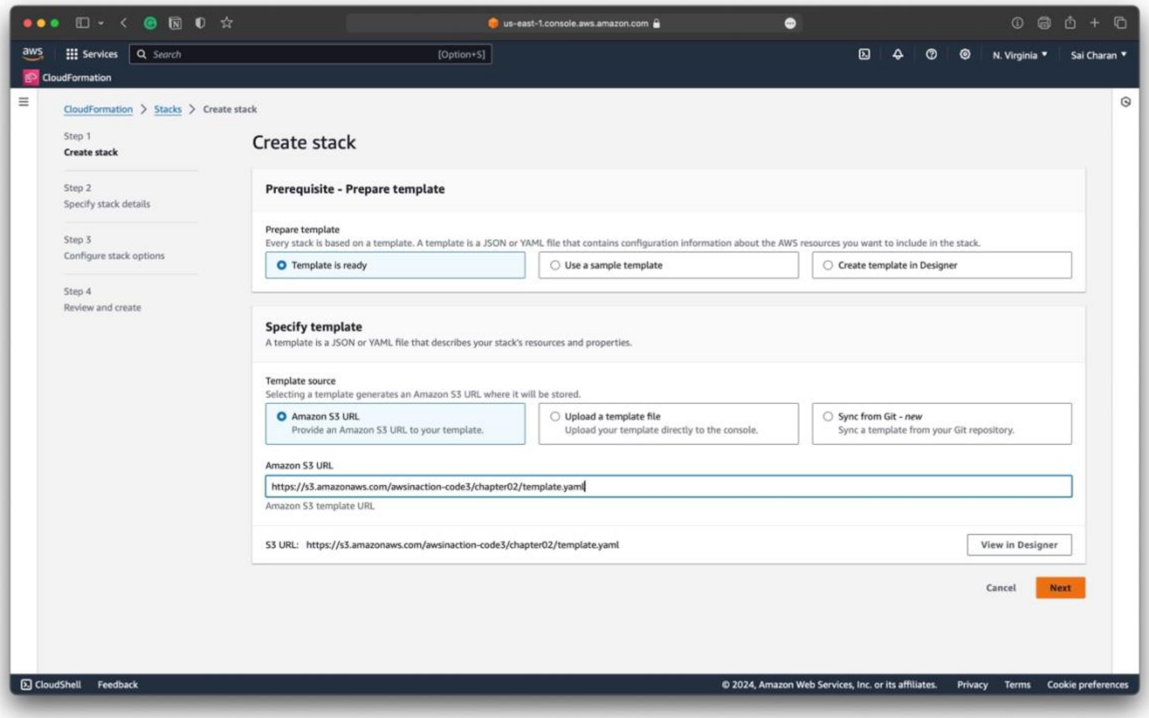
Creating Infrastructure:



CloudFormation > Stacks > Create stack

Step 1
Create stack

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review and create

Create stack

Prerequisite - Prepare template

Prepare template
Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

● Template is ready ○ Use a sample template ○ Create template in Designer

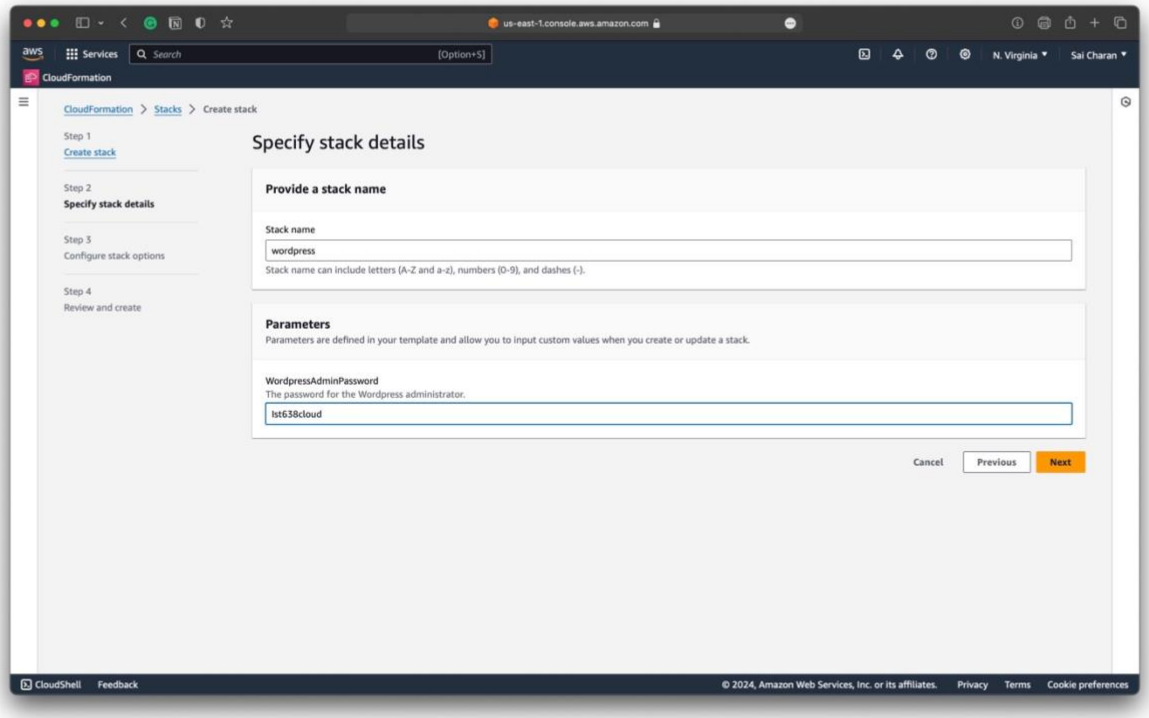
Specify template
A template is a JSON or YAML file that describes your stack's resources and properties.

Template source
Selecting a template generates an Amazon S3 URL where it will be stored.

● Amazon S3 URL — Provide an Amazon S3 URL to your template.
○ Upload a template file — Upload your template directly to the console.
○ Sync from Git - *new* — Sync a template from your Git repository.

Amazon S3 URL
https://s3.amazonaws.com/awsinaction-code3/chapter02/template.yaml
Amazon S3 template URL

S3 URL: https://s3.amazonaws.com/awsinaction-code3/chapter02/template.yaml



CloudFormation > Stacks > Create stack

Step 1
Create stack

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review and create

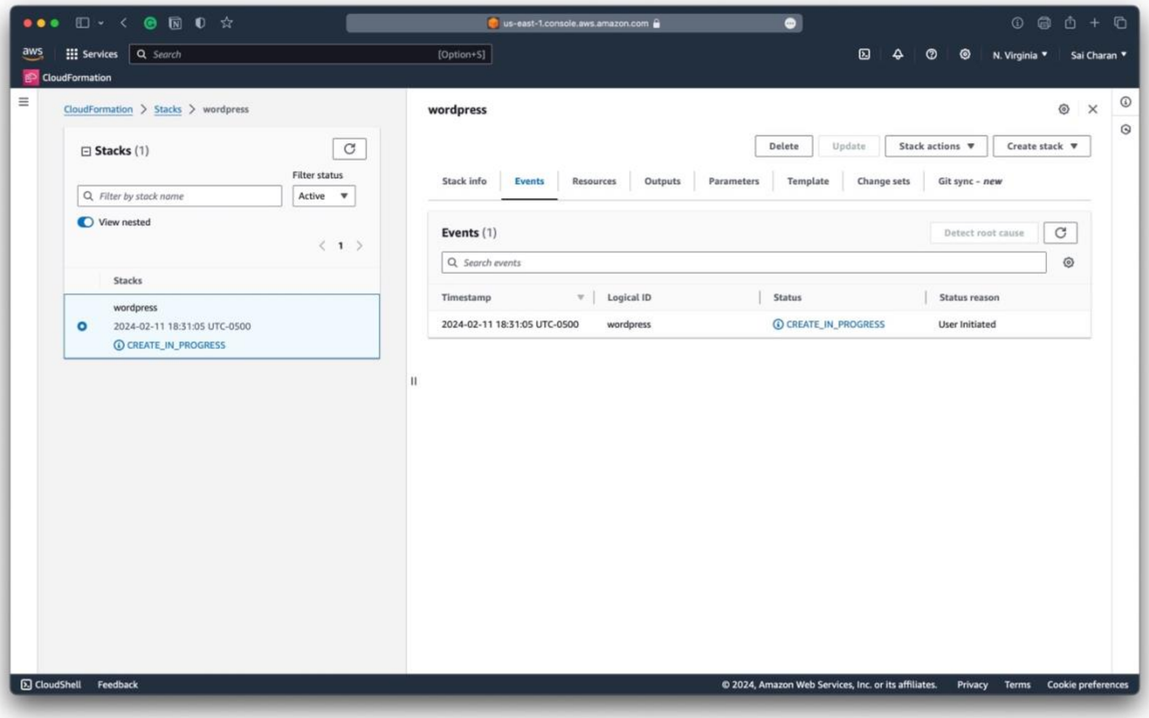
Specify stack details

Provide a stack name

Stack name
wordpress
Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters
Parameters are defined in your template and allow you to input custom values when you create or update a stack.

WordpressAdminPassword
The password for the Wordpress administrator.
Ist638cloud

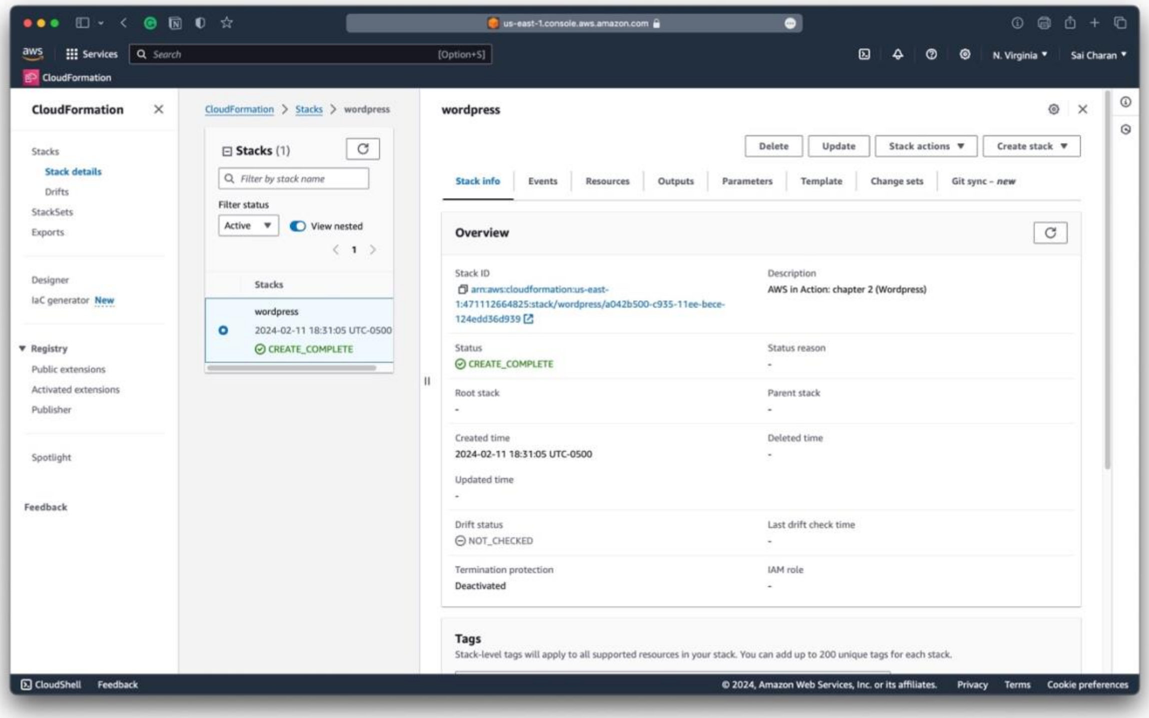


Stacks

wordpress
2024-02-11 18:31:05 UTC-0500
CREATE_IN_PROGRESS

Events (1)

Timestamp	Logical ID	Status	Status reason
2024-02-11 18:31:05 UTC-0500	wordpress	CREATE_IN_PROGRESS	User Initiated



Stacks

wordpress
2024-02-11 18:31:05 UTC-0500
CREATE_COMPLETE

Overview

Stack ID
arn:aws:cloudformation:us-east-1:471112664825:stack/wordpress/a042b500-c935-11ee-bece-124edd36d939

Description
AWS in Action: chapter 2 (Wordpress)

Status
CREATE_COMPLETE

Status reason
-

Root stack
-

Parent stack
-

Created time
2024-02-11 18:31:05 UTC-0500

Deleted time
-

Updated time
-

Drift status
NOT_CHECKED

Last drift check time
-

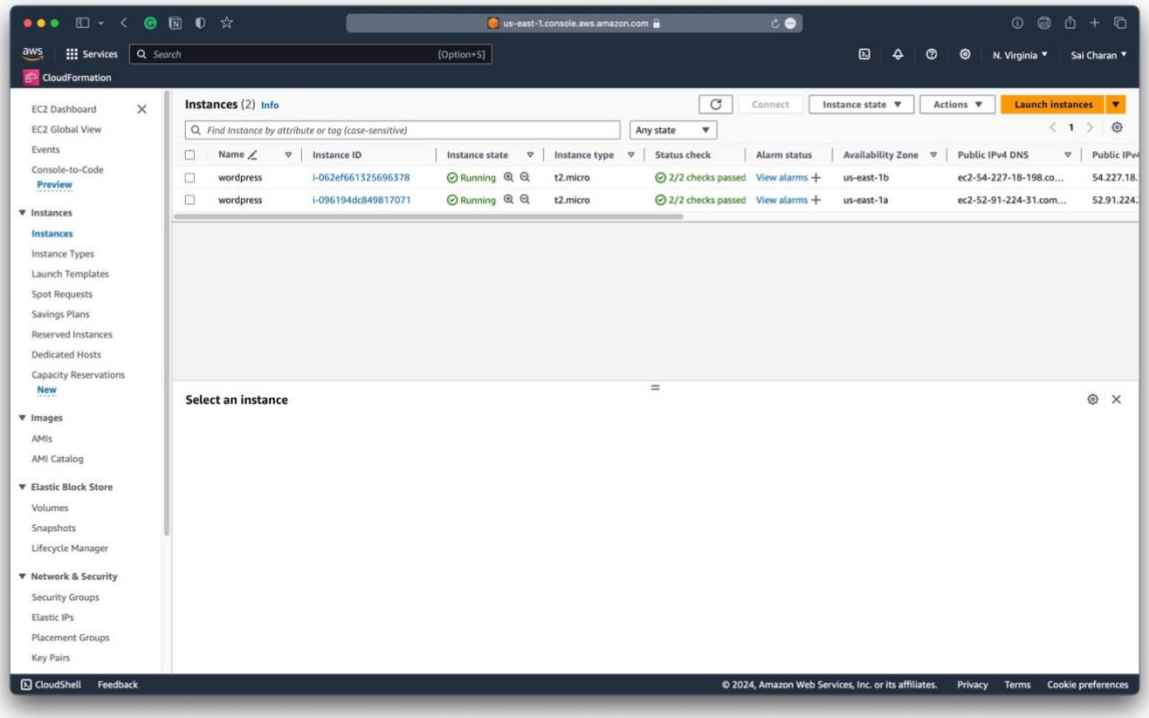
Termination protection
Deactivated

IAM role
-

Tags

Stack-level tags will apply to all supported resources in your stack. You can add up to 200 unique tags for each stack.

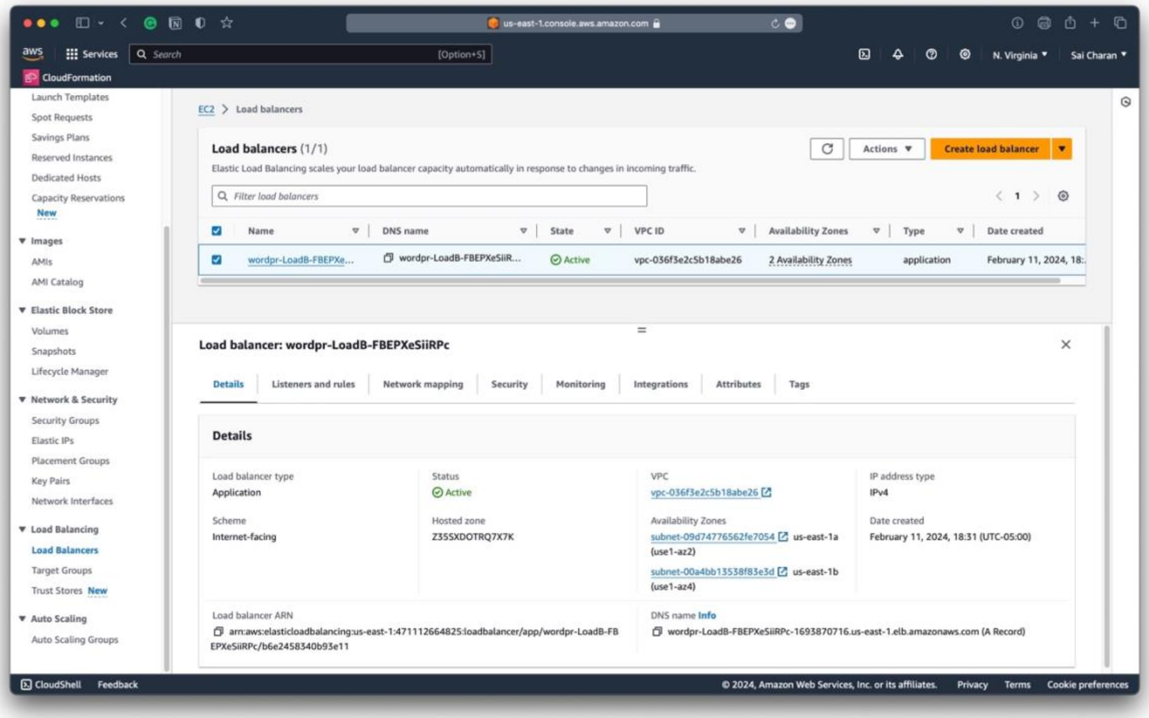
Exploring Infrastructure



Instances (2) Info

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4
□	wordpress	i-062ef661325696378	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-54-227-18-198.co...	54.227.18.
□	wordpress	i-096194dc849817071	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-52-91-224-31.com...	52.91.224.

Load Balancer:



Load balancers (1/1)

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

☑	Name	DNS name	State	VPC ID	Availability Zones	Type	Date created
☑	wordpr-LoadB-FBEPXe...	wordpr-LoadB-FBEPXeSiiR...	Active	vpc-036f3e2c5b18abe26	2 Availability Zones	application	February 11, 2024, 18:

Load balancer: wordpr-LoadB-FBEPXeSiiRPc

Details

Load balancer type
Application

Status
Active

VPC
vpc-036f3e2c5b18abe26

IP address type
IPv4

Scheme
Internet-facing

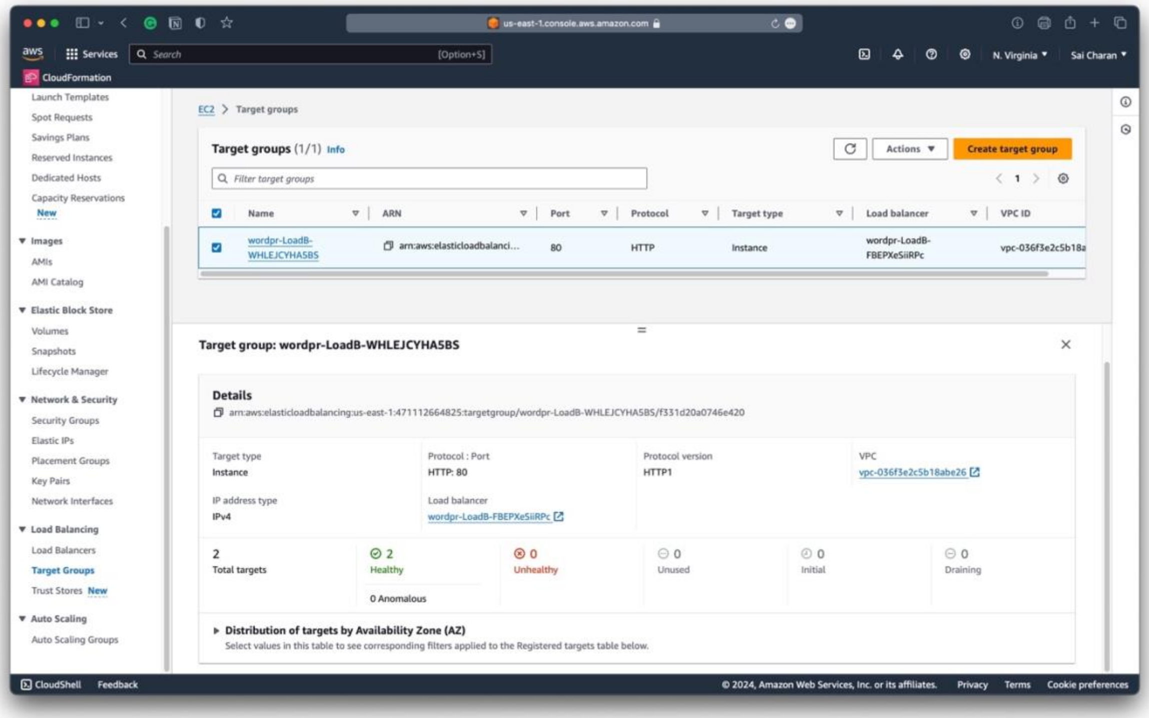
Hosted zone
Z35SXDOTRQ7X7K

Availability Zones
subnet-09d74776562fe7054 us-east-1a (use1-az2)
subnet-00a4bb13538f83e3d us-east-1b (use1-az4)

Date created
February 11, 2024, 18:31 (UTC-05:00)

Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:471112664825:loadbalancer/app/wordpr-LoadB-FBEPXeSiiRPc/b6e2458340b93e11

DNS name Info
wordpr-LoadB-FBEPXeSiiRPc-1693870716.us-east-1.elb.amazonaws.com (A Record)



Target groups (1/1) Info

☑	Name	ARN	Port	Protocol	Target type	Load balancer	VPC ID
☑	wordpr-LoadB-WHLEJCYHA5BS	arn:aws:elasticloadbalanci...	80	HTTP	Instance	wordpr-LoadB-FBEPXeSiiRPc	vpc-036f3e2c5b18a

Target group: wordpr-LoadB-WHLEJCYHA5BS

Details

arn:aws:elasticloadbalancing:us-east-1:471112664825:targetgroup/wordpr-LoadB-WHLEJCYHA5BS/f331d20a0746e420

Target type: Instance

IP address type: IPv4

Protocol : Port: HTTP: 80

Load balancer: wordpr-LoadB-FBEPXeSiiRPc

Protocol version: HTTP1

VPC: vpc-036f3e2c5b18abe26

2 Total targets

2 Healthy

0 Anomalous

0 Unhealthy

0 Unused

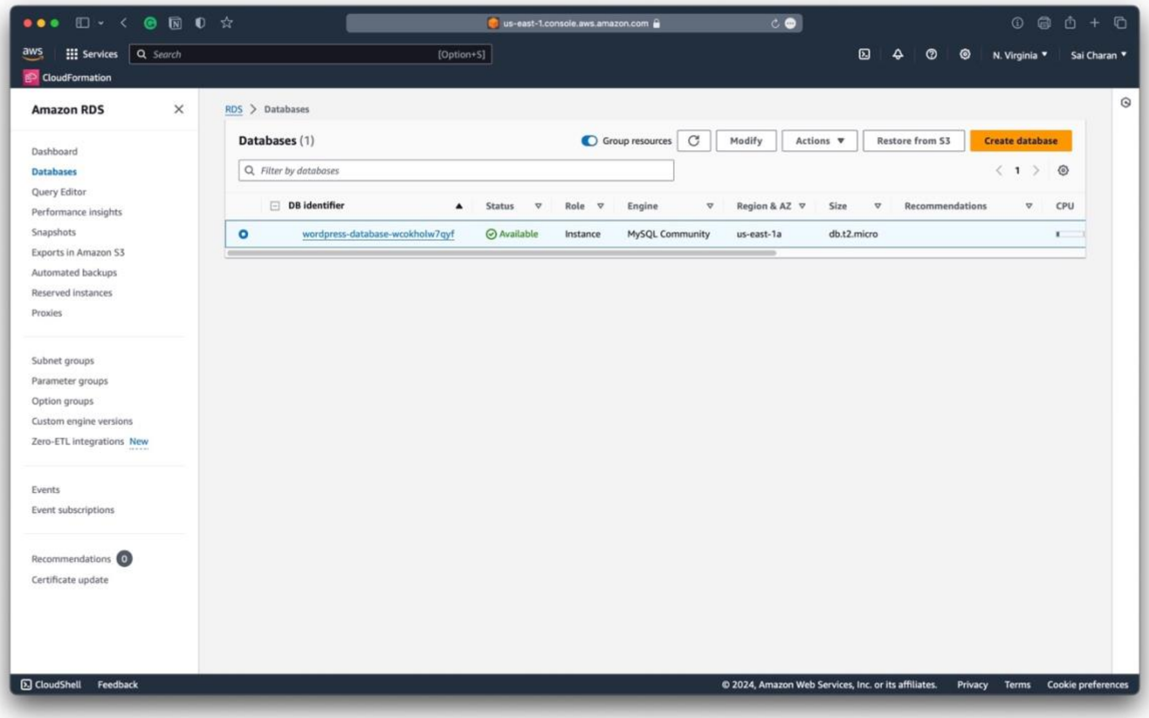
0 Initial

0 Draining

▸ Distribution of targets by Availability Zone (AZ)

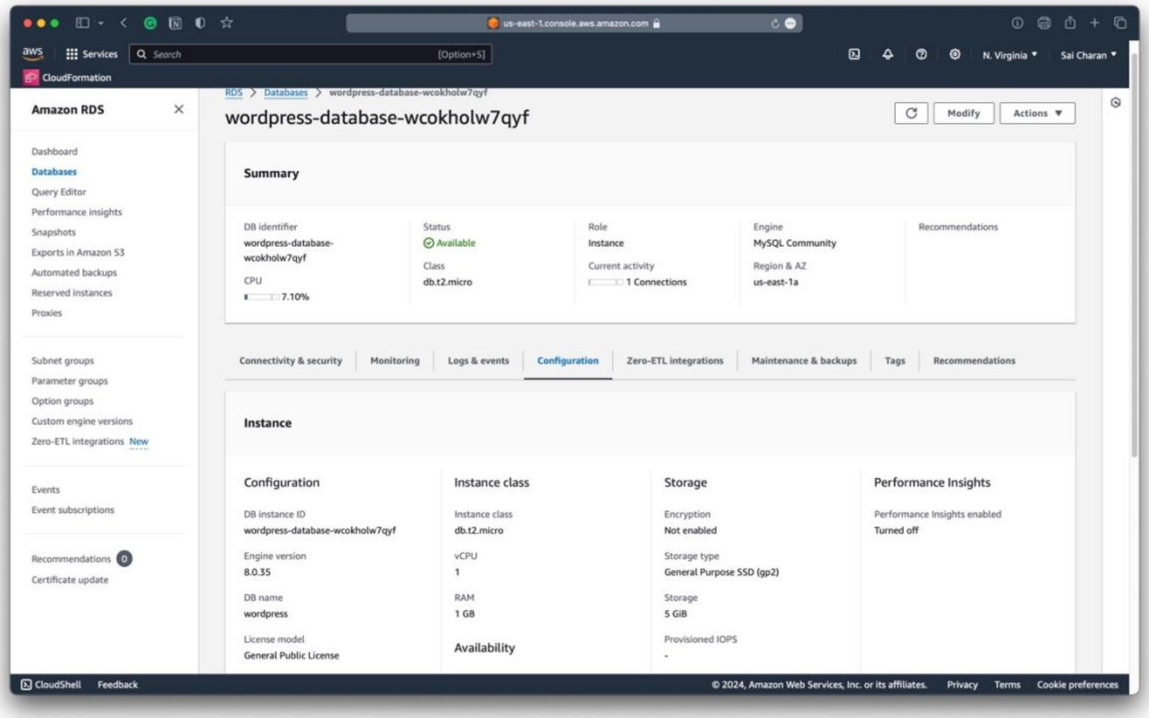
Select values in this table to see corresponding filters applied to the Registered targets table below.

MySQL database



Databases (1)

	DB identifier	Status	Role	Engine	Region & AZ	Size	Recommendations	CPU
●	wordpress-database-wcokholw7qyf	Available	Instance	MySQL Community	us-east-1a	db.t2.micro		



wordpress-database-wcokholw7qyf

Summary

DB identifier: wordpress-database-wcokholw7qyf

CPU: 7.10%

Status: Available

Class: db.t2.micro

Role: Instance

Current activity: 1 Connections

Engine: MySQL Community

Region & AZ: us-east-1a

Recommendations

Instance

Configuration

DB instance ID: wordpress-database-wcokholw7qyf

Engine version: 8.0.35

DB name: wordpress

License model: General Public License

Instance class

Instance class: db.t2.micro

vCPU: 1

RAM: 1 GB

Availability

Storage

Encryption: Not enabled

Storage type: General Purpose SSD (gp2)

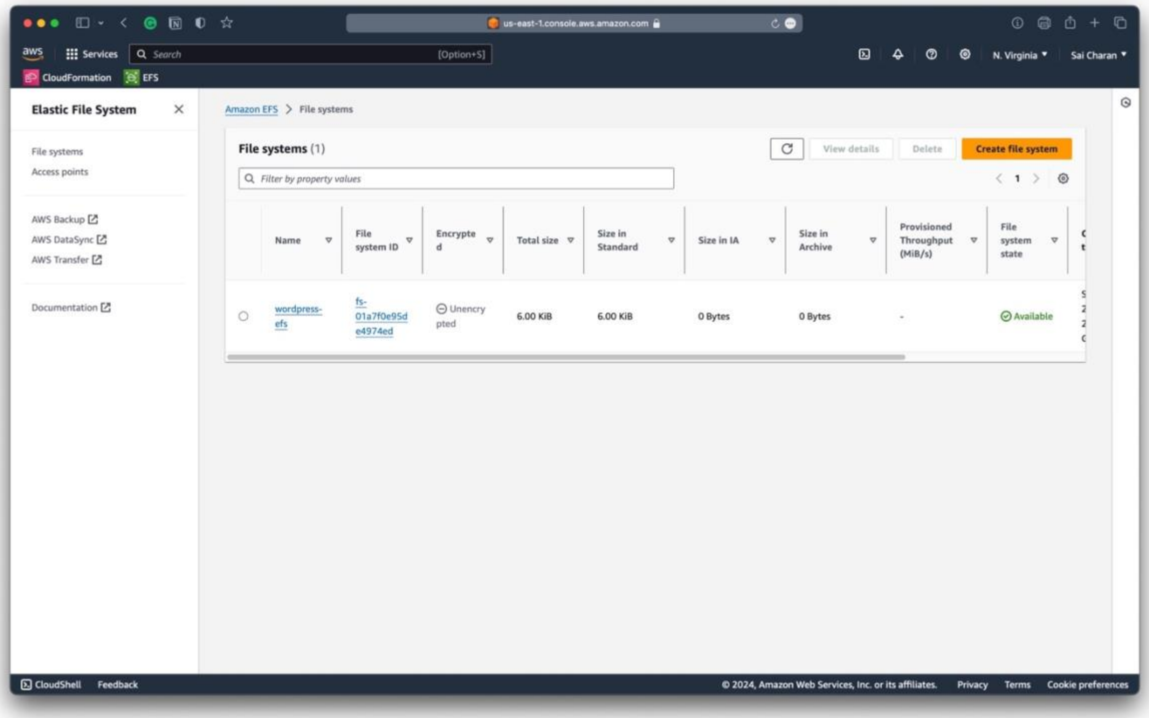
Storage: 5 GiB

Provisioned IOPS: -

Performance Insights

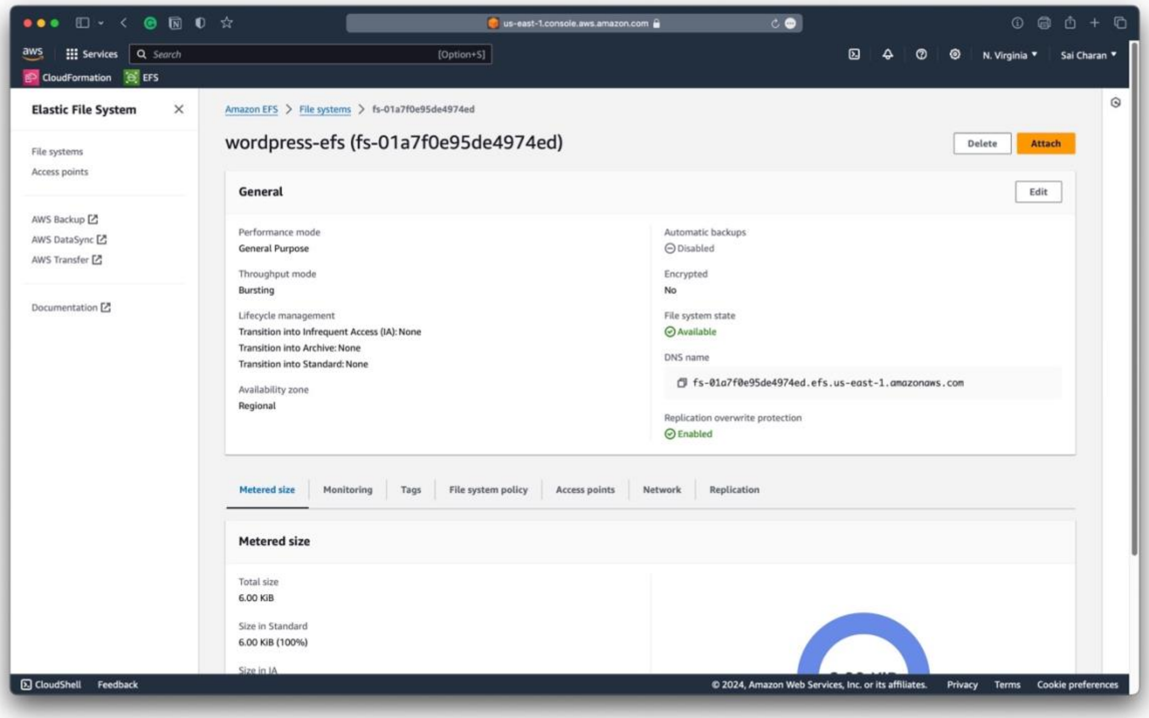
Performance Insights enabled: Turned off

Elastic File System



File systems (1)

	Name	File system ID	Encrypted	Total size	Size in Standard	Size in IA	Size in Archive	Provisioned Throughput (MiB/s)	File system state
○	wordpress-efs	fs-01a7f0e95de4974ed	Unencrypted	6.00 KiB	6.00 KiB	0 Bytes	0 Bytes	-	Available



Amazon EFS > File systems > fs-01a7f0e95de4974ed

wordpress-efs (fs-01a7f0e95de4974ed)

General

Performance mode: General Purpose

Throughput mode: Bursting

Lifecycle management:
Transition into Infrequent Access (IA): None
Transition into Archive: None
Transition into Standard: None

Availability zone: Regional

Automatic backups: Disabled

Encrypted: No

File system state: Available

DNS name: fs-01a7f0e95de4974ed.efs.us-east-1.amazonaws.com

Replication overwrite protection: Enabled

Metered size | Monitoring | Tags | File system policy | Access points | Network | Replication

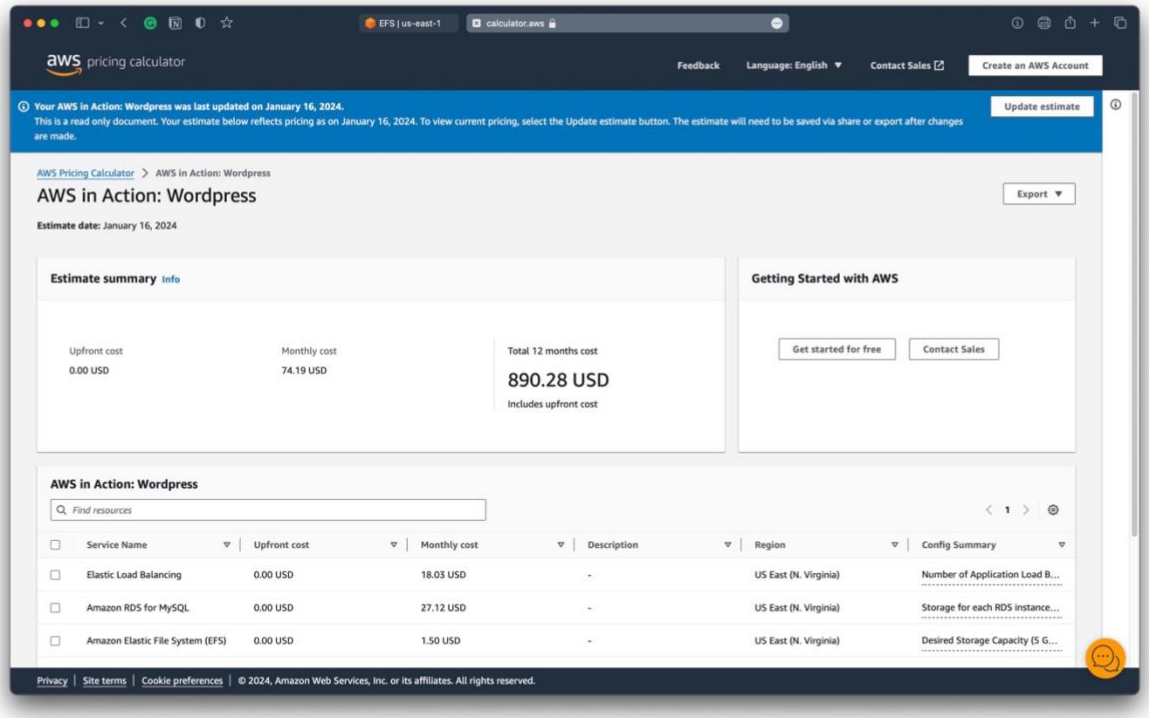
Metered size

Total size: 6.00 KiB

Size in Standard: 6.00 KiB (100%)

Size in IA

AWS Pricing Calculator



Your AWS in Action: Wordpress was last updated on January 16, 2024.
This is a read only document. Your estimate below reflects pricing as on January 16, 2024. To view current pricing, select the Update estimate button. The estimate will need to be saved via share or export after changes are made.

AWS Pricing Calculator > AWS in Action: Wordpress

AWS in Action: Wordpress

Estimate date: January 16, 2024

Estimate summary

Upfront cost: 0.00 USD

Monthly cost: 74.19 USD

Total 12 months cost: 890.28 USD
Includes upfront cost

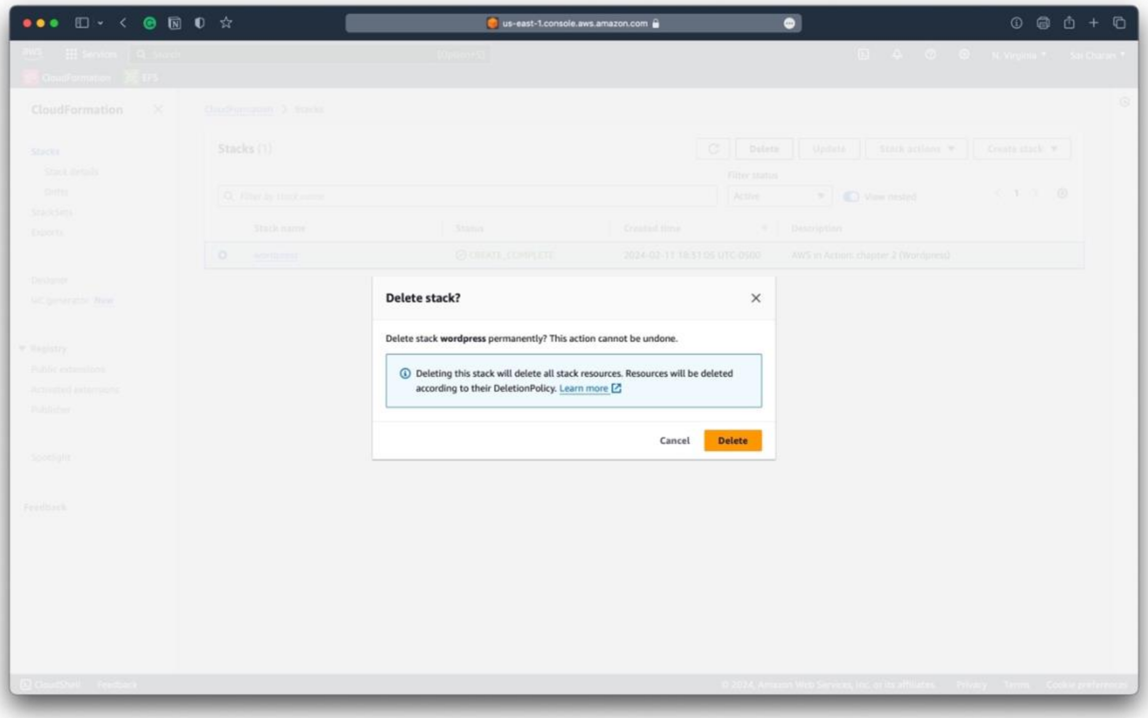
Getting Started with AWS

Get started for free | Contact Sales

AWS in Action: Wordpress

Service Name	Upfront cost	Monthly cost	Description	Region	Config Summary
Elastic Load Balancing	0.00 USD	18.03 USD	-	US East (N. Virginia)	Number of Application Load B...
Amazon RDS for MySQL	0.00 USD	27.12 USD	-	US East (N. Virginia)	Storage for each RDS instance...
Amazon Elastic File System (EFS)	0.00 USD	1.50 USD	-	US East (N. Virginia)	Desired Storage Capacity (5 G...

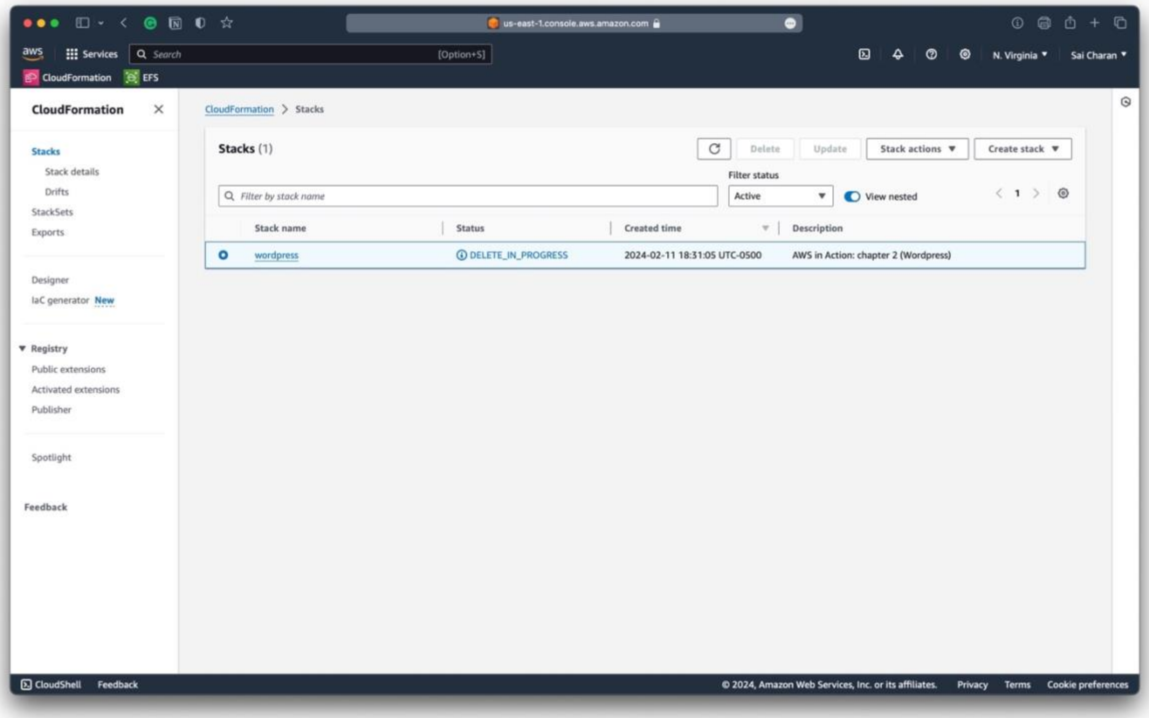
Deleting Infrastructure:



Delete stack?

Delete stack **wordpress** permanently? This action cannot be undone.

ⓘ Deleting this stack will delete all stack resources. Resources will be deleted according to their DeletionPolicy. Learn more

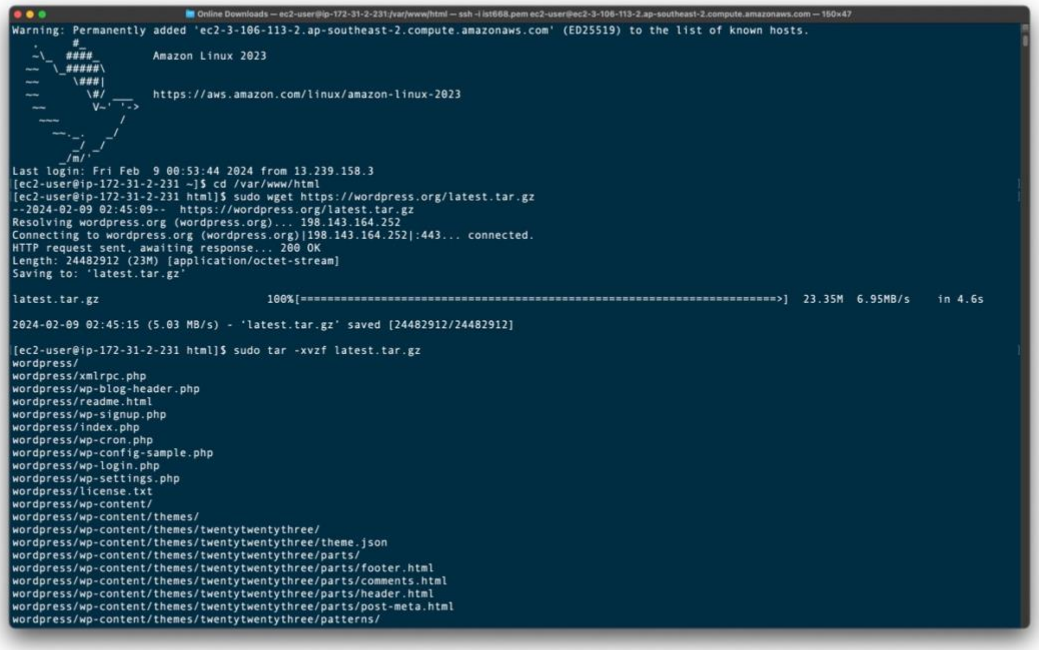


Stacks (1)

Filter status: Active

	Stack name	Status	Created time	Description
●	wordpress	ⓘ DELETE_IN_PROGRESS	2024-02-11 18:31:05 UTC-0500	AWS in Action: chapter 2 (Wordpress)

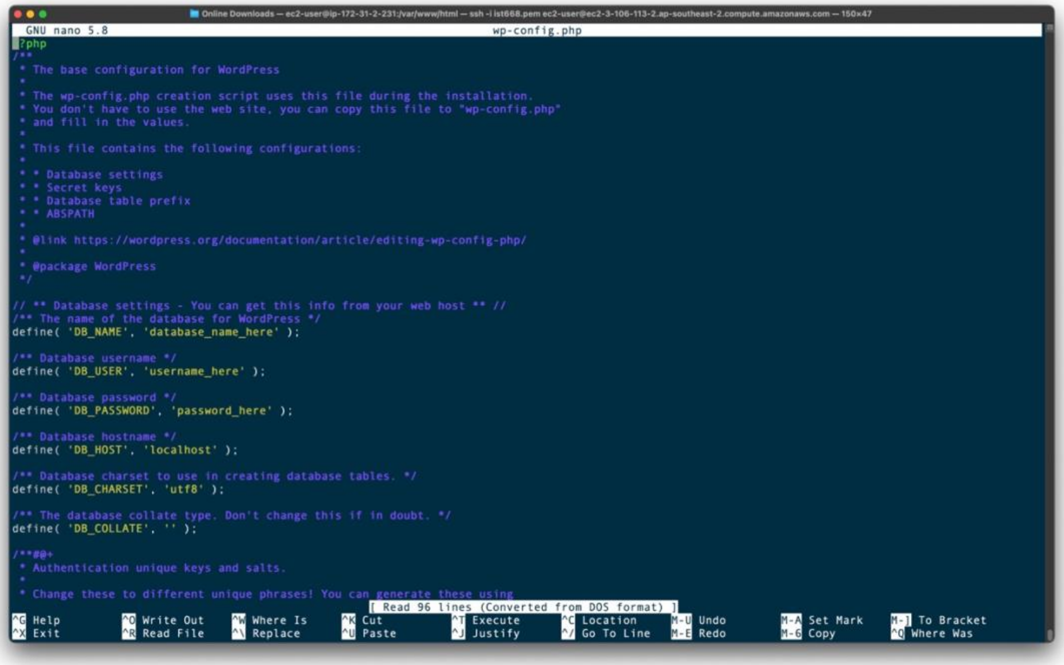
Before referring to the textbook, I tried using Mac Terminal and Internet source to finish the assignment but, I have faced a few issues with the terminal commands. as I'm still a beginner in learning the LINUX and Mac Terminal Commands. But after referring to the Textbook the process is easy to understand. I'm attaching a few of the screenshots that I have done using the Command with Mac Terminal and AWS instance.



Online Downloads — ec2-user@ip-172-31-2-231:/var/www/html — ssh -i ist668.pem ec2-user@ec2-3-106-113-2.ap-southeast-2.compute.amazonaws.com — 150×47

```
Warning: Permanently added 'ec2-3-106-113-2.ap-southeast-2.compute.amazonaws.com' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Feb  9 00:53:44 2024 from 13.239.158.3
[ec2-user@ip-172-31-2-231 ~]$ cd /var/www/html
[ec2-user@ip-172-31-2-231 html]$ sudo wget https://wordpress.org/latest.tar.gz
--2024-02-09 02:45:09--  https://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 24482912 (23M) [application/octet-stream]
Saving to: 'latest.tar.gz'

latest.tar.gz                  100%[=========================================================================>]  23.35M  6.95MB/s    in 4.6s

2024-02-09 02:45:15 (5.03 MB/s) - 'latest.tar.gz' saved [24482912/24482912]

[ec2-user@ip-172-31-2-231 html]$ sudo tar -xvzf latest.tar.gz
wordpress/
wordpress/xmlrpc.php
wordpress/wp-blog-header.php
wordpress/readme.html
wordpress/wp-signup.php
wordpress/index.php
wordpress/wp-cron.php
wordpress/wp-config-sample.php
wordpress/wp-login.php
wordpress/wp-settings.php
wordpress/license.txt
wordpress/wp-content/
wordpress/wp-content/themes/
wordpress/wp-content/themes/twentytwentythree/
wordpress/wp-content/themes/twentytwentythree/theme.json
wordpress/wp-content/themes/twentytwentythree/parts/
wordpress/wp-content/themes/twentytwentythree/parts/footer.html
wordpress/wp-content/themes/twentytwentythree/parts/comments.html
wordpress/wp-content/themes/twentytwentythree/parts/header.html
wordpress/wp-content/themes/twentytwentythree/parts/post-meta.html
wordpress/wp-content/themes/twentytwentythree/patterns/
```

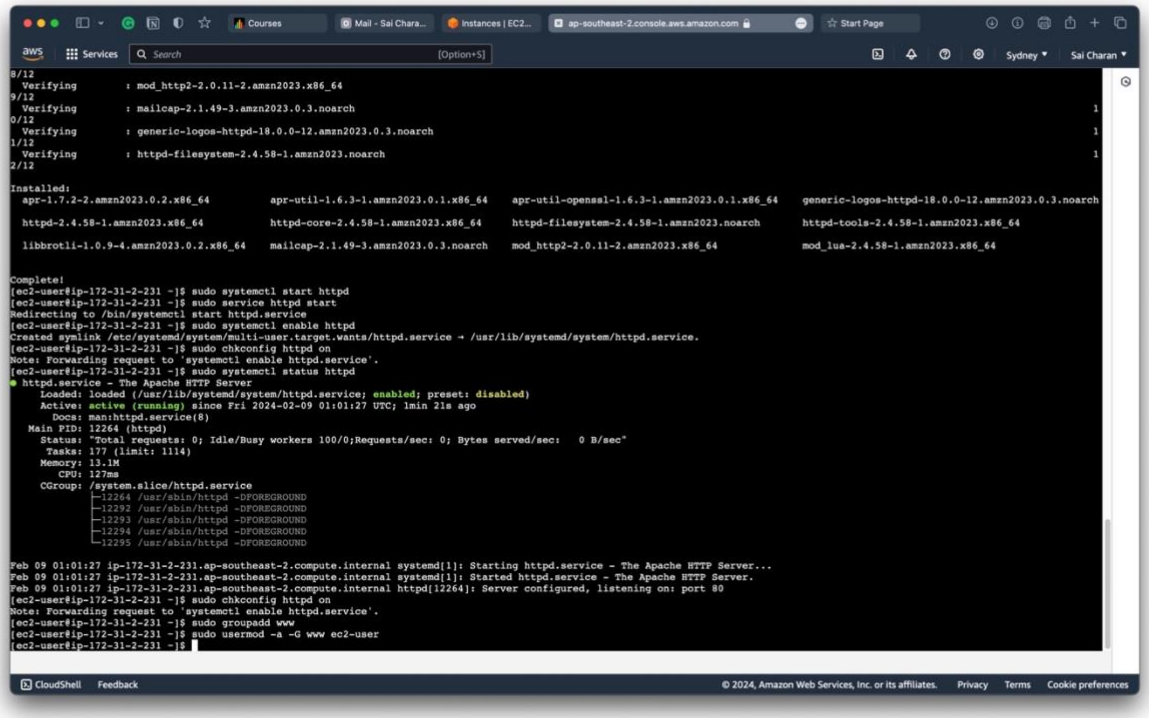
After connecting the instance using SSH in the terminal, I just directly downloaded "WordPress" with the Command " sudo wget https://wordpress.org/latest.tar.gz " .



Online Downloads — ec2-user@ip-172-31-2-231:/var/www/html — ssh -i ist668.pem ec2-user@ec2-3-106-113-2.ap-southeast-2.compute.amazonaws.com — 150×47

```
  GNU nano 5.8                                   wp-config.php
?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the installation.
 * You don't have to use the web site, you can copy this file to "wp-config.php"
 * and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * Database settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://wordpress.org/documentation/article/editing-wp-config-php/
 *
 * @package WordPress
 */

// ** Database settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'database_name_here' );

/** Database username */
define( 'DB_USER', 'username_here' );

/** Database password */
define( 'DB_PASSWORD', 'password_here' );

/** Database hostname */
define( 'DB_HOST', 'localhost' );

/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );

/** The database collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );

/**#@+
 * Authentication unique keys and salts.
 *
 * Change these to different unique phrases! You can generate these using
                                     [ Read 96 lines (Converted from DOS format) ]
^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark   M-] To Bracket
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy       ^Q Where Was
```

Even when I tried connecting to the instance using Web Connect Console, I failed with commands.



```
8/12
  Verifying        : mod_http2-2.0.11-2.amzn2023.x86_64
9/12
  Verifying        : mailcap-2.1.49-3.amzn2023.0.3.noarch                                                                          1
0/12
  Verifying        : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch                                                             1
1/12
  Verifying        : httpd-filesystem-2.4.58-1.amzn2023.noarch                                                                     1
2/12

Installed:
  apr-1.7.2-2.amzn2023.0.2.x86_64          apr-util-1.6.3-1.amzn2023.0.1.x86_64    apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64    generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch

  httpd-2.4.58-1.amzn2023.x86_64           httpd-core-2.4.58-1.amzn2023.x86_64     httpd-filesystem-2.4.58-1.amzn2023.noarch       httpd-tools-2.4.58-1.amzn2023.x86_64

  libbrotli-1.0.9-4.amzn2023.0.2.x86_64    mailcap-2.1.49-3.amzn2023.0.3.noarch    mod_http2-2.0.11-2.amzn2023.x86_64              mod_lua-2.4.58-1.amzn2023.x86_64


Complete!
[ec2-user@ip-172-31-2-231 ~]$ sudo systemctl start httpd
[ec2-user@ip-172-31-2-231 ~]$ sudo service httpd start
Redirecting to /bin/systemctl start httpd.service
[ec2-user@ip-172-31-2-231 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-172-31-2-231 ~]$ sudo chkconfig httpd on
Note: Forwarding request to 'systemctl enable httpd.service'.
[ec2-user@ip-172-31-2-231 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
     Active: active (running) since Fri 2024-02-09 01:01:27 UTC; 1min 21s ago
       Docs: man:httpd.service(8)
   Main PID: 12264 (httpd)
     Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec:   0 B/sec"
      Tasks: 177 (limit: 1114)
     Memory: 13.1M
        CPU: 127ms
     CGroup: /system.slice/httpd.service
             ├─12264 /usr/sbin/httpd -DFOREGROUND
             ├─12292 /usr/sbin/httpd -DFOREGROUND
             ├─12293 /usr/sbin/httpd -DFOREGROUND
             ├─12294 /usr/sbin/httpd -DFOREGROUND
             └─12295 /usr/sbin/httpd -DFOREGROUND

Feb 09 01:01:27 ip-172-31-2-231.ap-southeast-2.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Feb 09 01:01:27 ip-172-31-2-231.ap-southeast-2.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Feb 09 01:01:27 ip-172-31-2-231.ap-southeast-2.compute.internal httpd[12264]: Server configured, listening on: port 80
[ec2-user@ip-172-31-2-231 ~]$ sudo chkconfig httpd on
Note: Forwarding request to 'systemctl enable httpd.service'.
[ec2-user@ip-172-31-2-231 ~]$ sudo groupadd www
[ec2-user@ip-172-31-2-231 ~]$ sudo usermod -a -G www ec2-user
[ec2-user@ip-172-31-2-231 ~]$ 
```