

# DAY 2

Student Name - NAVEEN MADHAVAN

Sunday, September 26, 2021

## CLOUD COMPUTING ESSENTIALS - SEPT '21

### Assignment - 2

Create an EC2 instance with termination protection

1. Create an EC2 instance

SS : Instance details

SS : Every step (instance type, storage, tags. Network)

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"

Quick Start

My AMIs

AWS Marketplace

Community AMIs

Free tier only ⓘ

**Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-087c17d1fe0178315 (64-bit x86) / ami-029c64b3c205e6cce (64-bit Arm)**

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is approaching end of life on December 31, 2020 and has been removed from this wizard.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

**macOS Big Sur 11.6 - ami-0355f1ed5537c0368**

The macOS Big Sur AMI is an EBS-backed, AWS-supported image. This AMI includes the AWS Command Line Interface, Command Line Tools for Xcode, Amazon SSM Agent, and Homebrew. The AWS Homebrew Tap includes the latest versions of multiple AWS packages included in the AMI.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

**macOS Catalina 10.15.7 - ami-0ae0b6d49088fc747**

The macOS Catalina AMI is an EBS-backed, AWS-supported image. This AMI includes the AWS Command Line Interface, Command Line Tools for Xcode, Amazon SSM Agent, and Homebrew. The AWS Homebrew Tap includes the latest versions of multiple AWS packages included in the AMI.

Select 64-bit (x86) 64-bit (Arm)

Select 64-bit (Mac)

Select 64-bit (Mac)

Instance Type - t2.micro

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance families Current generation Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

Family	Type	vCPUs <span>ⓘ</span>	Memory (GiB) <span>ⓘ</span>	Instance Storage (GB) <span>ⓘ</span>	EBS-Optimized Available <span>ⓘ</span>	Network Performance <span>ⓘ</span>	IPv6 Support <span>ⓘ</span>
t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
t2	<b>t2.micro</b> Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes
t2	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
t3	t3.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

## SS Instance Details

### 2. Enable termination protection

The screenshot shows the 'Configure Instance Details' step of the AWS Launch Instance Wizard. The 'Enable termination protection' checkbox is checked, indicating that the instance will be protected against accidental termination. Other settings shown include placement group, capacity reservation, domain join directory, IAM role, shutdown behavior, stop-hibernate behavior, monitoring, tenancy, elastic inference, and credit specification.

Step 3: Configure Instance Details

Placement group  Add instance to placement group

Capacity Reservation  Open

Domain join directory  No directory

IAM role  None

Shutdown behavior  Stop

Stop - Hibernate behavior  Enable hibernation as an additional stop behavior

Enable termination protection  Protect against accidental termination

Monitoring  Enable CloudWatch detailed monitoring  
Additional charges apply.

Tenancy  Shared - Run a shared hardware instance  
Additional charges will apply for dedicated tenancy.

Elastic Inference  Add an Elastic Inference accelerator  
Additional charges apply.

Credit specification  Unlimited  
Additional charges may apply

## Storage Details

The screenshot shows the 'Add Storage' step of the AWS Launch Instance Wizard. Two EBS volumes are being added: one 8GB General Purpose SSD (gp2) volume and one 8GB General Purpose SSD (gp2) volume. Both volumes have 'Delete on Termination' checked and are set to 'Not Encrypted'. A note at the bottom states that free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage.

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-0699a041095ac5492	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted
EBS	/dev/sdb	Search (case-insensit)	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

Add New Volume

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

## Tags

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	(128 characters maximum)	Value	(256 characters maximum)	Instances	Volumes	Network Interfaces
Name		Server 1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Add another tag (Up to 50 tags maximum)

Cancel Previous Review and Launch Next: Configure Security Group

## Security Groups

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group:  Create a new security group  Select an existing security group

Security group name: My virtual firewalls

Description: Opens SSH, RDP, HTTP & HTTPS

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Anywhere	0.0.0.0/0, ::/0 e.g. SSH for Admin Desktop
RDP	TCP	3389	Anywhere	0.0.0.0/0, ::/0 e.g. SSH for Admin Desktop
HTTP	TCP	80	Anywhere	0.0.0.0/0, ::/0 e.g. SSH for Admin Desktop
HTTPS	TCP	443	Anywhere	0.0.0.0/0, ::/0 e.g. SSH for Admin Desktop

Add Rule

**Warning**  
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.

Cancel Previous Review and Launch

## Review Instance details

**Step 7: Review Instance Launch**

HTTP	TCP	443	0.0.0.0/0
HTTPS	TCP	443	::/0

**Instance Details**

Number of instances: 1  
 Network: vpc-0efa195fd1c48e1ee  
 Subnet: No preference (default subnet in any Availability Zone)  
 EBS-optimized: No  
 Monitoring: No  
 Termination protection: Yes  
 Shutdown behavior: Stop  
 Stop - Hibernate behavior: Disabled  
 Capacity Reservation: open  
 IAM role: None  
 Domain join directory: None  
 Tenancy: default  
 Credit specification: Standard  
 Host ID  
 Host resource group name  
     Affinity: Off  
     Kernel ID: Use default  
     RAM disk ID: Use default  
     Enclave: false  
 Metadata accessible: Enabled  
 Metadata version: V1 and V2 (token optional)  
 Metadata token response hop limit: 1

Purchasing option: On demand

[Edit instance details](#)

[Cancel](#) [Previous](#) [Launch](#)

## Creating and downloading new key pair

**Step 7: Review Instance Launch**

EBS-optimized: No  
 Monitoring: No  
 Termination protection: Yes  
 Shutdown behavior: Stop  
 Stop - Hibernate behavior: Disabled  
 Capacity Reservation: open  
 IAM role: None  
 Domain join directory: None  
 Tenancy: default  
 Credit specification: Standard  
 Host ID  
 Host resource group name  
     Affinity: Off  
     Kernel ID: Use default  
     RAM disk ID: Use default  
     Enclave: false  
 Metadata accessible: Enabled  
 Metadata version: V1 and V2 (token optional)  
 Metadata token response hop limit: 1  
 User data  
 Assign Public IP: Use subnet setting  
 Assign IPv6 IP: Use subnet setting

**Select an existing key pair or create a new key pair**

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance. Amazon EC2 supports ED25519 and RSA key pair types.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Create a new key pair  
 RSA  ED25519  
 Key pair name: medbrosis  
 Download Key Pair

You have to download the **private key file** (\*.pem file) before you can continue. **Store it in a secure and accessible location**. You will not be able to download the file again after it's created.

[Edit storage](#) [Edit tags](#)

[Cancel](#) [Launch Instances](#)

Launched instance successfully

The screenshot shows the AWS Launch Instance Wizard page. At the top, it says "Your instances are now launching". Below that, it shows the instance ID "i-0e5a5ba99d58e9c63" and a link to "View launch log". The URL in the address bar is "https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchInstanceWizard".

## Launch Status

The screenshot shows a tip about estimated charges: "Get notified of estimated charges" with the sub-instruction "Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier)".

### How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click [View Instances](#) to monitor your instances' status. Once your instances are in the **running** state, you can **connect** to them from the Instances screen. [Find out](#) how to connect to your instances.

### Here are some helpful resources to get you started

- [How to connect to your Linux instance](#)
- [Amazon EC2: User Guide](#)
- [Learn about AWS Free Usage Tier](#)
- [Amazon EC2: Discussion Forum](#)

While your instances are launching you can also

- [Create status check alarms](#) to be notified when these instances fail status checks. (Additional charges may apply)
- [Create and attach additional EBS volumes](#) (Additional charges may apply)
- [Manage security groups](#)

The screenshot shows the AWS EC2 Management Console Instances screen. It displays a single instance named "Server 1" with the instance ID "i-0e5a5ba99d58e9c63". The instance is listed as "Running" with a status check of "2/2 checks passed". The instance type is "t2.micro" and it is located in the "us-east-1d" availability zone with a public IPv4 DNS of "ec2-54-83-83-22". The left sidebar shows the "New EC2 Experience" feedback section and links to "EC2 Dashboard", "EC2 Global View", "Events", and "Tags".

### 3. Connect to your instance

#### SS : instance console

```
Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
11 package(s) needed for security, out of 35 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-25-213 ~]$ sudo yum update
```

i-0e5a5ba99d58e9c63 (Server 1)

Public IPs: 54.83.83.229 Private IPs: 172.31.25.213

#### Run a system update

```
device-mapper          x86_64      7:1.02.170-6.amzn2.5      amzn2-core      297 k
device-mapper-event     x86_64      7:1.02.170-6.amzn2.5      amzn2-core      192 k
device-mapper-event-libs x86_64      7:1.02.170-6.amzn2.5      amzn2-core      192 k
device-mapper-libs      x86_64      7:1.02.170-6.amzn2.5      amzn2-core      326 k
glibc                  x86_64      2.26.54.amzn2            amzn2-core      3.3 M
glibc-all-langpacks    x86_64      2.26.54.amzn2            amzn2-core      7.0 M
glibc-common           x86_64      2.26.54.amzn2            amzn2-core      772 k
glibc-locale-source    x86_64      2.26.54.amzn2            amzn2-core      3.2 M
glibc-minimal-langpack x86_64      2.26.54.amzn2            amzn2-core      31 k
grub2-common           noarch     1:2.06-2.amzn2.0.6        amzn2-core      1.7 M
grub2-efi-x64-ec2      x86_64      1:2.06-2.amzn2.0.6        amzn2-core      281 k
grub2-pc-modules       noarch     1:2.06-2.amzn2.0.6        amzn2-core      938 k
kernel-tools           x86_64      4.14.246-187.474.amzn2  amzn2-core      152 k
libblkid                x86_64      2.30.2-2.amzn2.0.5       amzn2-core      190 k
libcrypt                x86_64      2.26.54.amzn2            amzn2-core      51 k
libcurl                 x86_64      7.76.1-7.amzn2.0.2       amzn2-core      314 k
libfdisk                x86_64      2.30.2-2.amzn2.0.5       amzn2-core      237 k
libmount                x86_64      2.30.2-2.amzn2.0.5       amzn2-core      212 k
libsmartcols            x86_64      2.30.2-2.amzn2.0.5       amzn2-core      155 k
libuuid                 x86_64      2.30.2-2.amzn2.0.5       amzn2-core      79 k
lvm2                   x86_64      7:2.02.187-6.amzn2.5     amzn2-core      1.3 M
lvm2-libs               x86_64      7:2.02.187-6.amzn2.5     amzn2-core      1.1 M
openldap                x86_64      2.4.44-23.amzn2.0.2      amzn2-core      350 k
systemd                x86_64      219.78.amzn2.0.15        amzn2-core      5.0 M
systemd-libs             x86_64      219.78.amzn2.0.15        amzn2-core      408 k
systemd-sysv             x86_64      219.78.amzn2.0.15        amzn2-core      97 k
util-linux              x86_64      2.30.2-2.amzn2.0.5       amzn2-core      2.3 M

Transaction Summary
=====
Install  7 Packages
Upgrade  28 Packages

Total download size: 56 M
Is this ok [y/d/N]:
```

i-0e5a5ba99d58e9c63 (Server 1)

Public IPs: 54.83.83.229 Private IPs: 172.31.25.213

Instance termination failed as termination protection enabled.

The screenshot shows the AWS EC2 Management Console. A red banner at the top states: "Failed to terminate an instance: The instance 'i-0e5a5ba99d58e9c63' may not be terminated. Modify its 'disableApiTermination' instance attribute and try again." Below this, the "Instances (1/1)" table lists one instance named "Server 1" with the ID "i-0e5a5ba99d58e9c63". The instance is shown as "Running" with the type "t2.micro". The status check shows "2/2 checks passed" and no alarms. The instance is located in the "us-east-1d" availability zone with a public IPv4 address of "ec2-54-83-8". On the left sidebar, under the "Instances" section, there is a link to "Images" which is currently selected.

#### 4. Disabling termination protection

The screenshot shows the same AWS EC2 Management Console interface. The "Actions" dropdown menu for the selected instance "Server 1" has been opened. The "Change termination protection" option is highlighted, indicating it is the next step to take.

## Disabled termination protection

The screenshot shows the 'Change termination protection' page for an EC2 instance. The instance ID is i-0e5a5ba99d58e9c63 (Server 1). The 'Termination protection' checkbox is unchecked, indicating it is disabled. A warning message states: 'Termination protection disabled. The instance is no longer protected against accidental termination. If the instance is terminated, data stored on ephemeral storage is lost.' There are 'Cancel' and 'Save' buttons at the bottom.

## 5.Terminated ec2 instance successfully.

The screenshot shows the EC2 Instances page. It displays two notifications at the top: 'Disabled termination protection for i-0e5a5ba99d58e9c63' and 'Successfully terminated i-0e5a5ba99d58e9c63'. Below these, the 'Instances (1/1)' table shows one row for 'Server 1' (Instance ID: i-0e5a5ba99d58e9c63), which is currently 'Shutting-down'. The table includes columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4 DNS. At the bottom, the instance details for 'i-0e5a5ba99d58e9c63 (Server 1)' are shown, including its public and private IP addresses, instance state, and public IPv4 DNS.

## Assignment 3

### Working with volumes

#### 1. Create a volume

The screenshot shows the 'Create Volume' page in the AWS Management Console. A green success message box displays 'Volume created successfully' with a checkmark icon. Below it, the 'Volume ID' is listed as 'vol-02c0281cadb6e6580'. At the bottom right of the message box is a blue 'Close' button.

#### 2. Attach volume to instance

Created a new instance (Server 2 Ubuntu) to attach volume.

The screenshot shows the 'Instances' page in the AWS Management Console. It lists two instances: 'Server 2 Ubuntu' (Instance ID: i-0d4ff5e26bc0060fe, Status: Running) and 'Server 1' (Instance ID: i-0e5a5ba99d58e9c63, Status: Terminated). The left sidebar shows navigation options like 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Tags', 'Limits', 'Instances', and 'Images'.

#### Attaching Volume

The screenshot shows the 'Volumes' page in the AWS Management Console. It lists two volumes: 'vol-02c0281cadb6e6580' (100 GiB gp2) and 'Server 2 Ubu...' (8 GiB gp2). An 'Attach Volume' dialog box is open over the list, showing the selected volume and instance. The 'Volume' field is set to 'vol-02c0281cadb6e6580 in us-east-1c'. The 'Instance' field contains a search bar with 'Search instance ID or Name tag' and the placeholder 'in us-east-1c'. The 'Device' field is set to 'i-0d09993f305320beb (Server 2 Ubuntu) (running)'. At the bottom right of the dialog are 'Cancel' and 'Attach' buttons.

Name	Volume ID	Size	Type	IOPS	Throughput	Snapshot	Created	Availability Zone	State	Alarm Status
vol-02c0281...	100 GiB	gp2	300	-	-	September 26, 202...	us-east-1c	in-use	None	
Server 2 Ubu...	vol-0017946...	8 GiB	gp2	100	-	snap-0a52a8f5...	September 26, 202...	us-east-1c	in-use	None

## Detaching volume

Are you sure you want to detach this volume?

- vol-02c0281cadb6e6580

Cancel Yes, Detach

## 3. Deleting volume

Are you sure you want to delete this volume?

- vol-02c0281cadb6e6580

Cancel Yes, Delete

Volume deleted.

Name	Volume ID	Size	Type	IOPS	Throughput	Snapshot	Created	Availability Zone	State	Alarm Status
Server 2 Ubu...	vol-0017946...	8 GiB	gp2	100	-	snap-0a52a8f5...	September 26, 202...	us-east-1c	in-use	None

## Assignment 4

### Working with elastic beanstalk

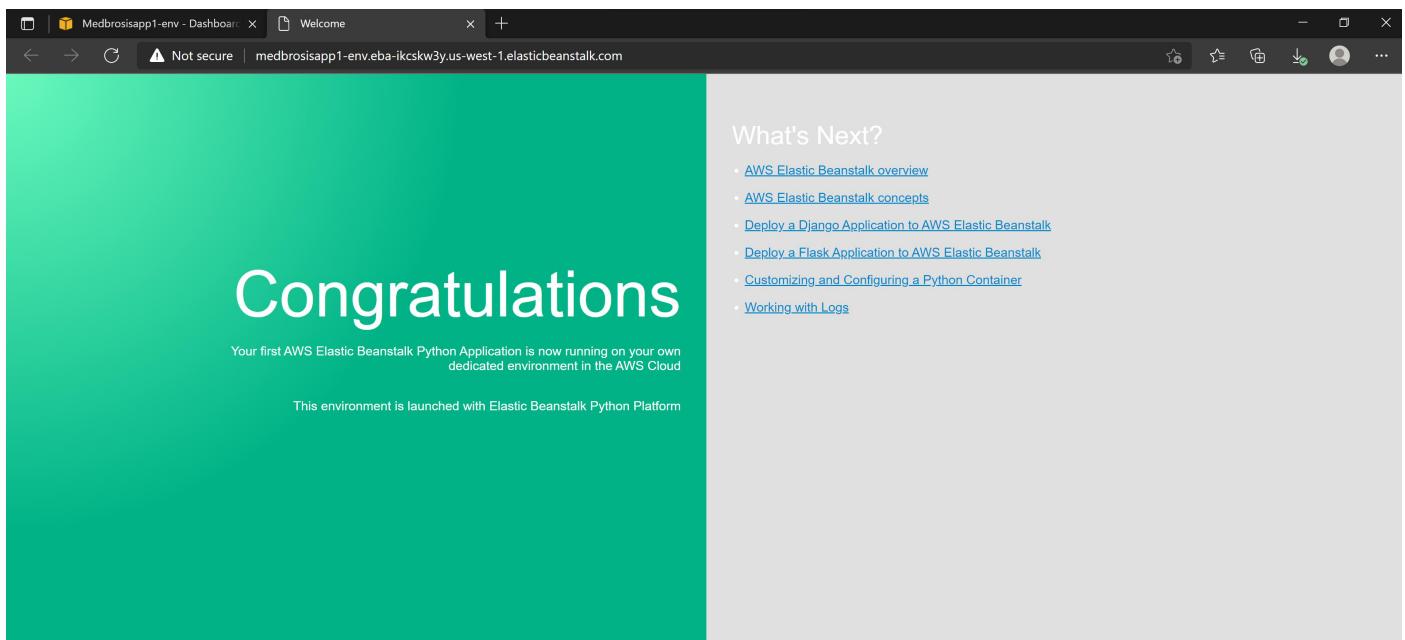
#### 1. Deploy sample application on elastic beanstalk

The screenshot shows the AWS Elastic Beanstalk console with the URL <https://us-west-1.console.aws.amazon.com/elasticbeanstalk/home?region=us-west-1#/launchEnvironment?applicationName=medbrosisapp1&environmentId=e-uw...>. The left sidebar shows 'medbrosisapp1' selected. The main area displays a progress message: 'Creating Medbrosisapp1-env' and 'This will take a few minutes.' Below this, a log window shows deployment logs:

```
10:33pm Environment health has transitioned to Pending. Initialization in progress (running for 45 seconds). There are no instances.  
10:33pm Created security group named:  
awseb-e-uw5zuqd35x-stack-AWSEBSecurityGroup-1R04OP1KJ45Y2  
10:32pm Created security group named:  
sg-021c6fc8e5c2913f5  
10:32pm Created target group named:  
arn:aws:elasticloadbalancing:us-west-1:454038252312:targetgroup/awseb-AWSEB-ZVLPLU974A4E/57a504d387fe085d  
10:32pm Using elasticbeanstalk-us-west-1-454038252312 as Amazon S3 storage bucket for environment data.  
10:32pm createEnvironment is starting.
```

The screenshot shows the AWS Elastic Beanstalk environment dashboard for 'Medbrosisapp1-env'. The URL is <https://us-west-1.console.aws.amazon.com/elasticbeanstalk/home?region=us-west-1#/environment/dashboard?applicationName=medbrosisapp1&environmentId=e-uw...>. The left sidebar shows 'Medbrosisapp1-env' selected. The main area displays a message about EnhancedHealthAuthEnabled and a table with environment details:

Health	Running version	Platform
Ok	Sample Application <a href="#">Upload and deploy</a>	Python 3.8 running on 64bit Amazon Linux 2/3.3.5 <a href="#">Change</a>



## 2. Create multiple environments within the same application.

The screenshot shows the AWS Elastic Beanstalk Applications page. The left sidebar lists 'Environments', 'Applications' (which is selected), and 'Change history'. Under 'Recent environments', 'Medbrosisapp1-env-1' and 'Medbrosisapp1-env' are listed. The main content area shows a table titled 'All applications' with one entry: 'medbrosisapp1'. The table includes columns for 'Application name', 'Environments', 'Date created', 'Last modified', and 'ARN'. The ARN value is 'arn:aws:elasticbeanstalk:us-west-1:454038252312:application/medbrosisapp1'.

Application name	Environments	Date created	Last modified	ARN
medbrosisapp1	Medbrosisapp1-env Medbrosisapp1-env-1	2021-09-26 22:32:29 UTC-0700	2021-09-26 22:32:29 UTC-0700	arn:aws:elasticbeanstalk:us-west-1:454038252312:application/medbrosisapp1

### 3. Deploy different versions of applications in different environments

The screenshot shows the AWS Elastic Beanstalk console with the URL <https://us-west-1.console.aws.amazon.com/elasticbeanstalk/home?region=us-west-1#/environment/dashboard?applicationName=medbrosisapp1&environmentId=...>. The left sidebar shows 'medbrosisapp1' selected under 'medbrosisapp1-env-1'. The main content area displays the 'Medbrosisapp1-env-1' environment details. It includes a note about Enhanced Health Auth Enabled, a 'Health' section showing 'Ok' status with a green checkmark icon, a 'Running version' section showing 'Sample Application' with a 'Upload and deploy' button, and a 'Platform' section showing 'Ruby 2.7 running on 64bit Amazon Linux 2/3.3.5' with a 'Change' button.

The screenshot shows the AWS Elastic Beanstalk environment dashboard for 'Medbrosisapp1-env-1' with the URL [medbrosisapp1-env-1.eba-ikcskw3y.us-west-1.elasticbeanstalk.com](https://medbrosisapp1-env-1.eba-ikcskw3y.us-west-1.elasticbeanstalk.com). The page has a red background and displays a large 'Congratulations' message. Below it, text says 'Your first AWS Elastic Beanstalk Ruby Application is now running on your own dedicated environment in the AWS Cloud' and 'This environment is launched with Elastic Beanstalk Ruby Platform'. To the right, there's a 'What's Next?' section with a list of links:

- [AWS Elastic Beanstalk overview](#)
- [AWS Elastic Beanstalk concepts](#)
- [Deploy a Ruby on Rails Application to AWS Elastic Beanstalk](#)
- [Deploy a Sinatra Application to AWS Elastic Beanstalk](#)
- [Customizing and Configuring a Ruby Container](#)
- [Working with Logs](#)

## Assignment 5

### Working with S3

#### 1. Create an S3 bucket

The screenshot shows the AWS S3 Management Console. A green success banner at the top states "Successfully created bucket 'medbrosis092721'". Below it, the "Buckets" section displays one bucket named "medbrosis092721" located in "US West (N. California) us-west-1". The bucket is marked as "Bucket and objects not public" and was created on "September 26, 2021, 20:55:53 (UTC-07:00)".

#### 2. Upload object into S3 bucket

The screenshot shows the AWS S3 Management Console with an "Upload succeeded" message. The "Upload: status" summary table indicates 1 file uploaded successfully (3.1 MB) and 0 files failed.

Destination	Succeeded	Failed
s3://medbrosis092721	1 file, 3.1 MB (100.00%)	0 files, 0 B (0%)

#### 3. Make bucket and object public

The screenshot shows the AWS S3 Management Console for the "medbrosis092721" bucket. The "Permissions" tab is selected. Under "Access", the setting "Objects can be public" is shown.

The screenshot shows the AWS S3 console with a green success banner at the top stating "Successfully edited public access". Below it, a modal window titled "Make public: status" displays a message: "The information below will no longer be available after you navigate away from this page." A summary table shows the following:

Source	Status	Details
s3://medbrosis092721	Successfully edited public access	1 object, 3.1 MB
	Failed to edit public access	0 objects

#### 4. Access the object using object url.

The screenshot shows a PDF document titled "CLOUD COMPUTING ESSENTIALS - SEPT '21" with student details: "Student Name - NAVEEN MADHAVAN" and "Saturday, September 25, 2021". Below the PDF, a screenshot of the AWS IAM console shows the "Users" section. It lists two users: "navneen" and "tony". Both users have "None" listed under "Groups", "Last activity", "MFA", "Password age", and "Active key age". The "User name" column lists "navneen" and "tony".

- End -

Thank you