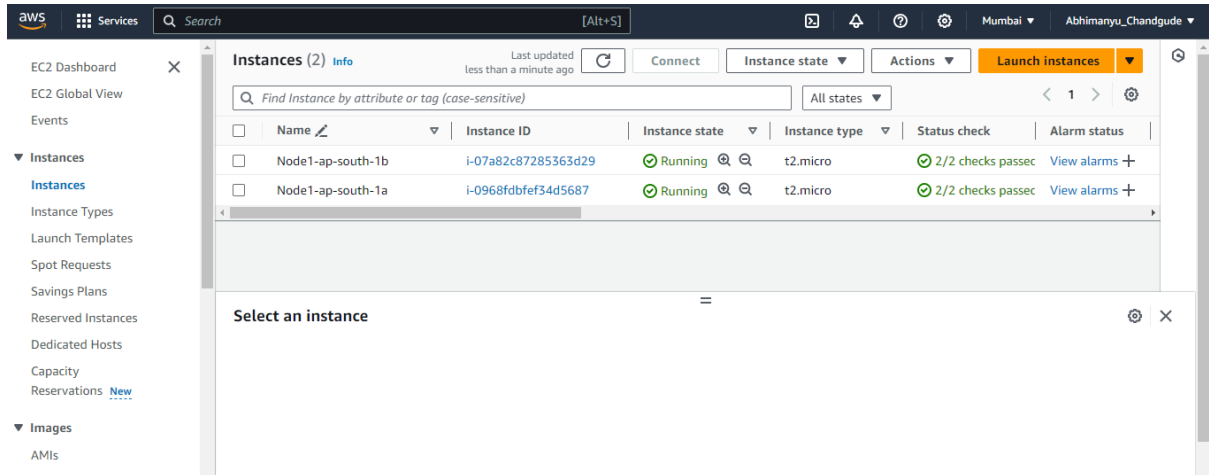
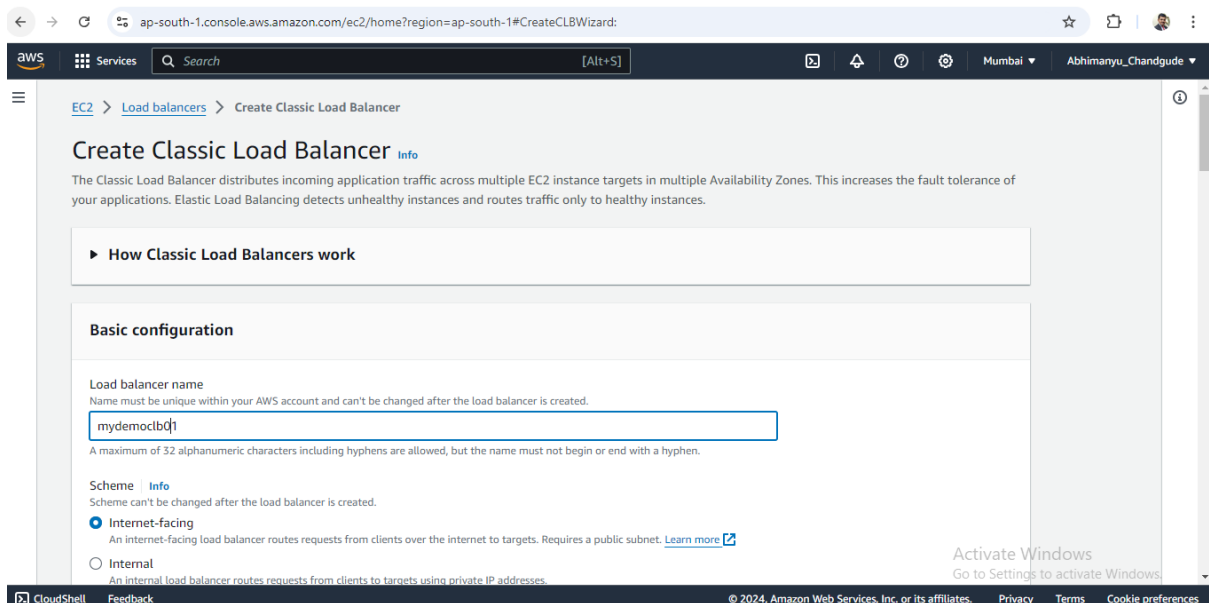


## 02<sup>nd</sup> September 2024 – ELB (Elastic Load Balancer)

1. Create two instances in two availability zones ap-south-1.



2. Create Classic Load Balancer



3. Map the network for the instances

## Network mapping [Info](#)

The load balancer routes traffic to targets in the selected subnets, and in accordance with your network settings.

VPC [Info](#)

Select the virtual private cloud (VPC) for your targets or you can [create a new VPC](#). Only VPCs with an internet gateway are available for selection. The selected VPC cannot be changed after the load balancer is created. When selecting a VPC for your load balancer, ensure each subnet has a CIDR block with at least a /27 bitmask and at least 8 free IP addresses. [Learn more](#)

-  
vpc-03ca67c5e94452d3b  
IPv4 VPC CIDR: 172.31.0.0/16

Mappings

Select at least one Availability Zone and one subnet for each zone. We recommend selecting at least two Availability Zones. The load balancer will route traffic only to targets in the selected Availability Zones. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

Availability Zones

☒ ap-south-1a (aps1-az1)

Subnet

subnet-00c35046390d8ad10  
IPv4 subnet CIDR: 172.31.32.0/20

IPv4 address

Assigned by AWS

☒ ap-south-1b (aps1-az3)

Subnet

Activate Windows  
Go to Settings to activate Windows.

#### 4. Add instances to the load balancers

**Add instances**

Select EC2 instances to register to your load balancer. Requests will be routed to registered instances that meet the health check requirements. For maximum fault tolerance, we recommend maintaining approximately equivalent numbers of instances in each Availability Zone enabled for the load balancer. If demand on your instances changes, you can register or deregister instances without disrupting the flow of requests to your application. [Learn more](#)

VPC  
vpc-03ca67c5e94452d3b

**Available instances (2/2)**

Filter available instances

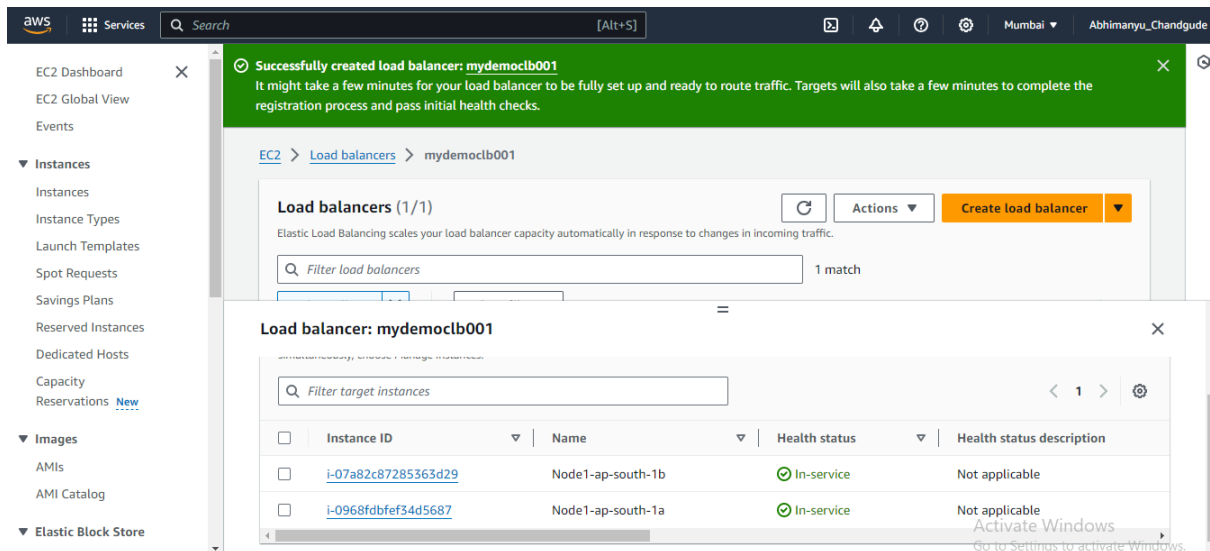
<input checked="" type="checkbox"/>	Instance ID	Name	State	Security groups	Zone
<input checked="" type="checkbox"/>	i-07a82c87285363d29	Node1-ap-south-1b	Running	Mywebsgdemo01	ap-south-1a
<input checked="" type="checkbox"/>	i-0968fdbfef34d5687	Node1-ap-south-1a	Running	Mywebsgdemo01	ap-south-1b

## 5. Created Load Balancer

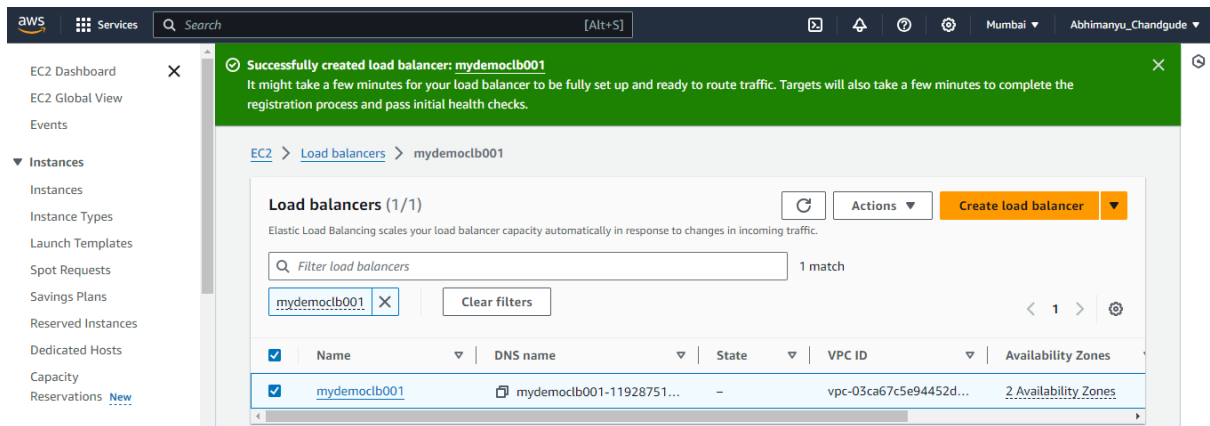
The screenshot shows the AWS Management Console interface. On the left is a navigation menu with options like EC2 Dashboard, Events, Instances, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, and Images. The main area displays a green success message at the top stating that the load balancer 'mydemoclb001' has been successfully created. Below this, the breadcrumb path is 'EC2 > Load balancers > mydemoclb001'. The 'Load balancers (1/1)' section includes a refresh button, an 'Actions' dropdown, and a 'Create load balancer' button. A descriptive text states: 'Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.' There is a search bar containing 'Filter load balancers' and a result count of '1 match'. Below the search bar is a filter box with 'mydemoclb001' selected and a 'Clear filters' button. At the bottom, there is a table listing the load balancer details.

<input checked="" type="checkbox"/>	Name	DNS name	State	VPC ID	Availability Zones
<input checked="" type="checkbox"/>	<a href="#">mydemoclb001</a>	mydemoclb001-11928751...	-	vpc-03ca67c5e94452d...	2 Availability Zones

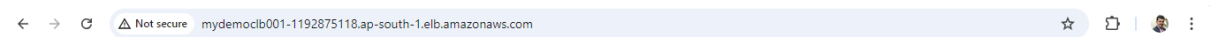
6. Health status is **In-service**



## 7. Copy DNS name URL and paste it into chrome browser



## 8. Output of index.html file



**Create and configure the service front-end-service so its accessible through ClusterIP and routes to the existing pod named front-end**

## 9. Output of healthcheck.html file

**Hi, I am running fine /h1>**

**You Have completed ELB**