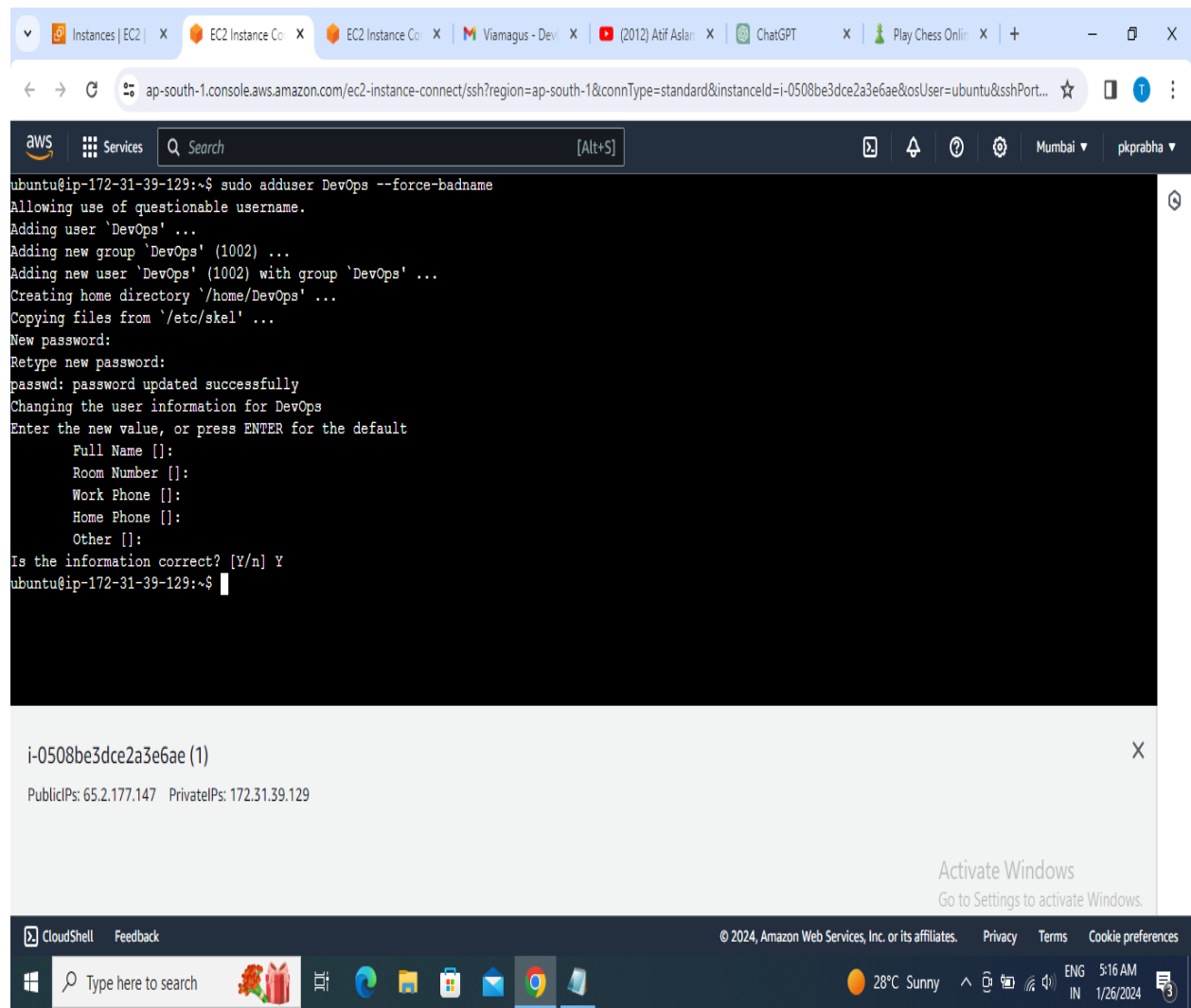


CREATE A SUDO USER



```
ubuntu@ip-172-31-39-129:~$ sudo adduser DevOps --force-badname
Adding user 'DevOps' ...
Adding new group 'DevOps' (1002) ...
Adding new user 'DevOps' (1002) with group 'DevOps' ...
Creating home directory '/home/DevOps' ...
Copying files from '/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for DevOps
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] Y
ubuntu@ip-172-31-39-129:~$
```

i-0508be3dce2a3e6ae (1)

PublicIPs: 65.2.177.147 PrivateIPs: 172.31.39.129

Activate Windows
Go to Settings to activate Windows.

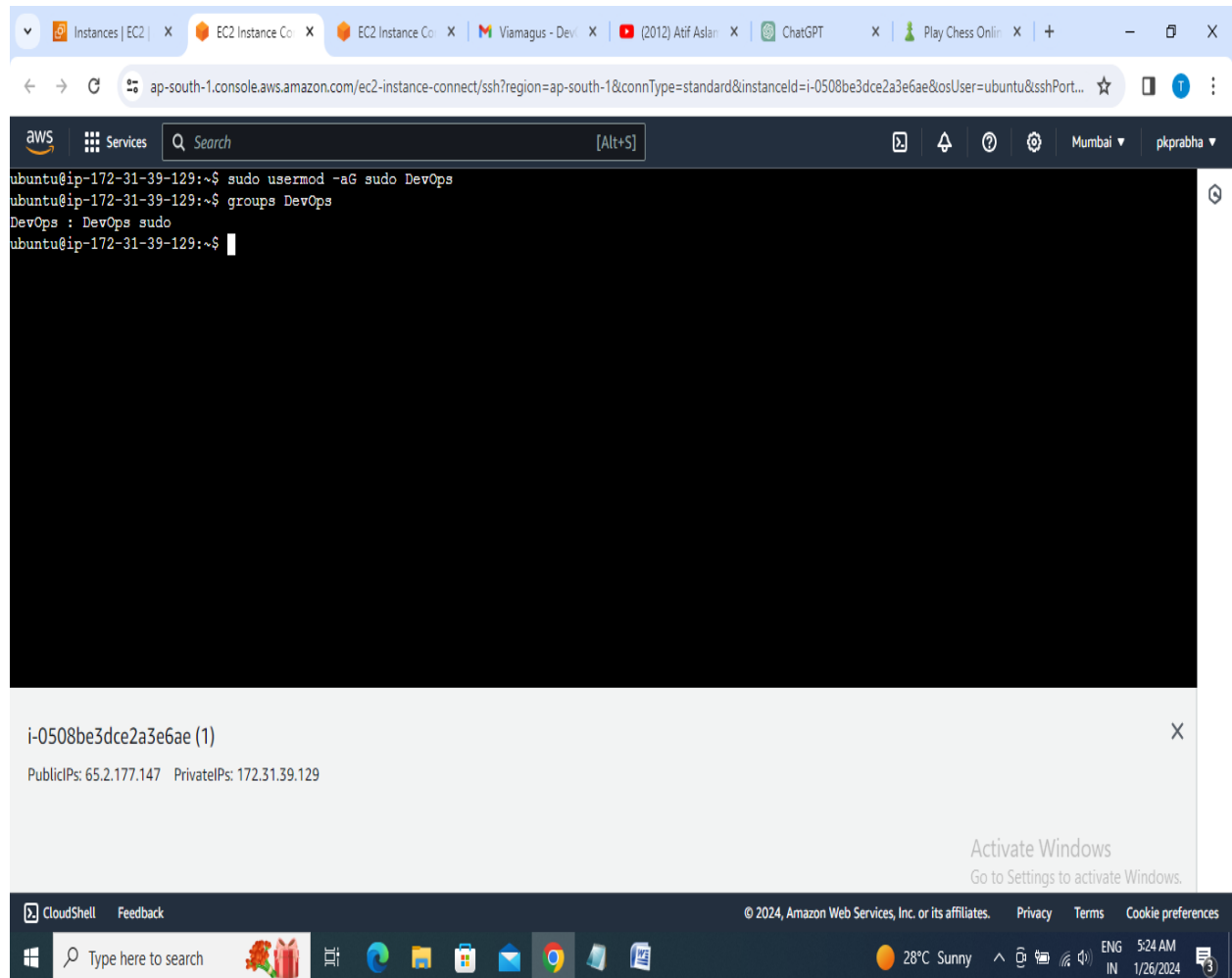
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Type here to search 28°C Sunny 5:16 AM 1/26/2024

Command:

sudo adduser DevOps --force-badname

CREATE A SUDO GROUP:



The screenshot shows a web browser window with multiple tabs, including 'Instances | EC2', 'EC2 Instance Co...', 'Viamagus - Dev...', '(2012) Atif Aslan...', 'ChatGPT', and 'Play Chess Onlin...'. The active tab is 'ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-0508be3dce2a3e6ae&osUser=ubuntu&sshPort...'. Below the browser window is a terminal window titled 'i-0508be3dce2a3e6ae (1)' with the following content:

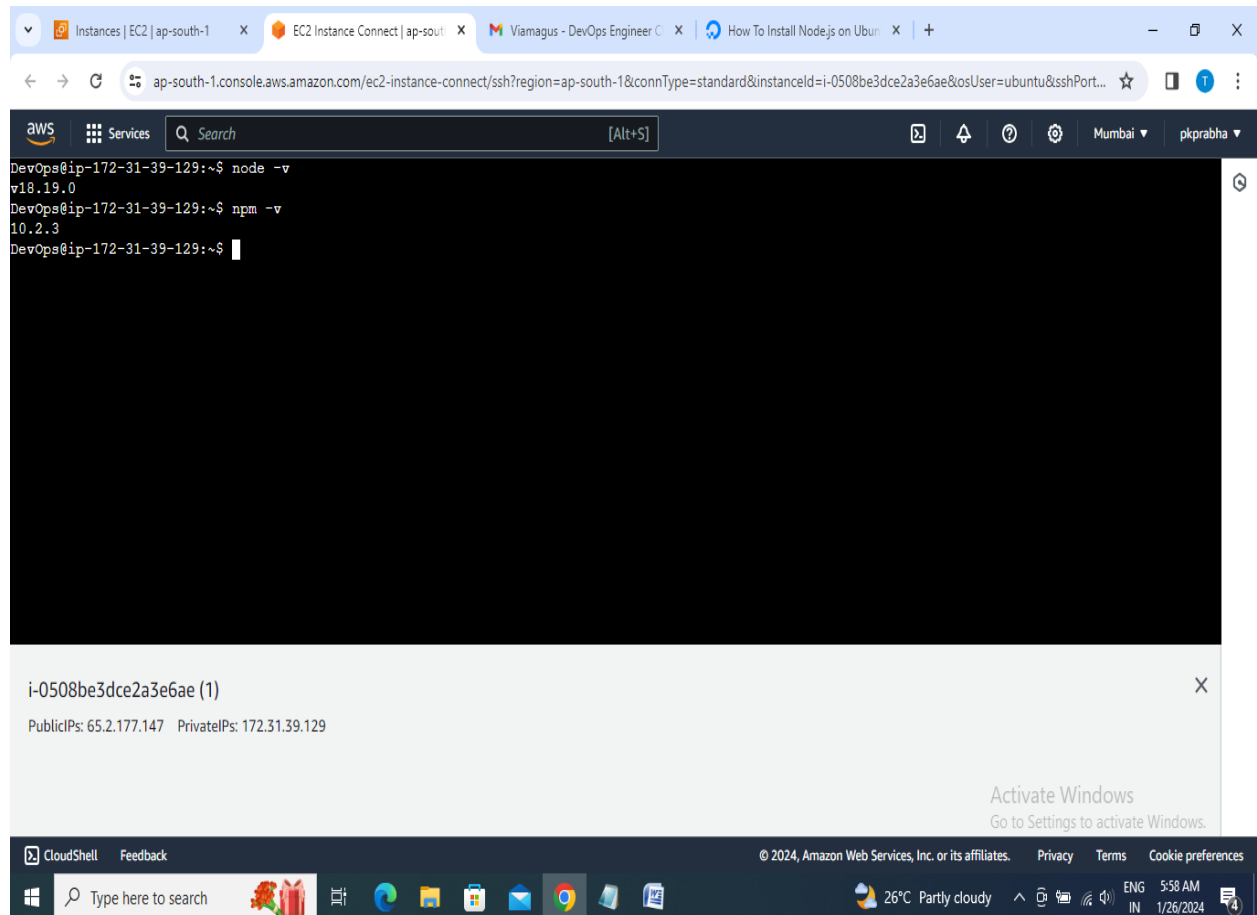
```
ubuntu@ip-172-31-39-129:~$ sudo usermod -aG sudo DevOps
ubuntu@ip-172-31-39-129:~$ groups DevOps
DevOps : DevOps sudo
ubuntu@ip-172-31-39-129:~$
```

The terminal window also displays the instance ID 'i-0508be3dce2a3e6ae (1)' and the public/private IP addresses: 'PublicIPs: 65.2.177.147 PrivateIPs: 172.31.39.129'. At the bottom of the terminal window, there is a Windows taskbar with the search bar, taskbar icons, and system tray information including the date and time '1/26/2024 5:24 AM'.

Command:

Sudo usermod -aG sudo DevOps

Installing Nodejs & npm:



The screenshot shows a web browser window with several tabs, including 'Instances | ap-south-1', 'EC2 Instance Connect | ap-south-1', 'Viamagus - DevOps Engineer C', and 'How To Install Nodejs on Ubuntu'. The active tab is 'ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-0508be3dce2a3e6ae&osUser=ubuntu&sshPort...'. Below the browser window is the AWS CloudShell interface. The terminal shows the following commands and output:

```
DevOps@ip-172-31-39-129:~$ node -v
v18.19.0
DevOps@ip-172-31-39-129:~$ npm -v
10.2.3
DevOps@ip-172-31-39-129:~$
```

Below the terminal output, there is a box for the instance ID 'i-0508be3dce2a3e6ae (1)' with public and private IP addresses: 'PublicIPs: 65.2.177.147 PrivateIPs: 172.31.39.129'. At the bottom of the CloudShell interface, there is a taskbar with various icons and a system tray showing the date and time as '1/26/2024 5:58 AM'.

Commands:

cd ~

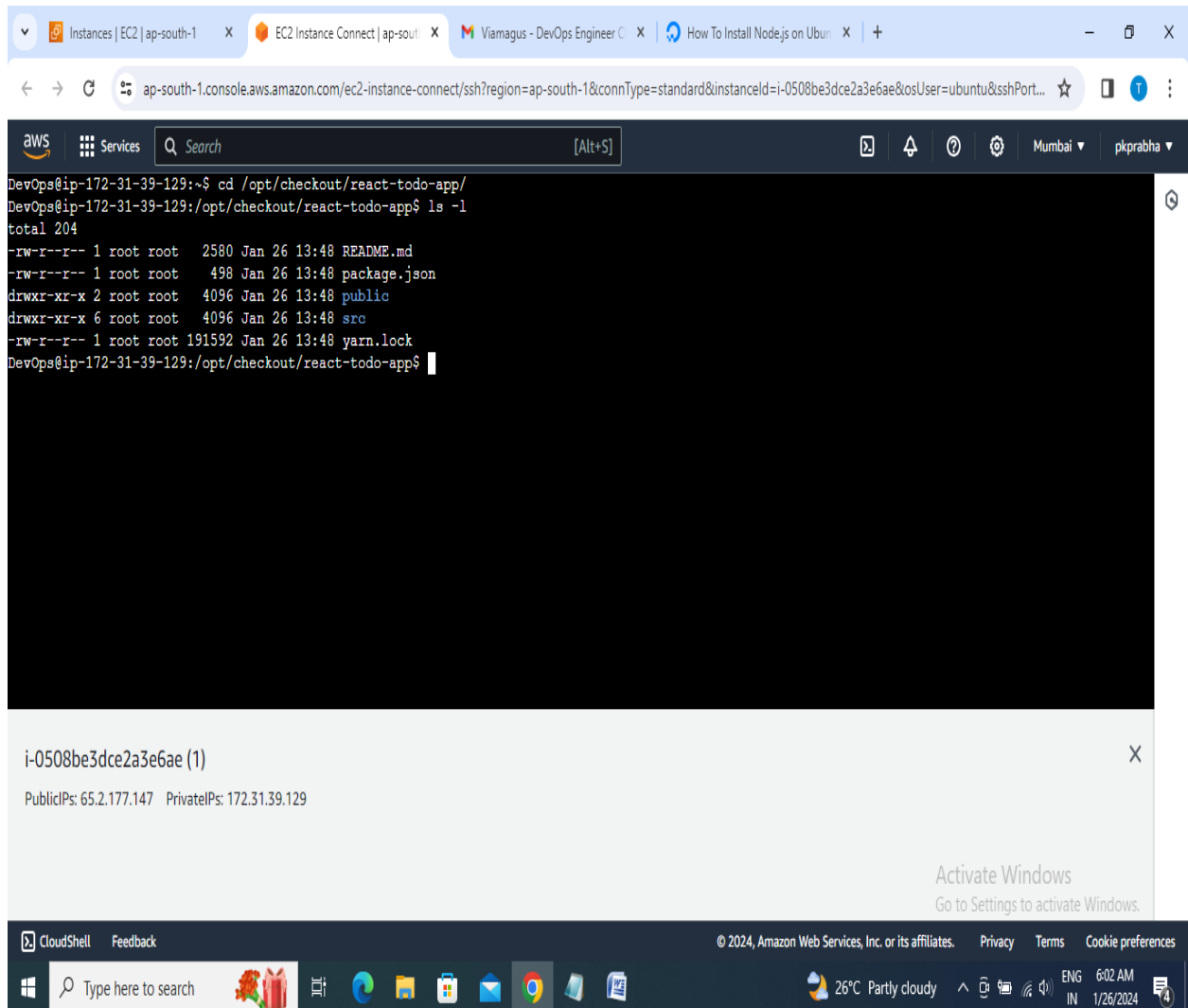
**curl -sL https://deb.nodesource.com/setup_18.x -o
[nodesource_setup.sh](#)**

sudo bash nodesource_setup.sh

sudo apt install nodejs

node -v

git clone:



The screenshot shows an AWS CloudShell terminal window. The terminal output is as follows:

```
DevOps@ip-172-31-39-129:~$ cd /opt/checkout/react-todo-app/
DevOps@ip-172-31-39-129:/opt/checkout/react-todo-app$ ls -l
total 204
-rw-r--r-- 1 root root 2580 Jan 26 13:48 README.md
-rw-r--r-- 1 root root 498 Jan 26 13:48 package.json
drwxr-xr-x 2 root root 4096 Jan 26 13:48 public
drwxr-xr-x 6 root root 4096 Jan 26 13:48 src
-rw-r--r-- 1 root root 191592 Jan 26 13:48 yarn.lock
DevOps@ip-172-31-39-129:/opt/checkout/react-todo-app$
```

Below the terminal window, a metadata box displays the instance ID: `i-0508be3dce2a3e6ae (1)`, along with Public and Private IP addresses. An "Activate Windows" watermark is visible in the bottom right corner of the terminal area.

Commands:

`mkdir -p /opt/checkout/`

`cd /opt/checkout/`

`git clone https://github.com/kabirbaidhya/react-todo-app`

react-todo-app folder will be cloned

Using npm install and npm run build:

```
DevOps@ip-172-31-39-129:/opt/checkout/react-todo-app$ sudo npm install
npm WARN deprecated chokidar@1.6.1: Chokidar 2 will break on node v14+. Upgrade to chokidar 3 with 15x less dependencies.
npm WARN deprecated querystring@0.2.0: The querystring API is considered Legacy. new code should use the URLSearchParams API instead.
npm WARN deprecated json3@3.3.2: Please use the native JSON object instead of JSON 3
npm WARN deprecated cryptiles@2.0.5: This version has been deprecated in accordance with the hapi support policy (hapi.im/support). Please upgrade
to the latest version to get the best features, bug fixes, and security patches. If you are unable to upgrade at this time, paid support is avail
able for older versions (hapi.im/commercial).
npm WARN deprecated domelementtype@1.3.0: update to domelementtype@1.3.1
npm WARN deprecated sntp@1.0.9: This module moved to @hapi/sntp. Please make sure to switch over as this distribution is no longer supported and
may contain bugs and critical security issues.
npm WARN deprecated boom@2.10.1: This version has been deprecated in accordance with the hapi support policy (hapi.im/support). Please upgrade to
the latest version to get the best features, bug fixes, and security patches. If you are unable to upgrade at this time, paid support is availab
le for older versions (hapi.im/commercial).
npm WARN deprecated uuid@3.0.1: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is
known to be problematic. See https://v8.dev/blog/math-random for details.
npm WARN deprecated flatten@1.0.2: flatten is deprecated in favor of utility frameworks such as lodash.
npm WARN deprecated clone@1.0.2: XSS vulnerability fixed in v1.0.3
npm WARN deprecated har-validator@2.0.6: this library is no longer supported
npm WARN deprecated hoek@2.16.3: This version has been deprecated in accordance with the hapi support policy (hapi.im/support). Please upgrade to
the latest version to get the best features, bug fixes, and security patches. If you are unable to upgrade at this time, paid support is availab
le for older versions (hapi.im/commercial).
npm WARN deprecated abab@1.0.3: Use your platform's native atob() and btoa() methods instead
npm WARN deprecated sane@1.4.1: some dependency vulnerabilities fixed, support for node < 10 dropped, and newer ECMAScript syntax/features added
npm WARN deprecated is-buffer@1.1.4: This version of 'is-buffer' is out-of-date. You must update to v1.1.6 or newer
```

i-0508be3dce2a3e6ae (1)
PublicIPs: 65.2.177.147 PrivateIPs: 172.31.39.129

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Type here to search

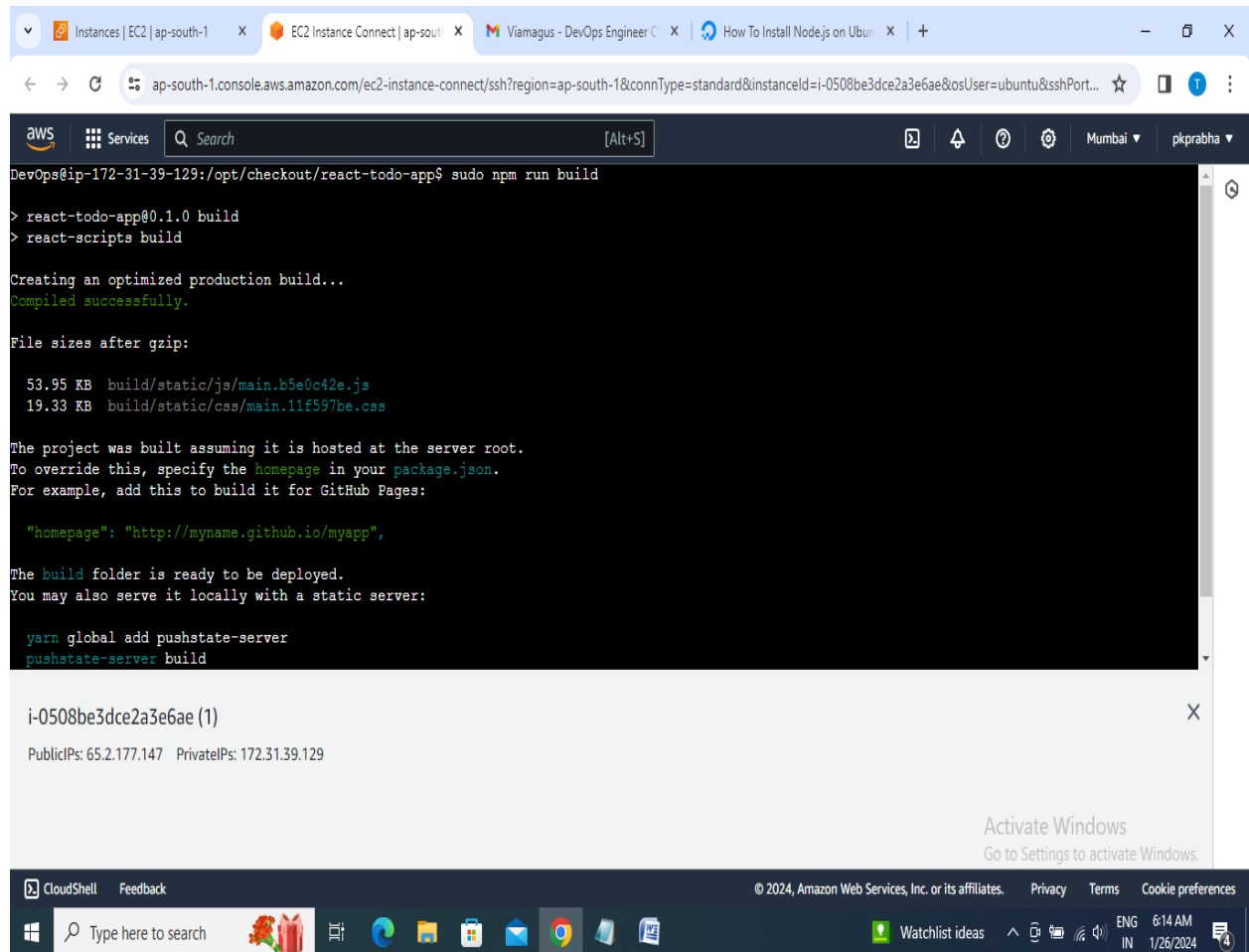
NGN/INR -1.39%

ENG 6:10 AM
IN 1/26/2024

Command:

Sudo npm install

Npm run build:



```
DevOps@ip-172-31-39-129:/opt/checkout/react-todo-app$ sudo npm run build
> react-todo-app@0.1.0 build
> react-scripts build

Creating an optimized production build...
Compiled successfully.

File sizes after gzip:

 53.95 KB  build/static/js/main.b5e0c42e.js
 19.33 KB  build/static/css/main.11f597be.css

The project was built assuming it is hosted at the server root.
To override this, specify the homepage in your package.json.
For example, add this to build it for GitHub Pages:

  "homepage": "http://myname.github.io/myapp",

The build folder is ready to be deployed.
You may also serve it locally with a static server:

 yarn global add pushstate-server
 pushstate-server build
```

i-0508be3dce2a3e6ae (1)

PublicIPs: 65.2.177.147 PrivateIPs: 172.31.39.129

Activate Windows
Go to Settings to activate Windows.

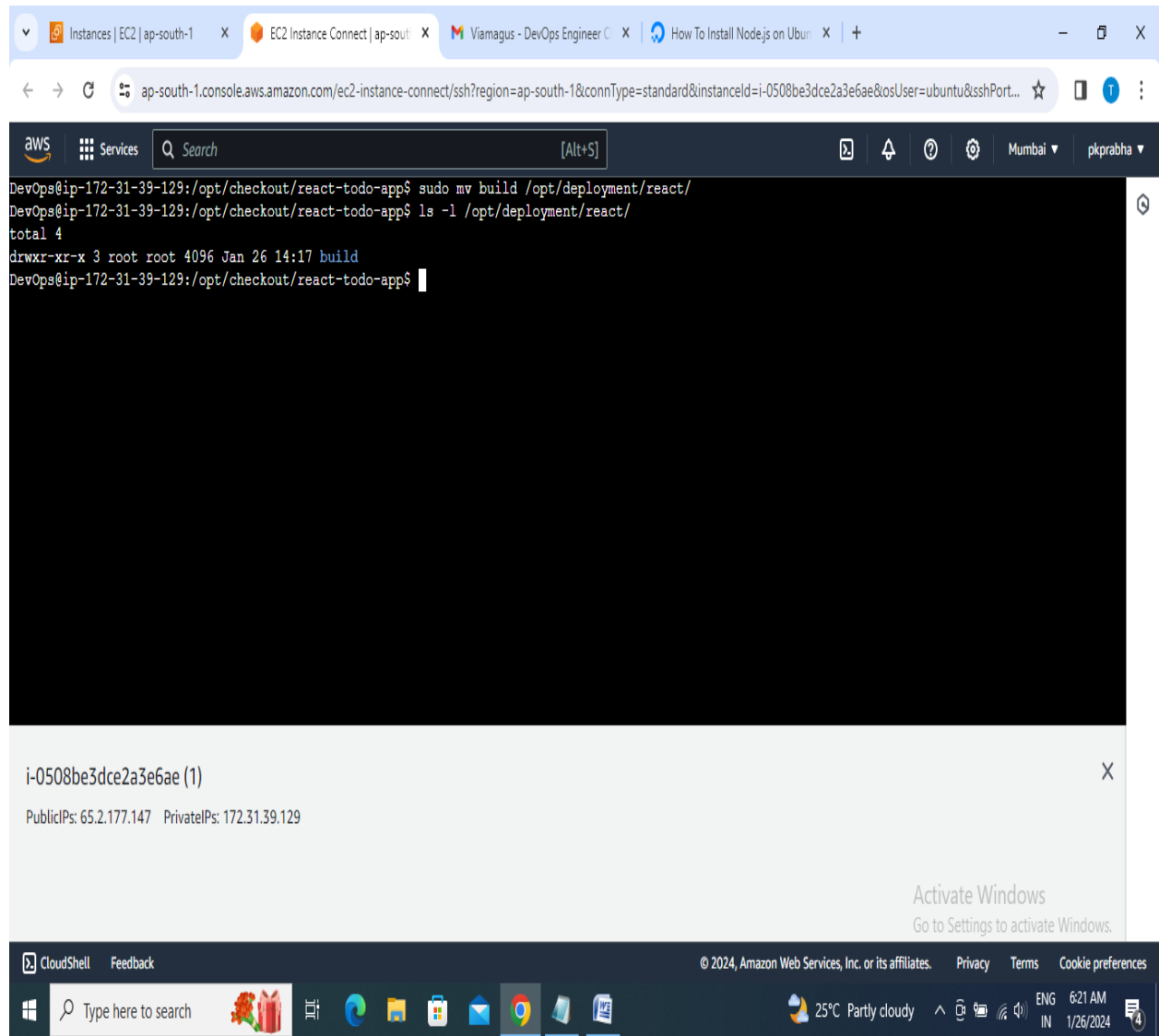
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Type here to search Watchlist ideas ENG IN 6:14 AM 1/26/2024

Command:

Sudo npm run build

MOVING THE BUILD FILE TO /OPT/DEPLOYMENT/REACT:



The screenshot shows a terminal window with the following commands and output:

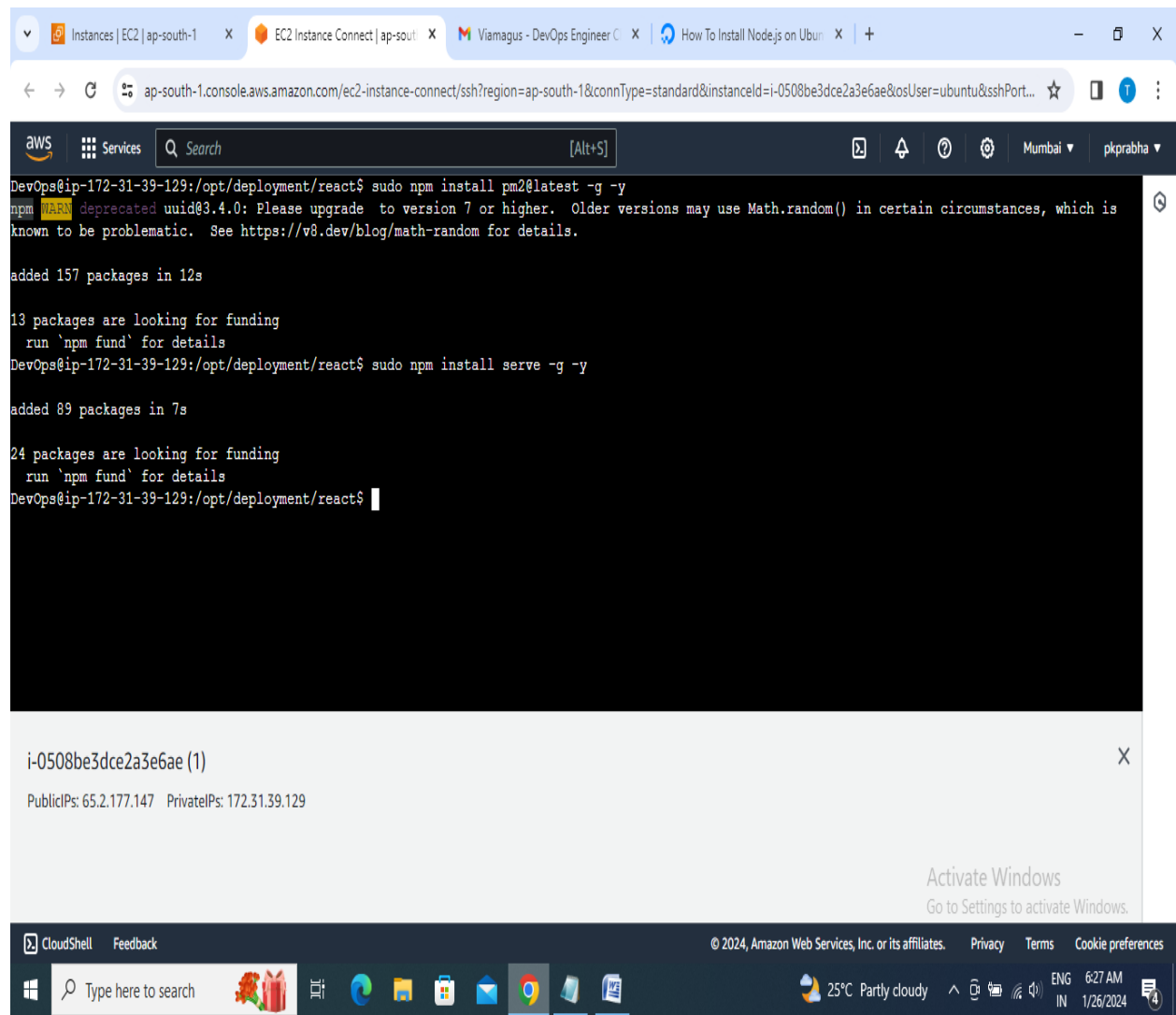
```
DevOps@ip-172-31-39-129:/opt/checkout/react-todo-app$ sudo mv build /opt/deployment/react/
DevOps@ip-172-31-39-129:/opt/checkout/react-todo-app$ ls -l /opt/deployment/react/
total 4
drwxr-xr-x 3 root root 4096 Jan 26 14:17 build
DevOps@ip-172-31-39-129:/opt/checkout/react-todo-app$
```

The terminal window is titled "i-0508be3dce2a3e6ae (1)" and shows the public and private IP addresses: "PublicIPs: 65.2.177.147 PrivateIPs: 172.31.39.129". The bottom of the window shows the "Activate Windows" watermark and the "CloudShell" logo.

Command:

`sudo mv /build /opt/deployment/react`

INSTALLING PM2 AND SERVE:



The screenshot shows a web browser with several tabs open, including 'Instances | EC2 | ap-south-1', 'EC2 Instance Connect | ap-south-1', 'Viamagus - DevOps Engineer C...', and 'How To Install Nodejs on Ubuntu'. The active tab is 'EC2 Instance Connect | ap-south-1', showing a terminal window for an EC2 instance. The terminal output is as follows:

```
DevOps@ip-172-31-39-129:/opt/deployment/react$ sudo npm install pm2@latest -g -y
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.

added 157 packages in 12s

13 packages are looking for funding
  run `npm fund` for details
DevOps@ip-172-31-39-129:/opt/deployment/react$ sudo npm install serve -g -y

added 89 packages in 7s

24 packages are looking for funding
  run `npm fund` for details
DevOps@ip-172-31-39-129:/opt/deployment/react$
```

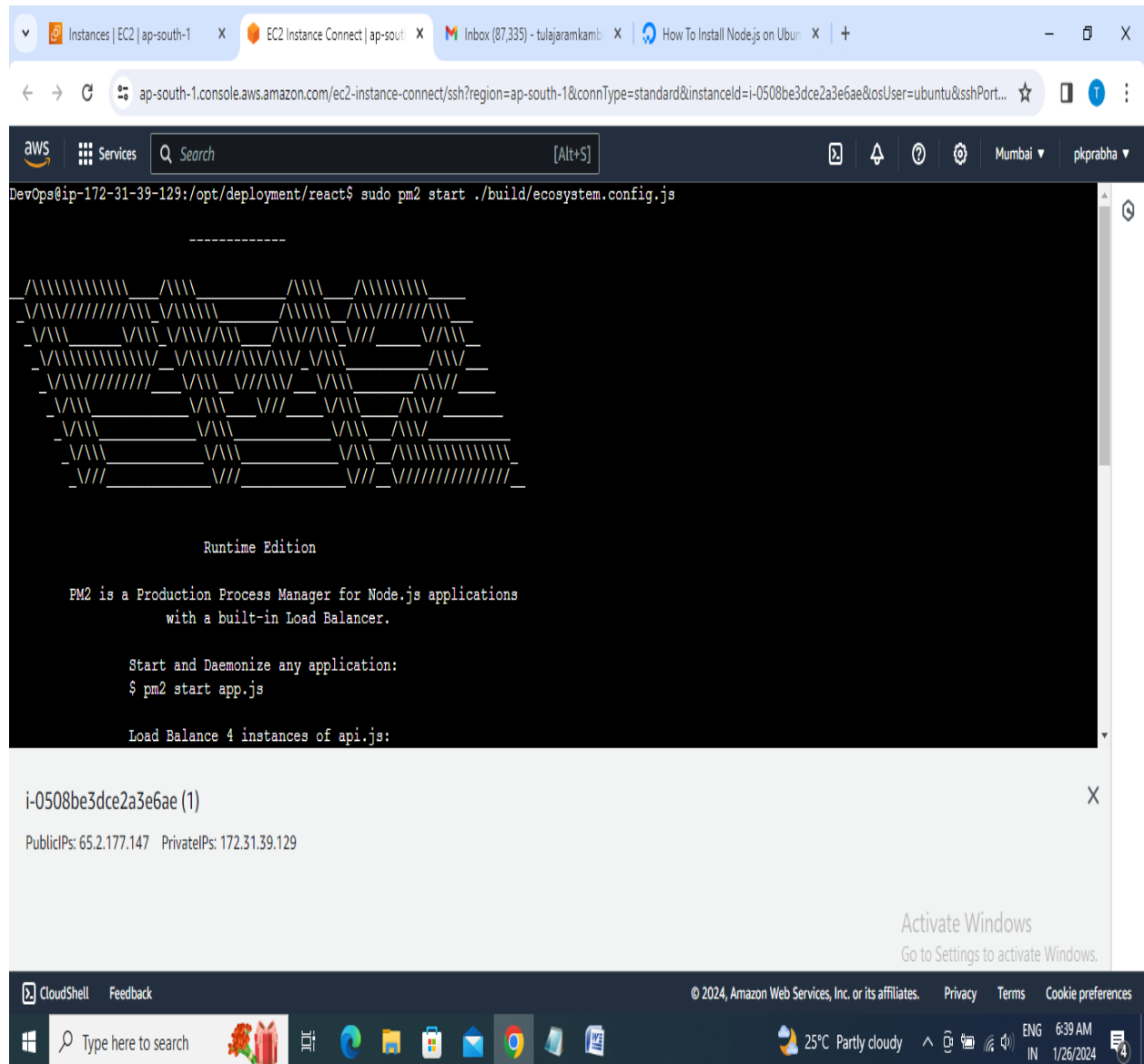
Below the terminal window, there is a summary box for the instance `i-0508be3dce2a3e6ae (1)` with Public IPs: 65.2.177.147 and Private IPs: 172.31.39.129. At the bottom of the terminal window, there is a Windows activation watermark: "Activate Windows Go to Settings to activate Windows."

Command:

sudo npm install pm2@latest -g -y

sudo npm install serve -g -y

DEPLOYMENT USING PM2 in /opt/deployment/react:



The screenshot shows an AWS CloudShell terminal window with the following content:

```
$ pm2 start api.js -i 4

Monitor in production:
$ pm2 monitor

Make pm2 auto-boot at server restart:
$ pm2 startup

To go further checkout:
http://pm2.io/

-----

[PM2] Spawning PM2 daemon with pm2_home=/root/.pm2
[PM2] PM2 Successfully daemonized
[PM2] [WARN] Applications react-todo-app not running, starting...
[PM2] App [react-todo-app] launched (1 instances)
```

id	name	namespace	version	mode	pid	uptime	U	status	cpu	mem	user	watching
0	react-todo-app	default	5.3.1	fork	2751	0s	0	online	0%	31.3mb	root	disabled

DevOps@ip-172-31-39-129:/opt/deployment/react\$

i-0508be3dce2a3e6ae (1)
PublicIPs: 65.2.177.147 PrivateIPs: 172.31.39.129

Activate Windows
Go to Settings to activate Windows.

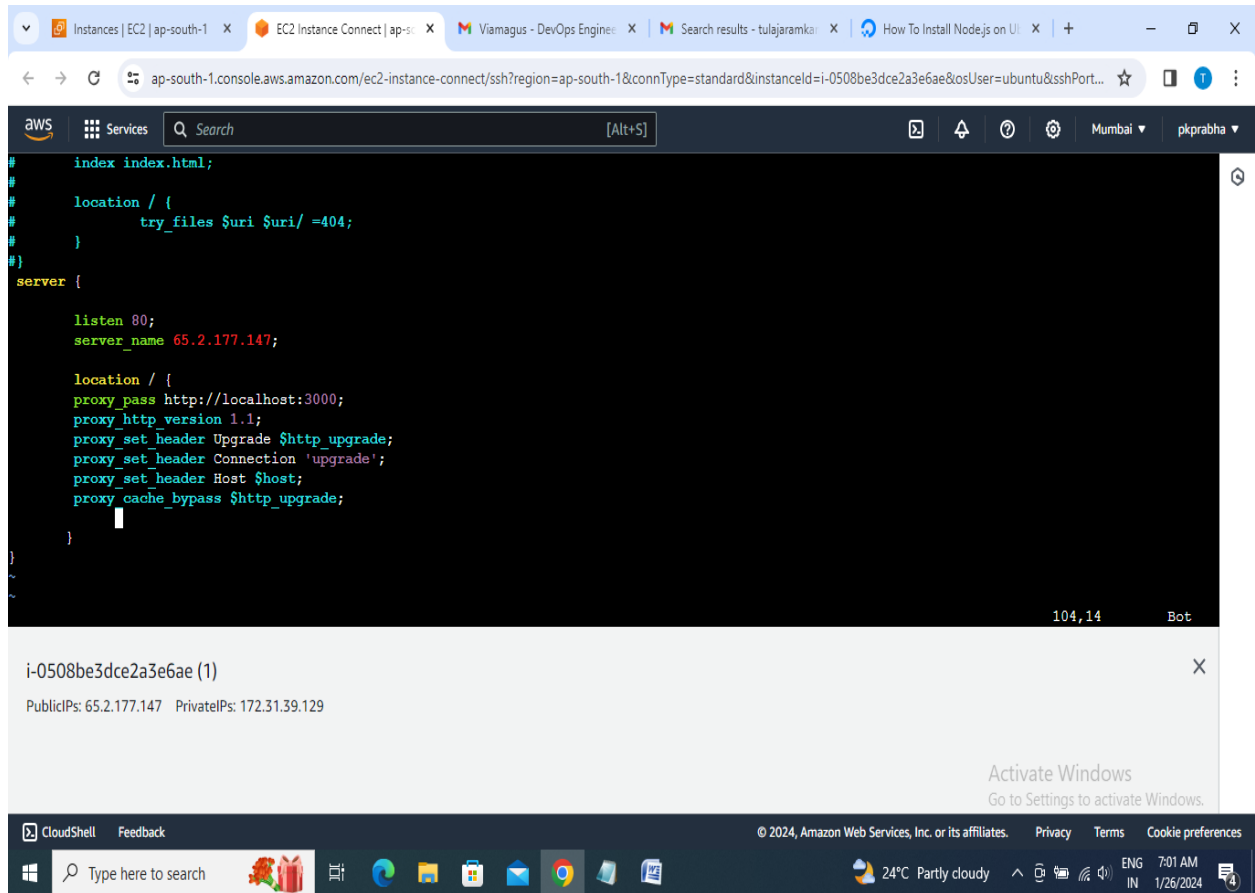
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
Type here to search 25°C Partly cloudy ENG IN 6:39 AM 1/26/2024

Command:

**Sudo cp /tmp/ecosystem.config.js
/opt/deployment/react/build**

Sudo pm2 start ./build/ecosystem.config.js

SET UP NGINX PROXY AND PROXY ALL REQUESTS TO REACT



The screenshot shows an AWS CloudShell terminal window with the following Nginx configuration:

```
# index index.html;
#
# location / {
#     try_files $uri $uri/ =404;
# }
server {
    listen 80;
    server_name 65.2.177.147;

    location / {
        proxy_pass http://localhost:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
    }
}
```

Below the terminal, a metadata box displays the instance ID `i-0508be3dce2a3e6ae (1)` and its public/private IP addresses. The bottom of the image shows the Windows taskbar with the system clock at 7:01 AM on 1/26/2024.

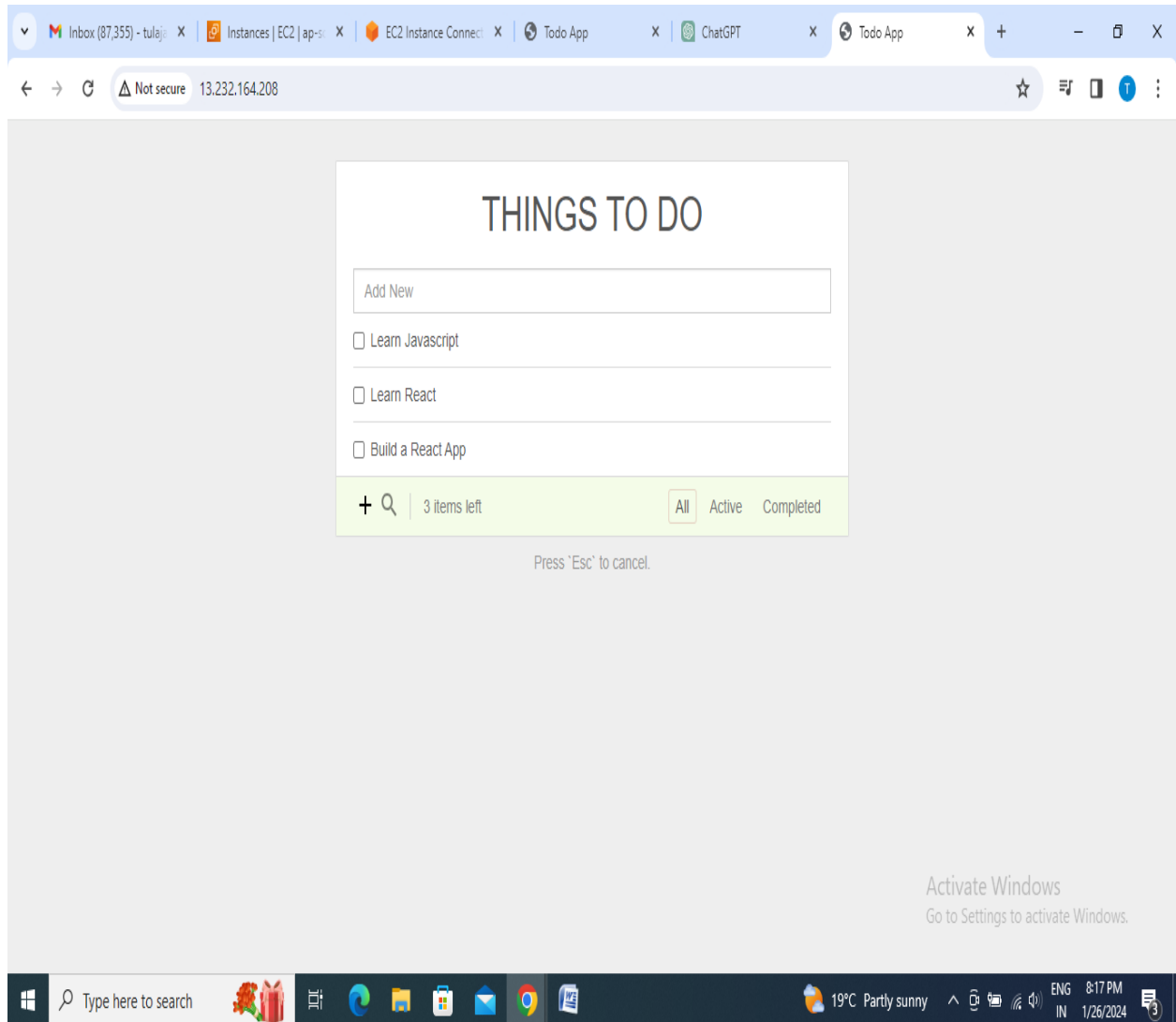
Setting up nginx proxy:

`Sudo vi /etc/nginx/sites-available/defaults`

Appending :

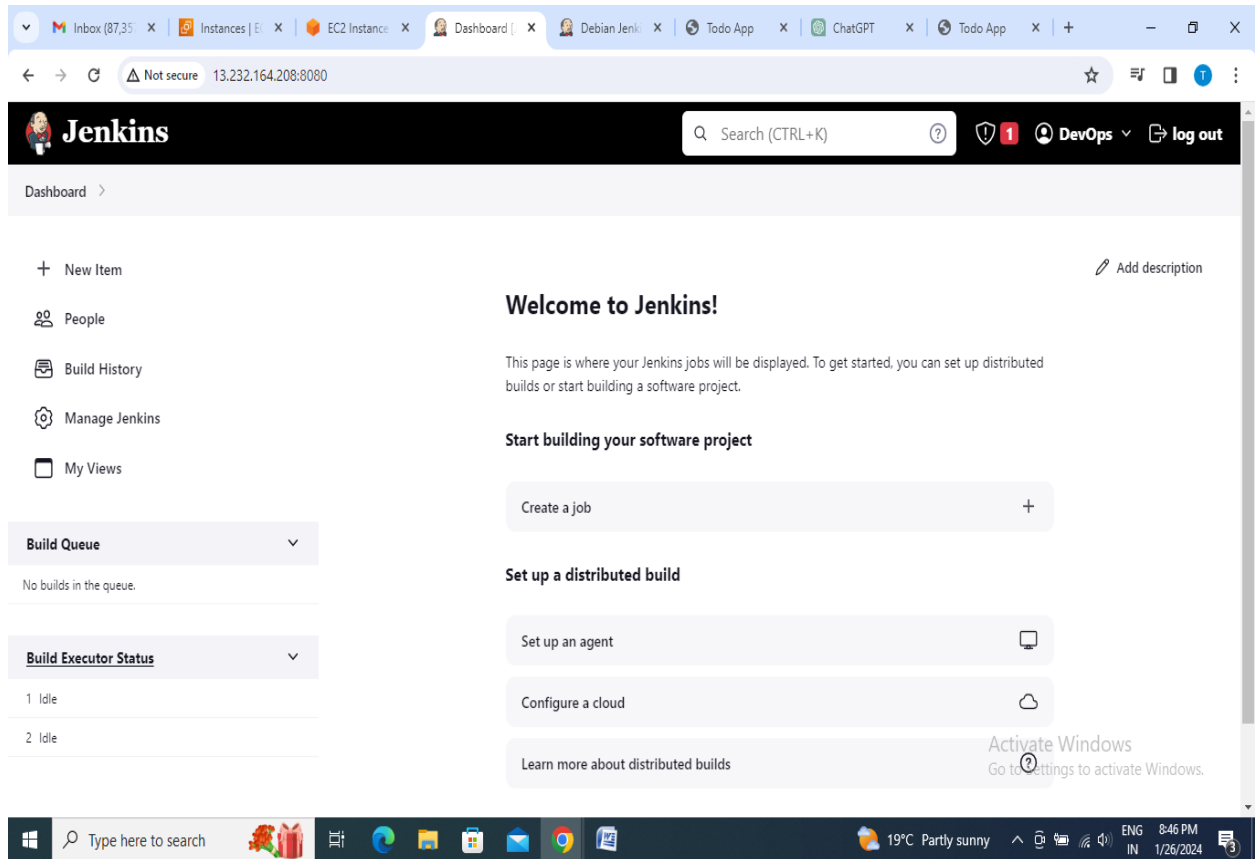
```
server {  
    listen 80;  
  
    server_name 13.233.130.65;  
  
    location / {  
        proxy_pass http://localhost:3000;  
        proxy_http_version 1.1;  
        proxy_set_header Upgrade $http_upgrade;  
        proxy_set_header Connection 'upgrade';  
        proxy_set_header Host $host;  
        proxy_cache_bypass $http_upgrade;  
  
    }  
}
```

OUTPUT



JENKINS PART

JENKINS INSTALLATION:

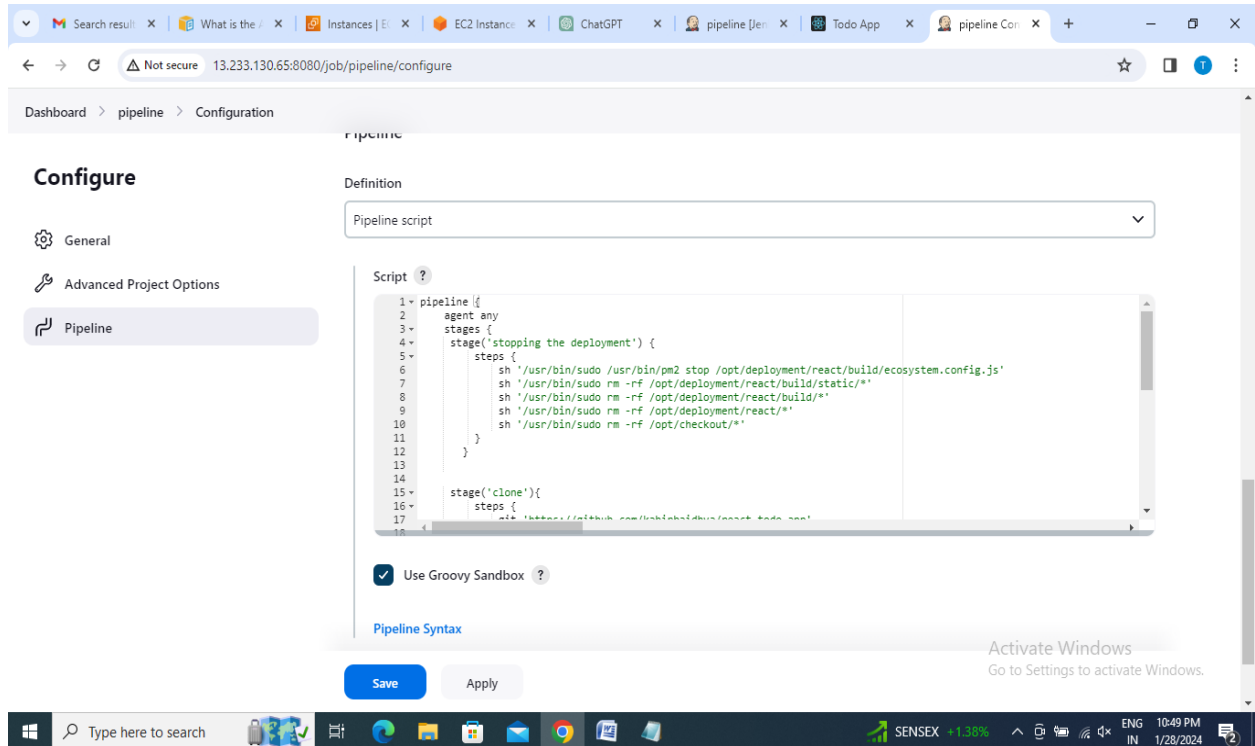


Commands:

```
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \  
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \  
https://pkg.jenkins.io/debian-stable binary/ | sudo tee \  
/etc/apt/sources.list.d/jenkins.list > /dev/null
```

```
sudo apt-get update  
sudo apt-get install fontconfig openjdk-17-jre  
sudo apt-get install Jenkins  
sudo service jenkins start      → PORT 8080
```

JENKINS PIPELINE:



Dashboard > pipeline > Configuration

Configure

Definition: Pipeline script

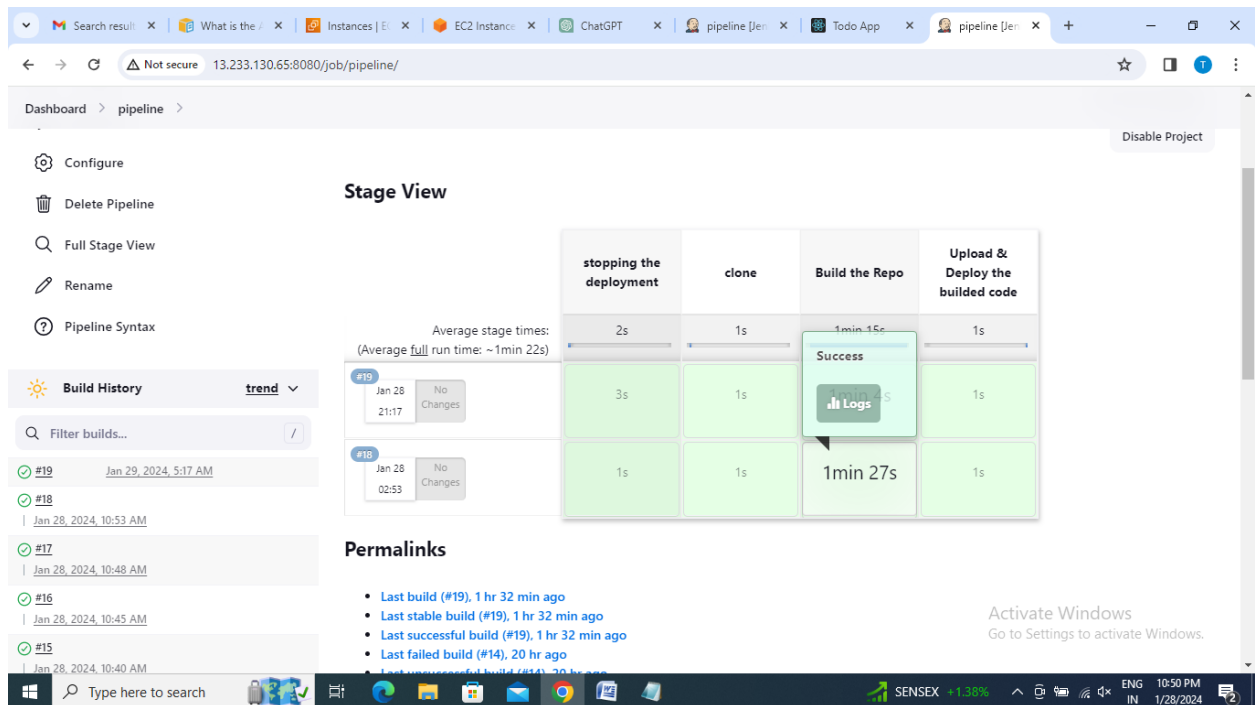
Script

```
1 pipeline {
2   agent any
3   stages {
4     stage('stopping the deployment') {
5       steps {
6         sh '/usr/bin/sudo /usr/bin/pm2 stop /opt/deployment/react/build/ecosystem.config.js'
7         sh '/usr/bin/sudo rm -rf /opt/deployment/react/build/static/*'
8         sh '/usr/bin/sudo rm -rf /opt/deployment/react/build/*'
9         sh '/usr/bin/sudo rm -rf /opt/deployment/react/*'
10        sh '/usr/bin/sudo rm -rf /opt/checkout/*'
11      }
12    }
13  }
14  stage('clone'){
15    steps {
16      ...
17    }
18  }
19 }
```

☒ Use Groovy Sandbox

Save Apply

BUILD OUTPUT:



Dashboard > pipeline

Stage View

Average stage times: (Average full run time: ~1min 22s)

	stopping the deployment	clone	Build the Repo	Upload & Deploy the build code
#19 Jan 28 21:17 No Changes	3s	1s	1min 15s	1s
#18 Jan 28 02:53 No Changes	1s	1s	1min 27s	1s

Build History

Filter builds...

- #19 Jan 29, 2024, 5:17 AM
- #18 Jan 28, 2024, 10:53 AM
- #17 Jan 28, 2024, 10:48 AM
- #16 Jan 28, 2024, 10:45 AM
- #15 Jan 28, 2024, 10:40 AM

Permalinks

- Last build (#19), 1 hr 32 min ago
- Last stable build (#19), 1 hr 32 min ago
- Last successful build (#19), 1 hr 32 min ago
- Last failed build (#14), 20 hr ago

```
pipeline {  
  agent any  
  stages {  
    stage('stopping the deployment') {  
      steps {  
        sh '/usr/bin/sudo /usr/bin/pm2 stop  
/opt/deployment/react/build/ecosystem.config.js'  
        sh '/usr/bin/sudo rm -rf  
/opt/deployment/react/build/static/*'  
        sh '/usr/bin/sudo rm -rf /opt/deployment/react/build/*'  
        sh '/usr/bin/sudo rm -rf /opt/deployment/react/*'  
        sh '/usr/bin/sudo rm -rf /opt/checkout/*'  
      }  
    }  
  }  
}
```

```
stage('clone'){  
  steps {  
    git 'https://github.com/kabirbaidhya/react-todo-app'
```



```
    sh 'pwd'

    sh '/usr/bin/sudo mv * /opt/checkout'

  }

}

stage('Build the Repo') {

  steps {

    sh 'cd /opt/checkout && /usr/bin/sudo npm install && /usr/bin/sudo npm run build'

    sh '/usr/bin/sudo cp -v /ecosystem/ecosystem.config.js /opt/checkout/build'

  }

}

stage('Upload & Deploy the builded code') {

  steps {

    sh '/usr/bin/sudo mv /opt/checkout/build /opt/deployment/react'

    sh '/usr/bin/sudo ln -s /opt/deployment/react/build/* /var/lib/jenkins/workspace/pipeline/'

  }

}
```

```
s3Upload consoleLogLevel: 'INFO',
dontSetBuildResultOnFailure: false,
dontWaitForConcurrentBuildCompletion: false, entries:
[[bucket: 'react-todo-app-1', excludedFile: '', flatten: false,
gzipFiles: false, keepForever: false, managedArtifacts: false,
noUploadOnFailure: false, selectedRegion: 'ap-south-1',
showDirectlyInBrowser: false, sourceFile: '**/build/*',
storageClass: 'STANDARD', uploadFromSlave: false,
useServerSideEncryption: false]], pluginFailureResultConstraint:
'FAILURE', profileName: 'react-todo-app', userMetadata: []
```

```
sh '/usr/bin/sudo rm -rf
/var/lib/jenkins/workspace/build'
```

```
sh '/usr/bin/sudo /usr/bin/pm2 start
/opt/deployment/react/build/ecosystem.config.js'
```

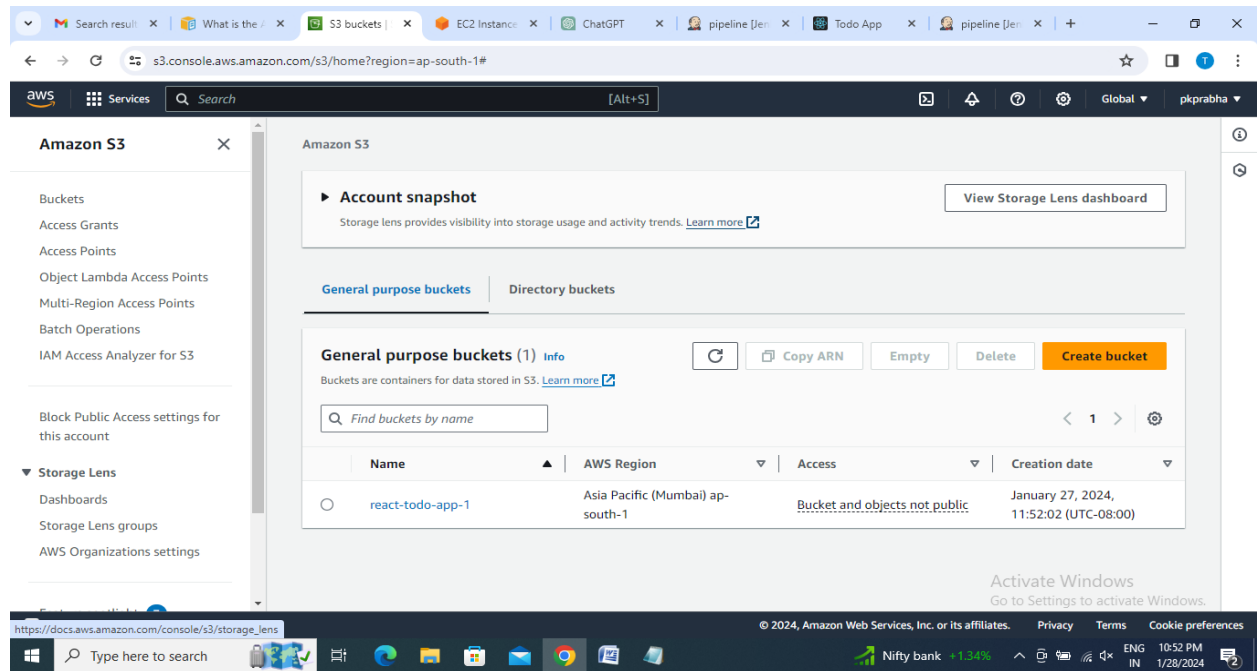
```
}
```

```
}
```

```
}
```

```
}
```

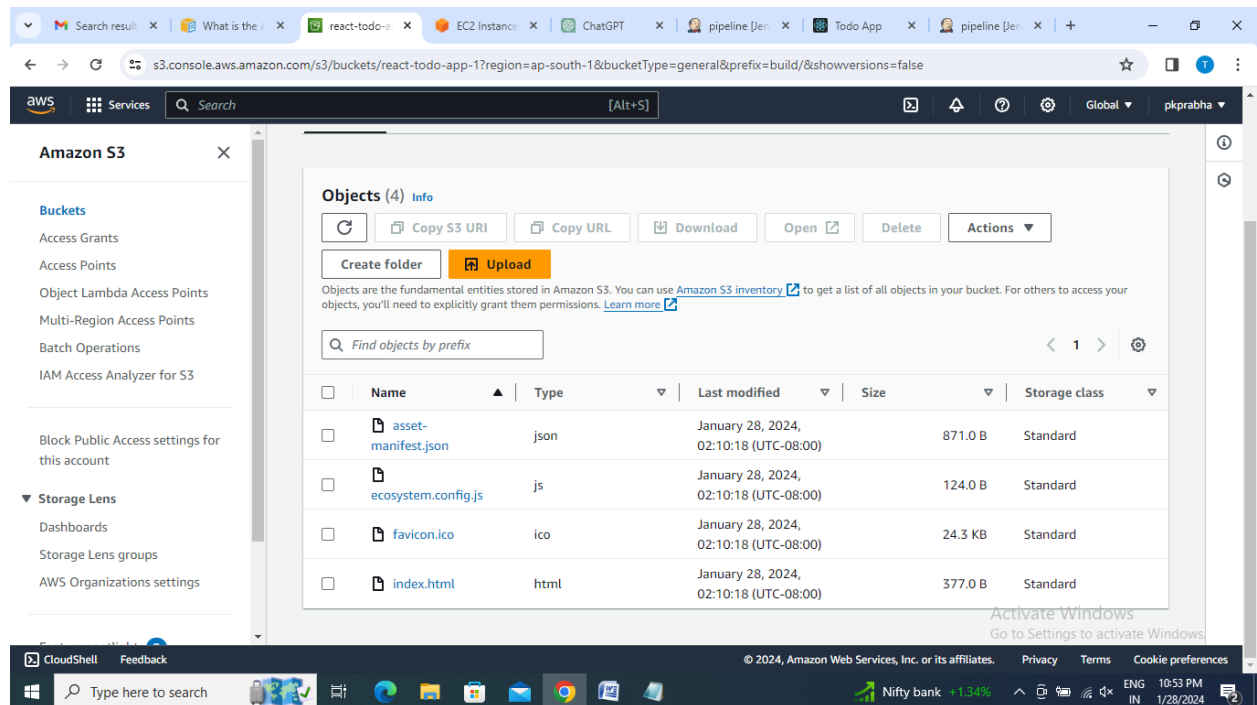
S3 BUCKET & ITS OUTPUT:



The screenshot shows the Amazon S3 console interface. On the left, there's a navigation menu with options like Buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Storage Lens, Dashboards, Storage Lens groups, and AWS Organizations settings. The main content area displays the 'Account snapshot' and 'General purpose buckets' section. A table lists the bucket 'react-todo-app-1' with its details.

Name	AWS Region	Access	Creation date
react-todo-app-1	Asia Pacific (Mumbai) ap-south-1	Bucket and objects not public	January 27, 2024, 11:52:02 (UTC-08:00)

OBJECTS IN S3 BUCKET:



The screenshot shows the Amazon S3 console interface for a specific bucket. The main content area displays the 'Objects (4)' section. A table lists the objects in the bucket with their details.

Name	Type	Last modified	Size	Storage class
asset-manifest.json	json	January 28, 2024, 02:10:18 (UTC-08:00)	871.0 B	Standard
ecosystem.config.js	js	January 28, 2024, 02:10:18 (UTC-08:00)	124.0 B	Standard
favicon.ico	ico	January 28, 2024, 02:10:18 (UTC-08:00)	24.3 KB	Standard
index.html	html	January 28, 2024, 02:10:18 (UTC-08:00)	377.0 B	Standard

5. BASH SCRIPT && 6. DOCKER FILE

```
#!/bin/bash
```

```
mkdir -p /home/ubuntu/JDK
```

```
cd /home/ubuntu/JDK
```

```
echo Downloading JDK file `date`
```

```
wget https://builds.openlogic.com/downloadJDK/openlogic-  
openjdk/17.0.8+7/openlogic-openjdk-
```

```
17.0.8+7-linux-x64.tar.gz
```

```
echo Installing JDK file `date`
```

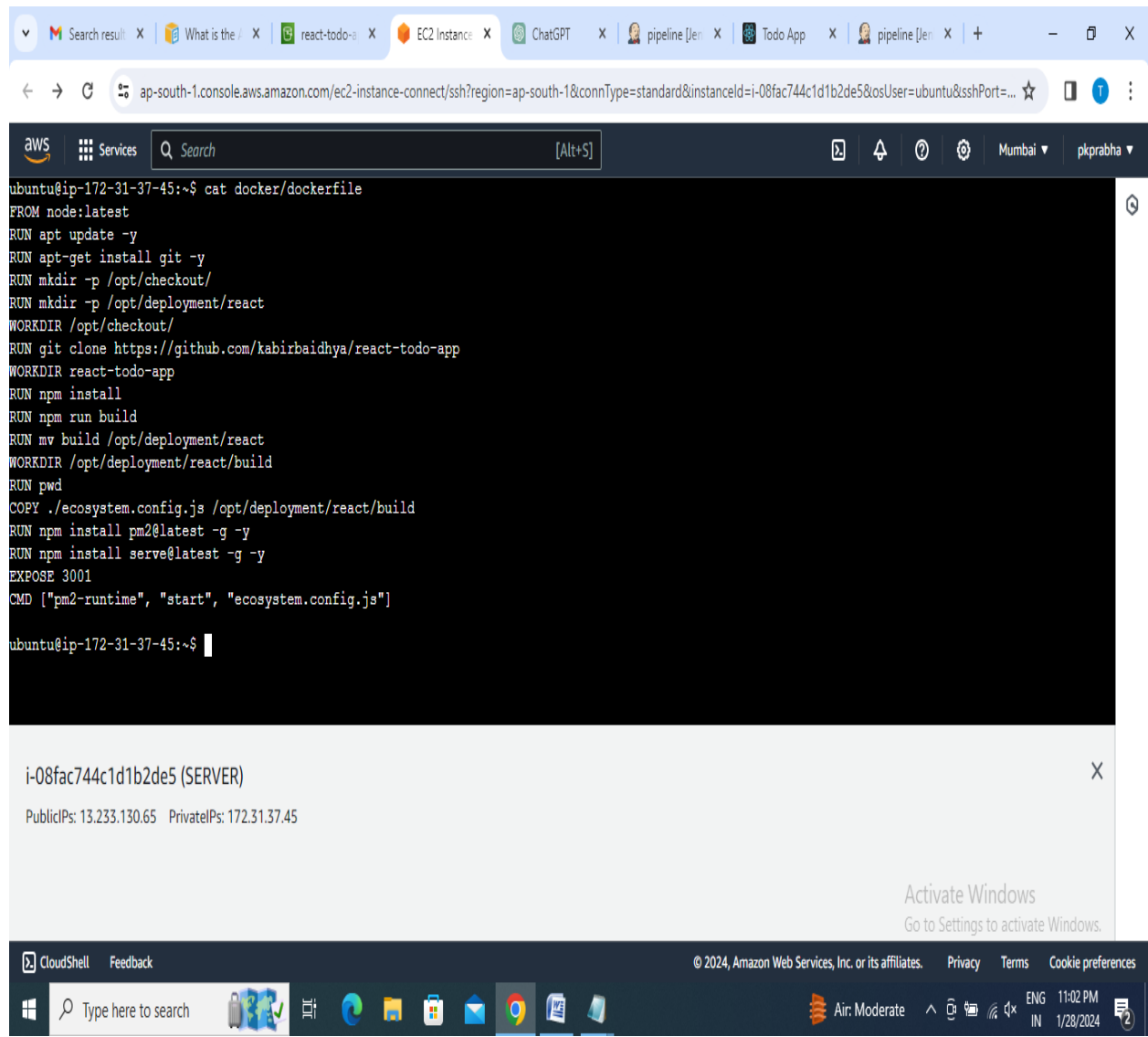
```
tar -xvf openlogic-openjdk-17.0.8+7-linux-x64.tar.gz
```

```
echo Environmental path for JDK , DevOps `date`
```

```
export PATH=$PATH:/usr/bin/java >> /home/DevOps/.bashrc
```

6. DOCKER PART:

Docker-file creation



The screenshot shows a web browser with multiple tabs open, including 'Search result', 'What is the', 'react-todo-app', 'EC2 Instance', 'ChatGPT', 'pipeline (Jen)', 'Todo App', and 'pipeline (Jen)'. The active tab is 'EC2 Instance', displaying the AWS console URL for an EC2 instance. Below the browser, the AWS CloudShell interface is visible, showing a terminal window with the following commands and output:

```
ubuntu@ip-172-31-37-45:~$ cat docker/dockerfile
FROM node:latest
RUN apt update -y
RUN apt-get install git -y
RUN mkdir -p /opt/checkout/
RUN mkdir -p /opt/deployment/react
WORKDIR /opt/checkout/
RUN git clone https://github.com/kabirbaidhya/react-todo-app
WORKDIR react-todo-app
RUN npm install
RUN npm run build
RUN mv build /opt/deployment/react
WORKDIR /opt/deployment/react/build
RUN pwd
COPY ./ecosystem.config.js /opt/deployment/react/build
RUN npm install pm2@latest -g -y
RUN npm install serve@latest -g -y
EXPOSE 3001
CMD ["pm2-runtime", "start", "ecosystem.config.js"]

ubuntu@ip-172-31-37-45:~$
```

Below the terminal, a box displays the instance ID 'i-08fac744c1d1b2de5 (SERVER)' and its IP addresses: 'PublicIPs: 13.233.130.65' and 'PrivateIPs: 172.31.37.45'. An 'Activate Windows' watermark is visible in the bottom right corner of the terminal area.

The bottom of the screenshot shows the AWS CloudShell footer with the text '© 2024, Amazon Web Services, Inc. or its affiliates.' and links for 'Privacy', 'Terms', and 'Cookie preferences'. The taskbar at the very bottom includes the Windows logo, a search bar, and various application icons.

Docker-compose creation

The screenshot shows a web browser window with several tabs open, including 'Search results', 'What is the...', 'react-todo-...', 'EC2 Instance...', 'ChatGPT', 'pipeline (Jen...', 'Todo App', and 'pipeline (Jen...'. The active tab is 'EC2 Instance...', which displays the AWS console URL: `ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-08fac744c1d1b2de5&osUser=ubuntu&sshPort=...`.

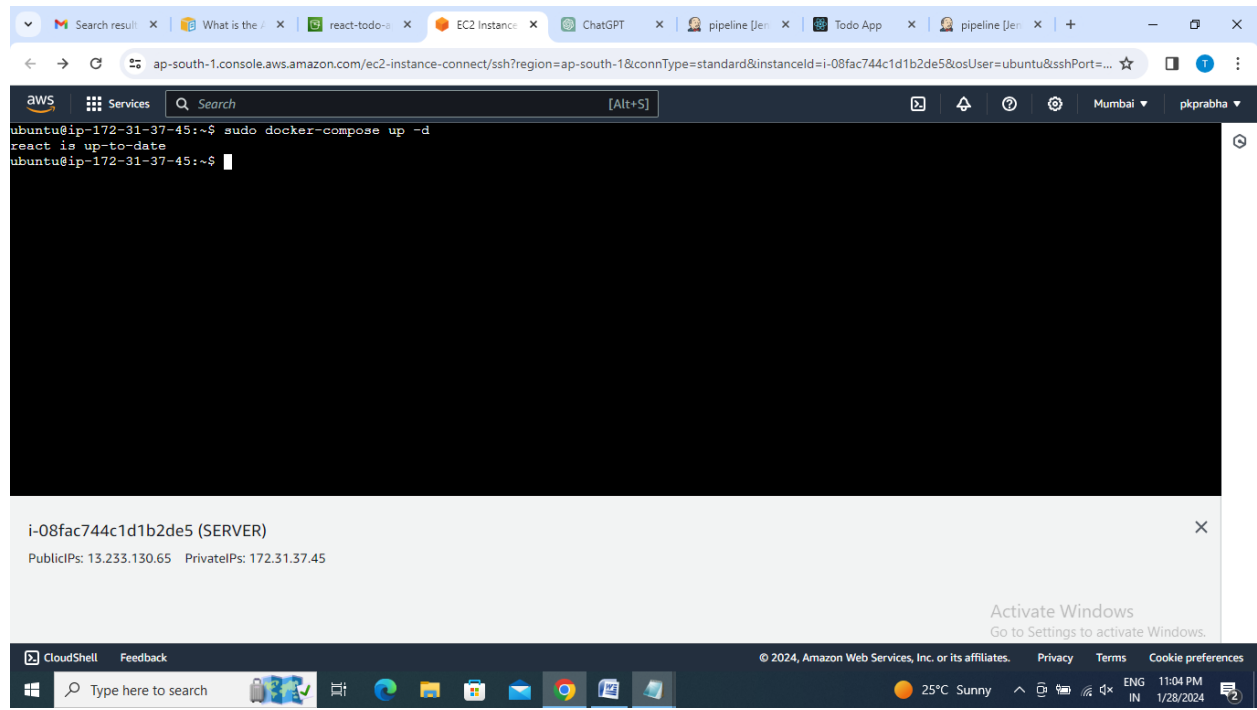
Below the browser window is the AWS CloudShell interface. The terminal shows the following commands and output:

```
ubuntu@ip-172-31-37-45:~$ cat docker-compose.yaml
version: "3"
services:
  react-todo-app:
    build:
      context: /home/ubuntu/docker
      container_name: react
    image: react-todo
    ports:
      - "3001:3001"
ubuntu@ip-172-31-37-45:~$
```

The CloudShell interface also displays the instance ID `i-08fac744c1d1b2de5 (SERVER)` and its public and private IP addresses: `PublicIPs: 13.233.130.65 PrivateIPs: 172.31.37.45`.

At the bottom of the screen, there is a Windows taskbar with the search bar, task view button, and several application icons (File Explorer, Mail, Chrome, etc.). The system tray shows the date and time as `11:02 PM 1/28/2024` and the language as `ENG IN`.

OUTPUT OF DOCKER-COMPOSE FILE:



The screenshot shows a web browser window with multiple tabs, including 'Search results', 'What is the', 'react-todo', 'EC2 Instance', 'ChatGPT', 'pipeline', 'Todo App', and 'pipeline'. The active tab is 'EC2 Instance', which displays the AWS console URL: `ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-08fac744c1d1b2de5&osUser=ubuntu&sshPort=...`. Below the browser window is a terminal window titled 'i-08fac744c1d1b2de5 (SERVER)' with public IP 13.233.130.65 and private IP 172.31.37.45. The terminal shows the following commands and output:

```
ubuntu@ip-172-31-37-45:~$ sudo docker-compose up -d
react is up-to-date
ubuntu@ip-172-31-37-45:~$
```

The terminal window also displays the instance ID 'i-08fac744c1d1b2de5 (SERVER)' and the public/private IP addresses. At the bottom of the terminal window, there is a status bar with the text '© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences' and a system tray showing '25°C Sunny', '11:04 PM', and '1/28/2024'.

Command:

Sudo docker-compose up -d

DOCKER FILE SCRIPT:

FROM node:latest

RUN apt update -y

RUN apt-get install git -y

RUN mkdir -p /opt/checkout/

RUN mkdir -p /opt/deployment/react

WORKDIR /opt/checkout/

RUN git clone https://github.com/kabirbaidhya/react-todo-app

WORKDIR react-todo-app

RUN npm install

RUN npm run build

RUN mv build /opt/deployment/react

WORKDIR /opt/deployment/react/build

RUN pwd

COPY ./ecosystem.config.js /opt/deployment/react/build

RUN npm install pm2@latest -g -y

RUN npm install serve@latest -g -y

EXPOSE 3001

CMD ["pm2-runtime", "start", "ecosystem.config.js"]

DOCKER-COMPOSE FILE:

version: "3"

services:

react-todo-app:

build:

context: /home/ubuntu/docker

container_name: react

image: react-todo

ports:

- "3001:3001"

THE-END

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXX

