**Docker Assignment 1**

**You have been asked to:**

● Pull the Ubuntu image

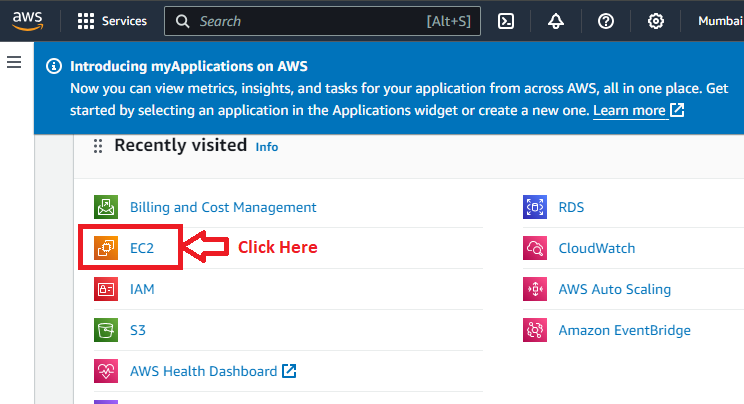
● Create a container using the Ubuntu, and map port 80 on the local

● Install apache2 on this container

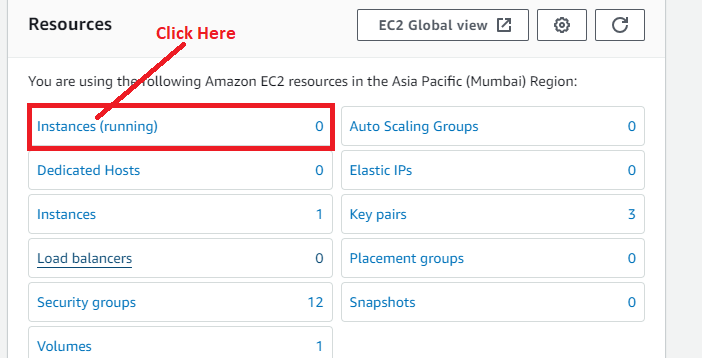
● Check if you are able to access the apache page on your browser

**A. Pull Ubuntu Container**

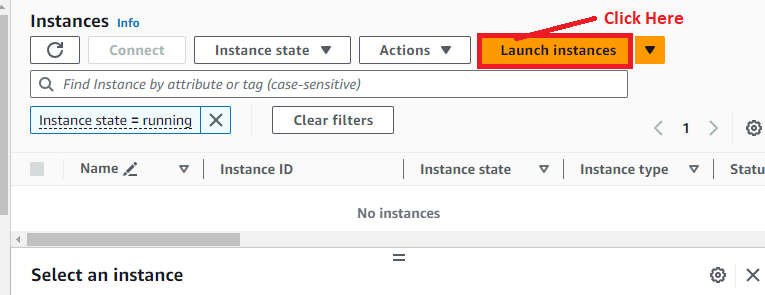
**Step 1: First, create** an **“EC2 Instance”** with the “**Ubuntu Machine”. Click** onthe **“EC2”.**

****

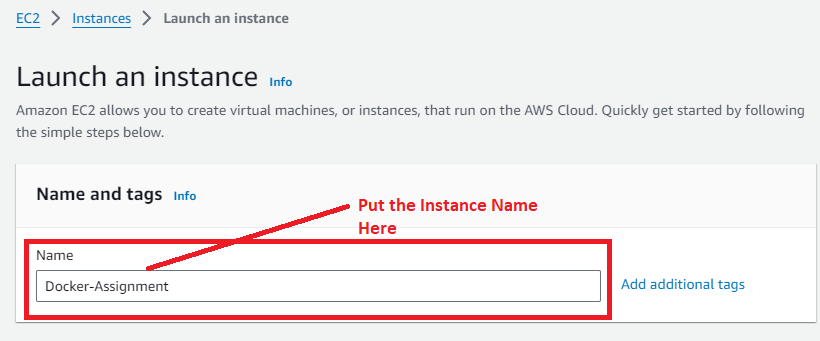
**Step 2: Click** on the **“Instances (running)”.**

****

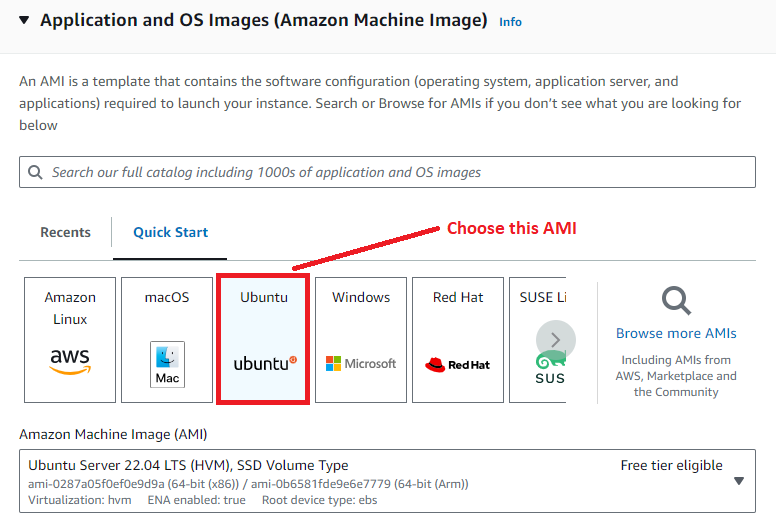
**Step 3: Click** onthe **“Launch Instances”.**

****

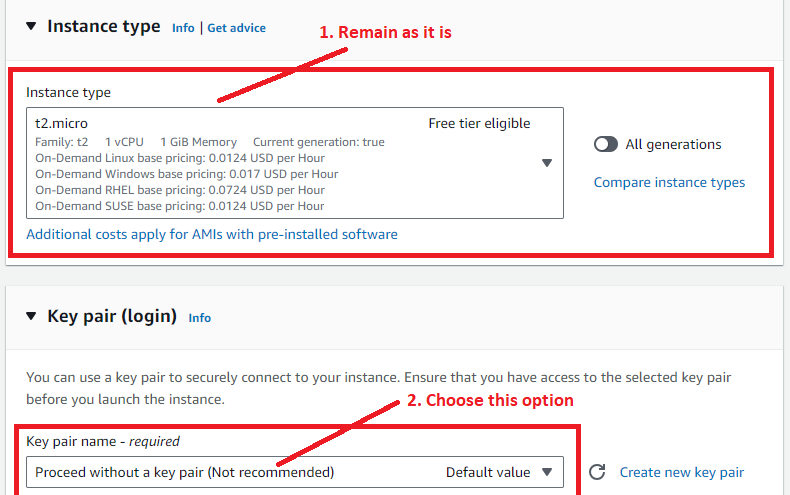
**Step 4: In** the **“Name and Tags” section, put** the **“Name”** asthe **“Docker Assignment”.**

****

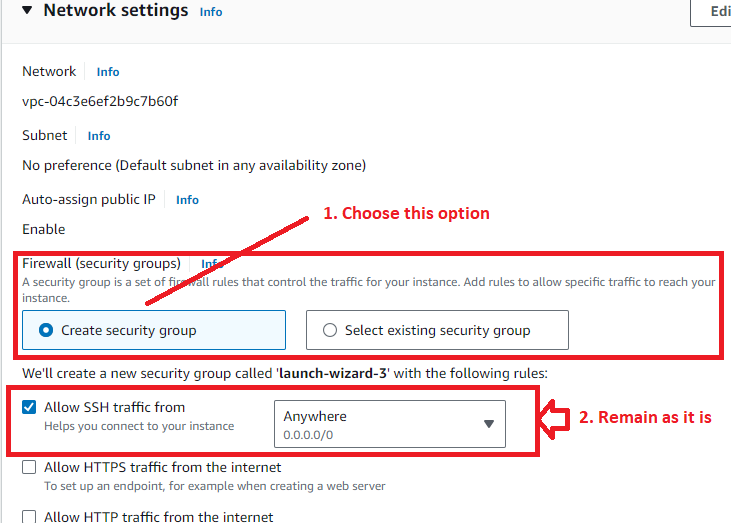
**Step 5: Choose** the **“AMI”** asthe **“Ubuntu”.**

****

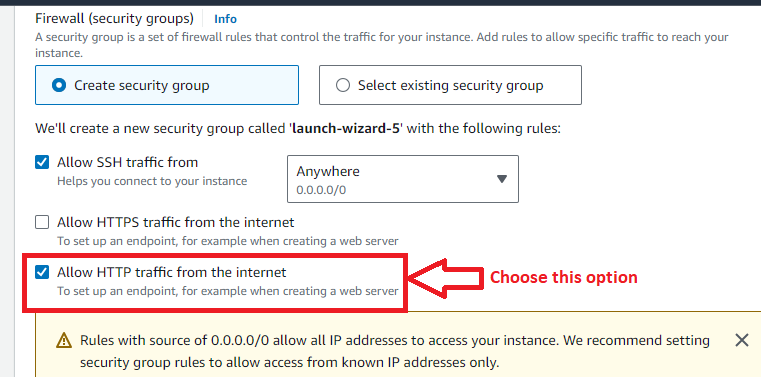
**Step 6: Select** the **“Instance type”** as the **“t2.micro” & choose** the **“key pair (login)”** as **“Proceed without a key pair (Not recommended)”.**

****

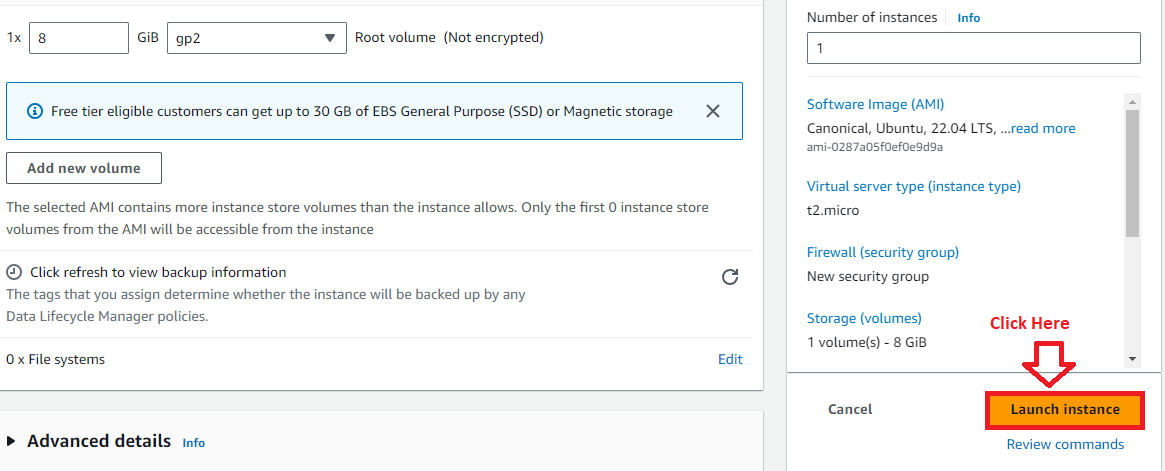
**Step 7: In** the **“Network Settings”, choose** the **“Firewall (security groups)”** as the **“Create security group”.**

****

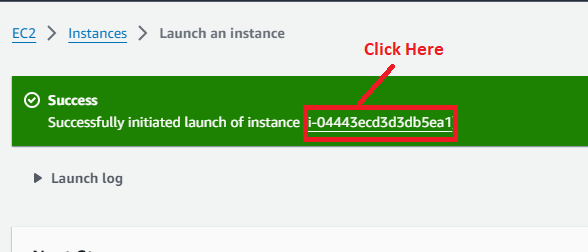
**Tick** the **“Allow HTTP Traffic from the internet” option** to **access** the **web page online.**

****

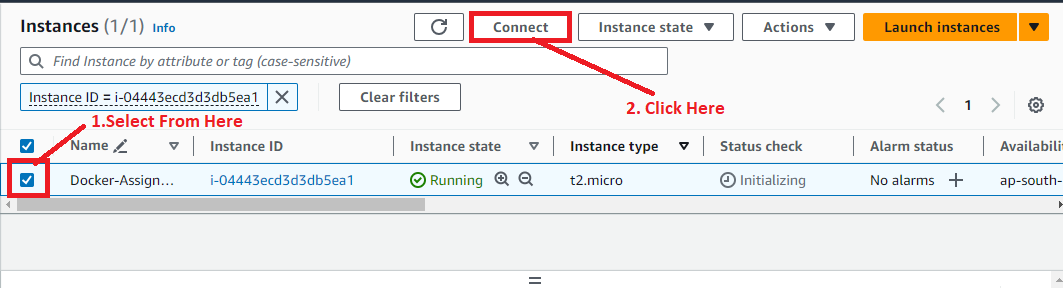
**Step 8:** **Click** on the **“Launch Instance”.**



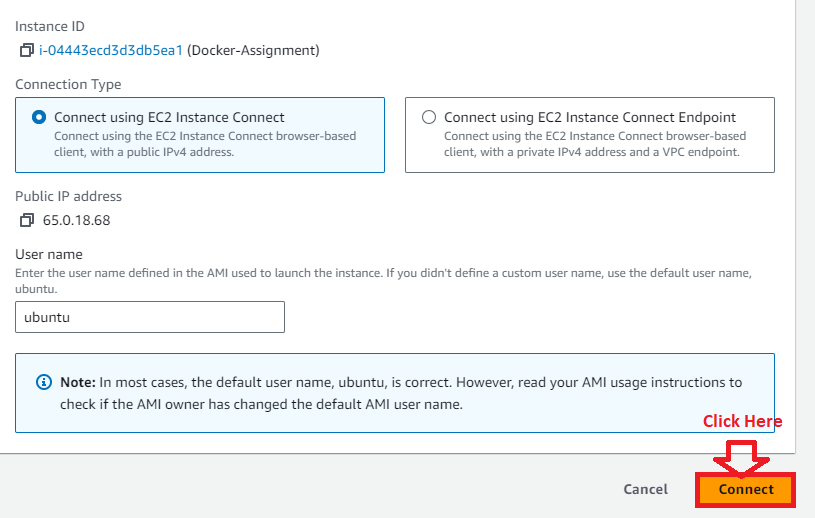
**Step 9: The instance** will be **launched successfully. Click** onthe **hyperlink.**

****

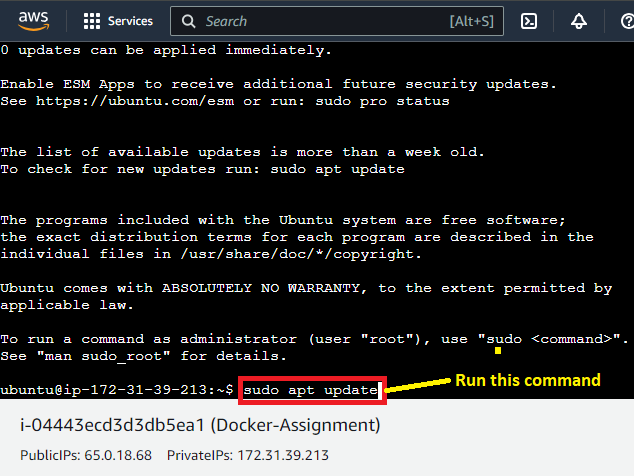
**Step 10:** **Select** the **Instance** & **click** on the **“Connect”.**

****

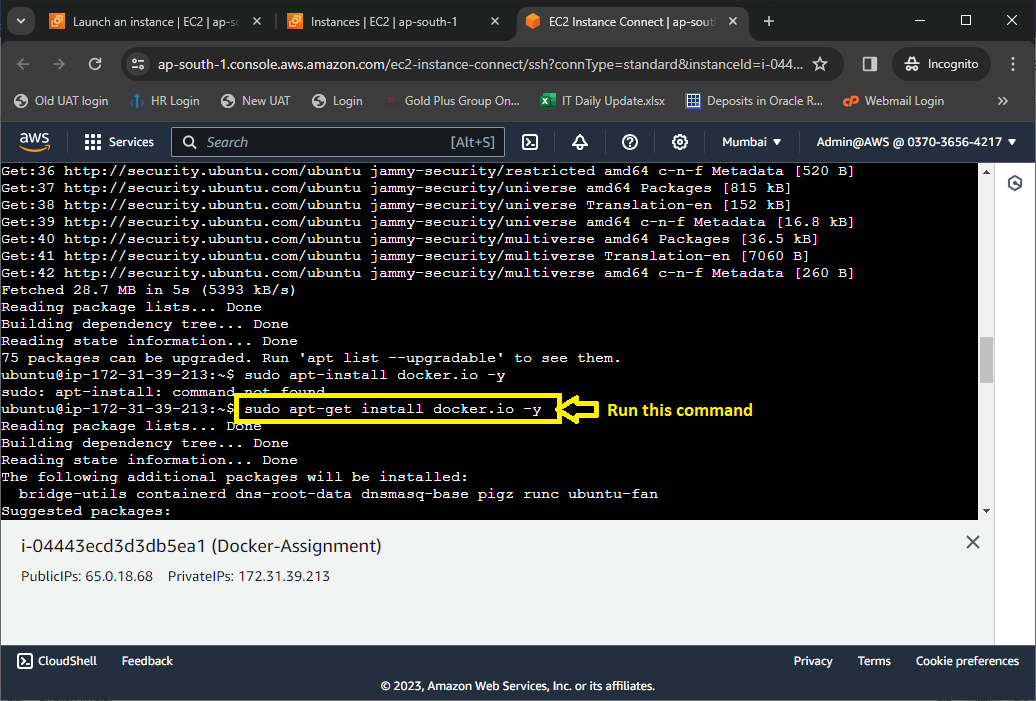
**Step 11: Click** onthe **“Connect”.**

****

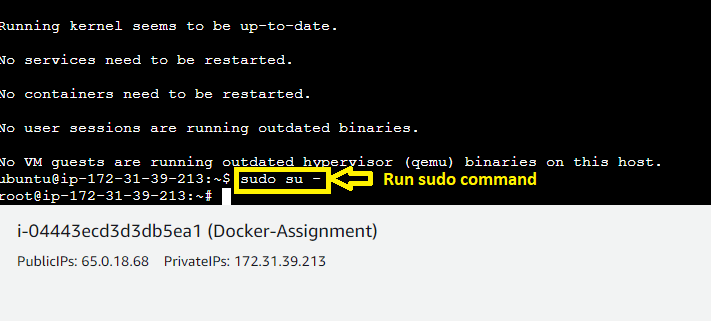
**Step 12: Run** the **“sudo apt update” command** for **updating** the **“Ubuntu” machine.**

****

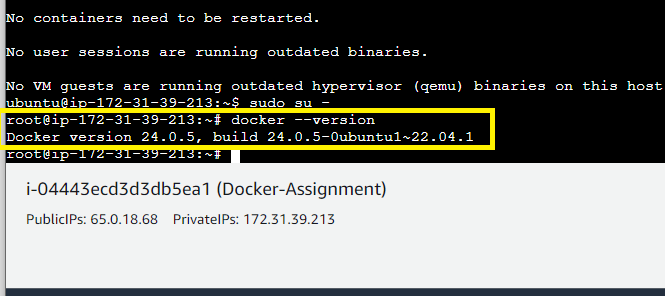
**Step 13: Install** the **“Docker” using** the **command: sudo apt-get install docker.io -y.**

****

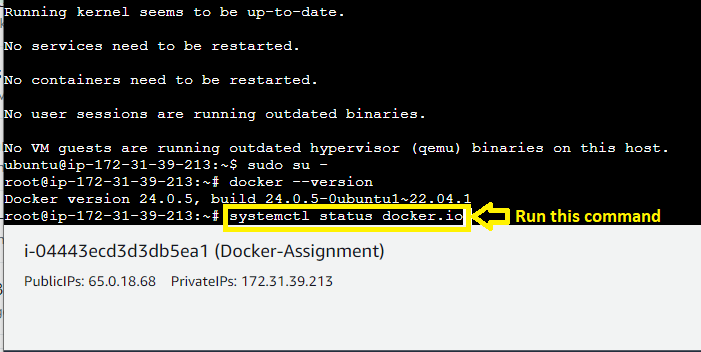
**Step 14: Run** the **“sudo su –“ command** for **accessing** the **Docker.**

****

**Step 15: To check** the **version** of **docker, type** the **“docker --version” command.**

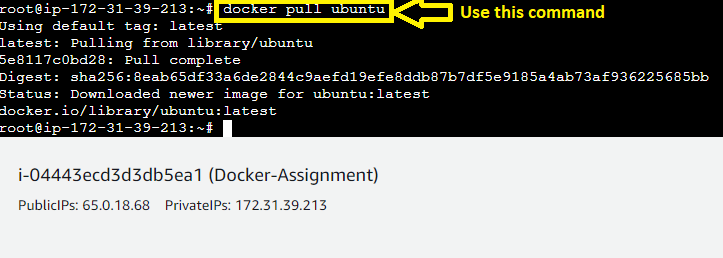


**Step 16: Check** the **status** of **docker** with **“systemctl status docker” command.**

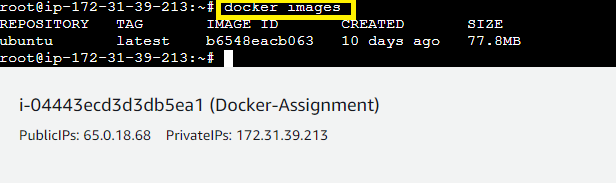


**Docker** is in **“Active” State**.

**Step 17: For pulling** the **“Ubuntu Image”, use** the **“docker pull Ubuntu”. Docker image** will be **fetched** from **docker directory.**

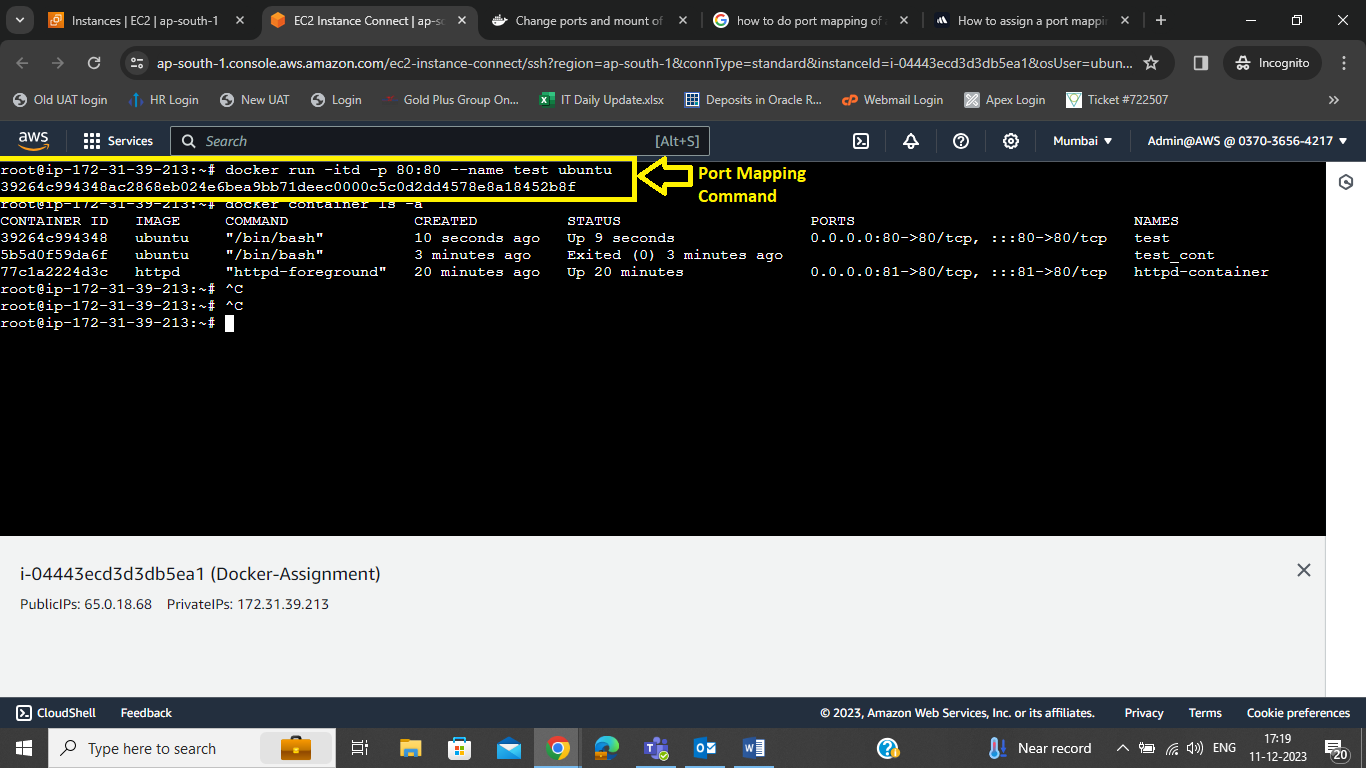
****

**Step 18: Type** the **“docker images” command** to **check** the **“Ubuntu” image.** The **“Ubuntu” image** will be **found.**

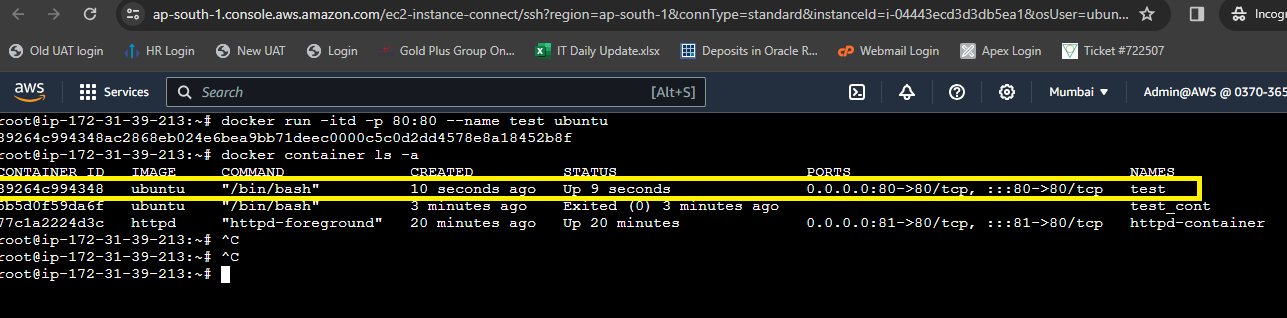
****

**B. Create a container using Ubuntu, and map port 80 on the local**

**Step 1: Run** this **command: “docker run -itd -p 80:80 --name test ubuntu”** & **it will create & port mapping** of a **container.**

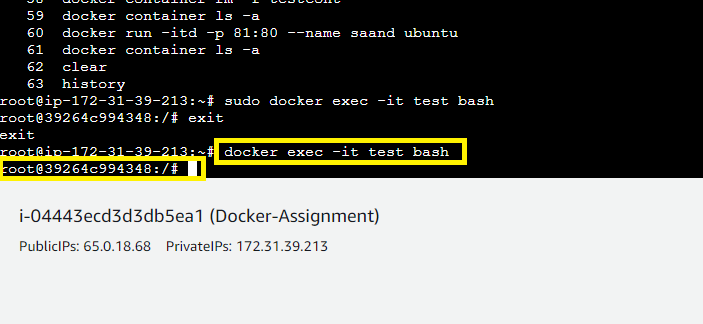
****

**Step 2: Run** the **“docker container ls –a” & you** will **notice** that **“test” container** is **mapped** on the **port 80 successfully.**

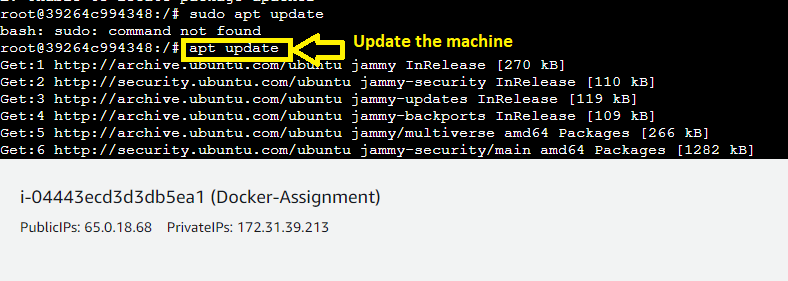
****

**C. Install apache2 on this container**

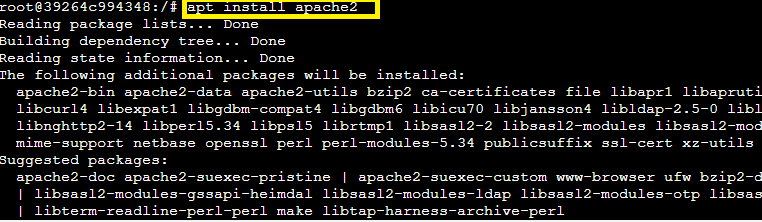
**Step 1: Go inside** the **“test” container. Use** this **command: “docker exec –it test bash”.**

****

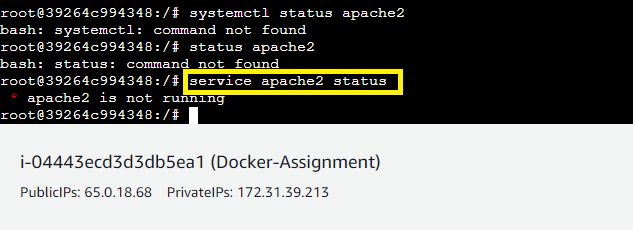
**Step 2: First, run** the **“apt update” command** to **update** the **machine.**

****

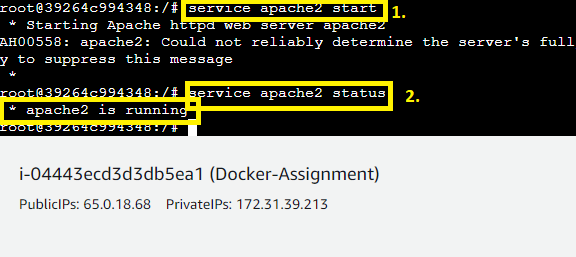
**Step 3: Run** the **“apt install apache2” command** to **Install** the **apache2 web server.**

****

**Step 4: Use** the **“service apache2 status” command** to **check** the **“Apache2” server status. The “Apache2** is not **running”.**

****

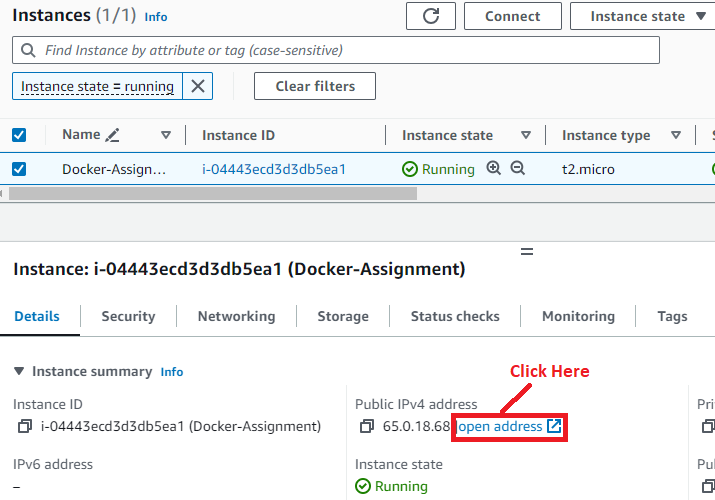
**Step 5: Use** this **command: “service apache2 start”** to **start** the **server.**

****

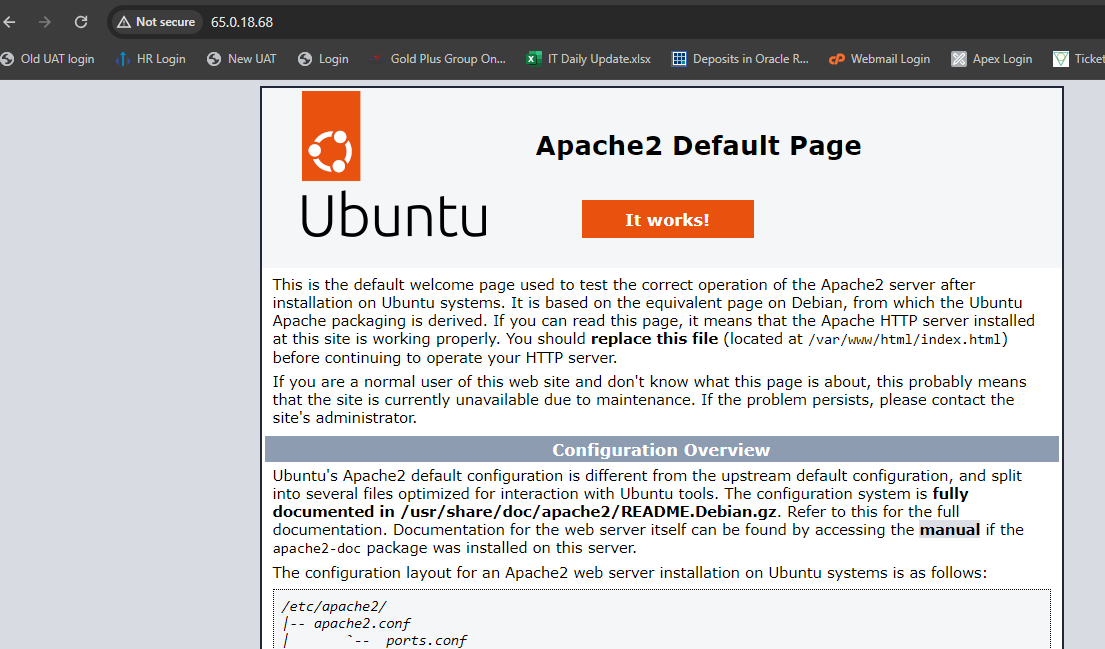
**The Apache 2** will be **start** after **running** this **command. You** can **check** the **status using** the **“service apache2 status”** command.

**D. Check if you are able to access the apache page on your browser**

**Step 1: Go** to the **“Instances” pages** & **click** on the **“open address”** in the **“Public IPv4 address”.**

****

**Step 2: The Apache web page** will be **successfully accessed.**

****