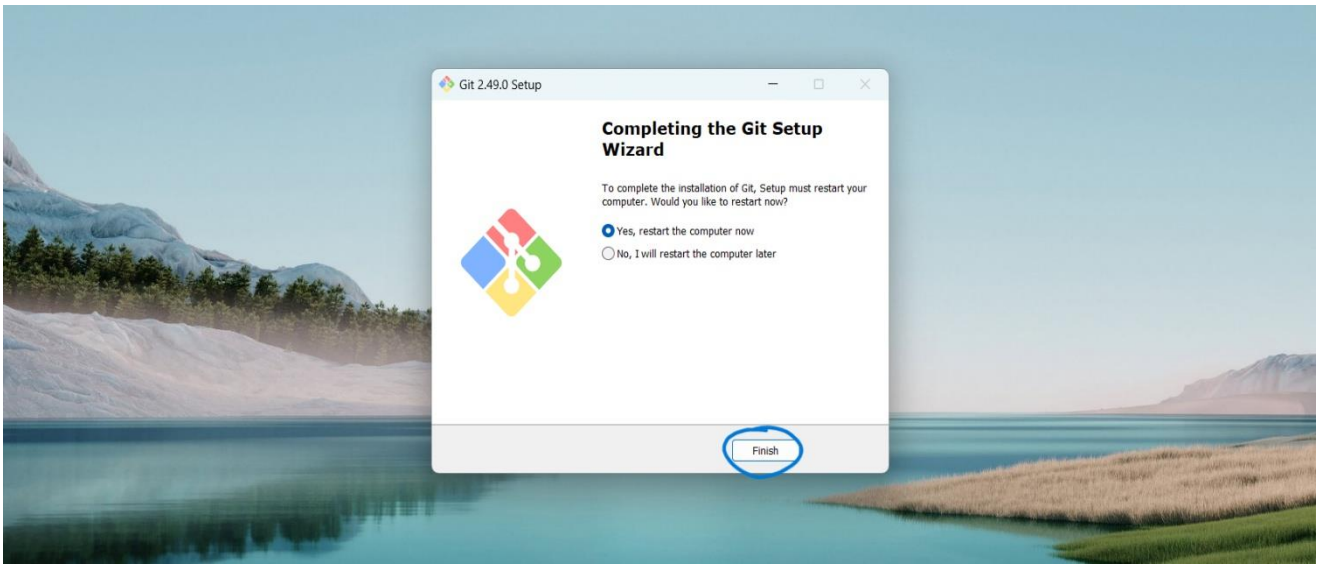




- Click on click here" to download
- After installation, you'll see **Git Bash** on your desktop or start menu.

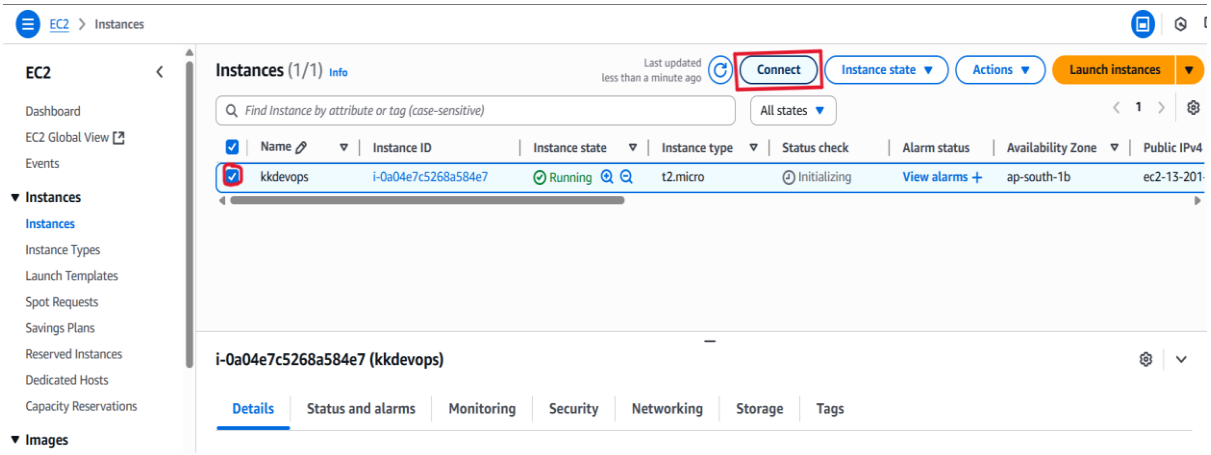


- Once downloaded**, install it by clicking Next → Next → Install**

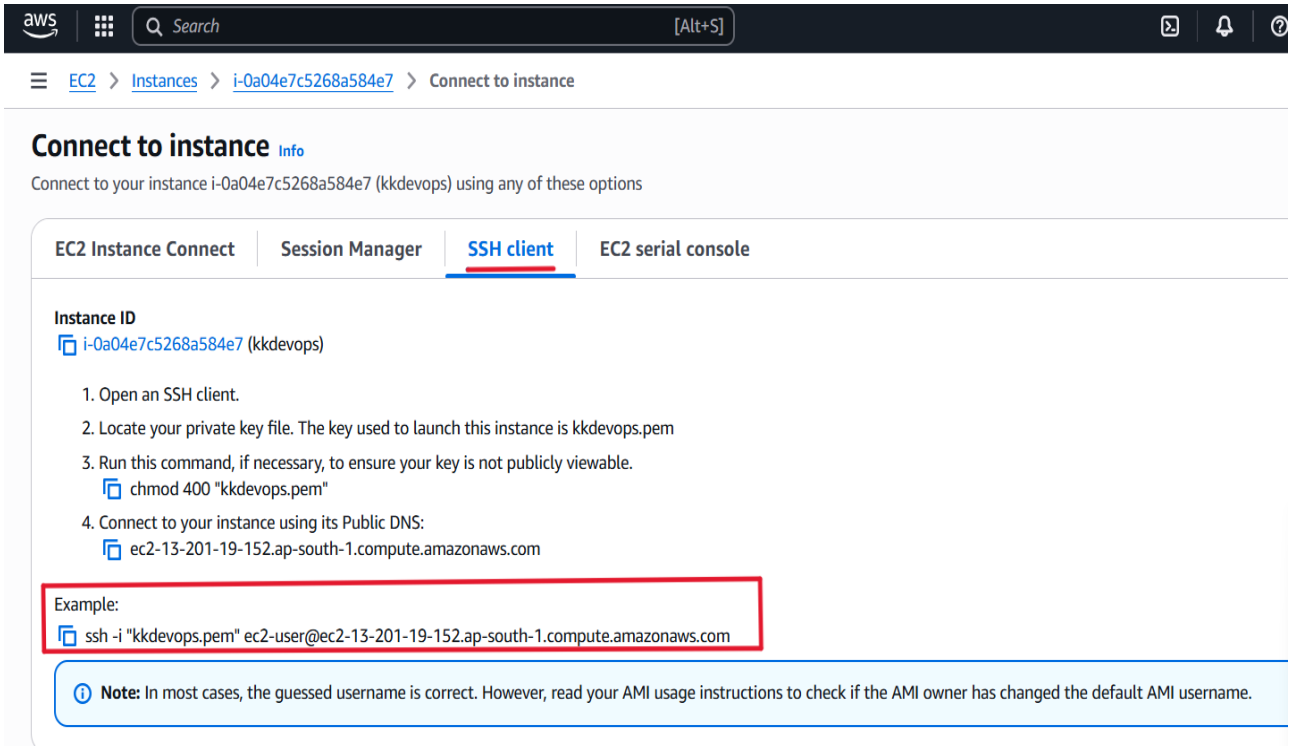


Step 2: Get the SSH Command from AWS Console

- Go to your **EC2 Dashboard**
- Select your running instance



- Cick the **"Connect"** button on top
- In the **"SSH client"** section, you'll see a command that looks like this: ssh -i "your-key.pem" ec2-user@<your-ec2-public-ip>



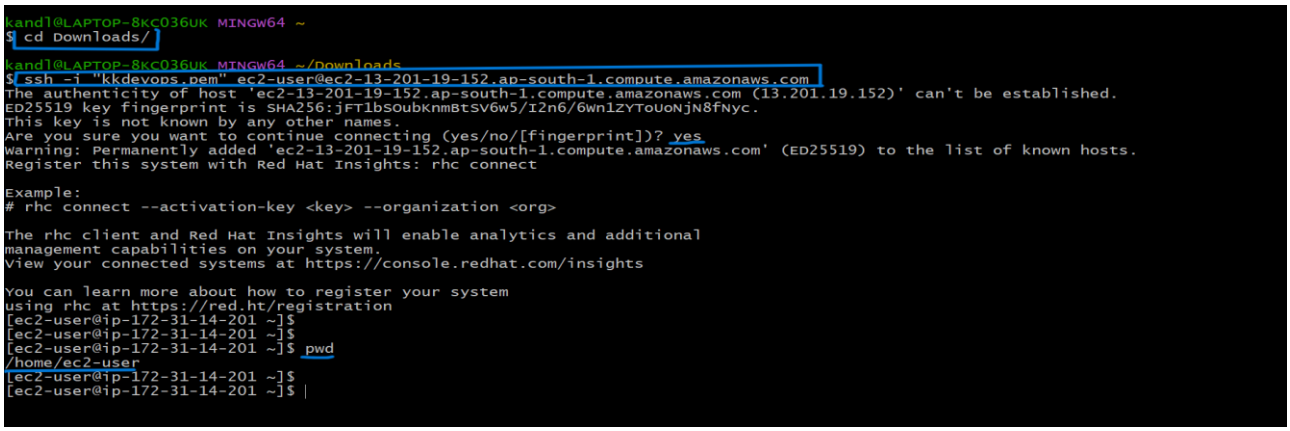
Copy that command it includes your .pem key and the EC2 public IP.

Step 4: Paste the SSH Command in Git Bash

- **Open Git Bash** on your system.
- Move to the folder where your .pem file is saved (usually in **Downloads**):

cd ~/Downloads

- Now **paste the SSH command** you copied from the AWS Console: Press **Enter**.



- The terminal may ask: Are you sure you want to continue connecting (yes/no)?
- That's it! You are now connected to your EC2 instance