

Task 7.2c

Task 1 Inspecting the Environment

What I did:

I explored the existing AWS setup to understand the starting architecture of the café's web application.

How I did it:

I used the Amazon VPC Console to inspect the virtual private cloud, including the public and private subnets, route tables, NAT gateways, internet gateway, and security groups. I also reviewed the existing CafeWebAppServer instance and security group CafeSG to check which ports were open (port 80 for HTTP, port 22 for SSH). This helped me answer the multiple-choice setup questions and understand the current infrastructure layout.

AWS Academy Cloud Architecting - Module 10 Challenge Lab Questions

View questions in: English

Question 1: Which ports are open in the *CafeSG* security group?

- Ports 80 and 443
- Port 80
- Ports 80, 443, and 3899
- Ports 22, 80, and 443

Submit

Question 2: Can you connect from the internet to instances in *Public Subnet 1*?

- Yes - If the instance has a public IP address, and the security group and network ACL allow it
- No - The public subnet has no internet gateway
- No - The public subnet has no NAT gateway configured for it
- No - The network access control list (network ACL) prevents any inbound traffic from the internet

Submit

Question 3: Should an instance in *Private Subnet 1* be able to reach the internet?

- Yes
- No

Submit

16°C Cloudy

Search

ENG IN

27-04-2025 02:47

Question 4: Should an instance in *Private Subnet 2* be able to reach the internet?

Yes
 No

Question 5: Can you connect to the *CafeWeb.AppServer* instance from the internet?

Yes
 No

Question 6: What is the name of the Amazon Machine Image (AMI)?

Amazon Linux
 WebServerAMI
 Cafe WebServer Image
 My Amazing Image

Submit

Task 2 Updating a network to work across multiple Availability Zones

What I did:

I enabled the network to work across multiple Availability Zones to ensure high availability.

How I did it:

I created a NAT Gateway in Public Subnet 2 to allow internet access for instances in Private Subnet 2. Then I updated the route table for Private Subnet 2 to direct internet-bound traffic through this new NAT Gateway. This mirrored the setup already in place for Private Subnet 1, ensuring both zones were ready for Auto Scaling.

Name	NAT gateway ID	Connectivity...	State	State message	Primary public I...	Primary private I...
-	nat-0298290d1fcff183e	Public	Available	-	44.210.159.170	10.0.0.195

Screenshot of the AWS VPC console showing the creation of a NAT gateway.

Create NAT gateway Info

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.

Connectivity type
Select a connectivity type for the NAT gateway.
 Public
 Private

Elastic IP allocation ID Info
Assign an Elastic IP address to the NAT gateway.

Additional settings Info

Route tables

VPC dashboard

Virtual private cloud

- Your VPCs
- Subnets
- Route tables**
- Internet gateways
- Egress-only internet gateways
- Carrier gateways
- DHCP option sets
- Elastic IPs
- Managed prefix lists
- NAT gateways
- Peering connections
- Route servers [New](#)

Security

- Network ACLs
- Security groups

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 16°C Cloudy ENG IN 02:50 27-04-2025

Screenshot of the AWS VPC console showing the creation of a route table.

Route tables (1/8) Info

Last updated 4 minutes ago

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
rtb-051477795c87f6702	-	-	-	Yes	vpc-016813ca9cd9430c
<input checked="" type="checkbox"/> Private Route Table 2	rtb-0a50171c8e1fe22d4	subnet-0526efc6cc526c1...	-	No	vpc-0444652a32ec39cde L2
<input type="checkbox"/> Public Route Table 2	rtb-02f36a50b56c78ae7	subnet-03623fc27d2397...	-	No	vpc-0444652a32ec39cde L2
<input type="checkbox"/> Private Route Table 1	rtb-0dcc30a26c82e2a65	subnet-07a3c369c991ca8...	-	No	vpc-0444652a32ec39cde L2
<input type="checkbox"/> Private Route Table 4	rtb-0fb7e7033dc16371c	subnet-0b29f080263999...	-	No	vpc-0444652a32ec39cde L2
<input type="checkbox"/> Private Route Table 3	rtb-0332382a56b4223d1	subnet-0d6ae306f89647...	-	No	vpc-0444652a32ec39cde L2

rtb-0a50171c8e1fe22d4 / Private Route Table 2

Details | **Routes** | Subnet associations | Edge associations | Route propagation | Tags

Routes (1)

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 16°C Cloudy ENG IN 02:50 27-04-2025

The screenshot shows the AWS VPC console's 'Edit routes' interface. A table lists routes with columns for Destination, Target, Status, and Propagated. One route goes to 'local' (Status: Active) and another to a NAT Gateway (Status: In Progress). Buttons at the bottom include 'Add route', 'Remove', 'Cancel', 'Preview', and 'Save changes'.



Task 3: Creating a Launch Template

What I did:

I created a launch template that would be used by the Auto Scaling group to spin up new EC2 instances.

How I did it:

I selected the Cafe WebServer Image AMI under “My AMIs” and chose the t2.micro instance type. I created and downloaded a new key pair for access. I also attached the CafeSG security group and assigned the CafeRole IAM role. Lastly, I added a tag with the key Name and value webserver to label my instances.

The screenshot shows the AWS EC2 console's 'Create launch template' wizard. It includes sections for selecting the instance type (t2.micro), creating a key pair (RSA), setting network and security groups, and a summary of the configuration. A note about storing the private key securely is highlighted.

The screenshot shows the AWS EC2 'Create launch template' wizard. In the 'Software Image (AMI)' section, the 'Cafe WebServer Image' (ami-07a1926d99e8a92f9) is selected. In the 'Virtual server type (instance type)' section, 't2.micro' is chosen. A tooltip for the 'Free tier' is displayed, stating: 'In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.' The 'Create Launch template' button is at the bottom right.

This screenshot shows the 'Advanced details' section of the launch template creation wizard. It includes fields for 'IAM instance profile' (set to 'arn:aws:iam::851725656643:instance-profile/CafeRole'), 'Hostname type' (set to 'Don't include in launch template'), 'DNS Hostname' (checkboxes for IPv4 and IPv6), 'Instance auto-recovery' (checkbox for 'Don't include in launch template'), 'Shutdown behavior' (checkbox for 'Don't include in launch template'), 'Stop - Hibernate behavior' (checkbox for 'Don't include in launch template'), and 'Termination protection' (checkbox for 'Don't include in launch template'). The same 'Free tier' tooltip and 'Create Launch template' button are visible.

Task4 Creating an Auto Scaling Group

What I did:

I created an Auto Scaling group to automatically launch new instances based on CPU usage.

How I did it:

Using the launch template I had created, I set up an Auto Scaling group with:

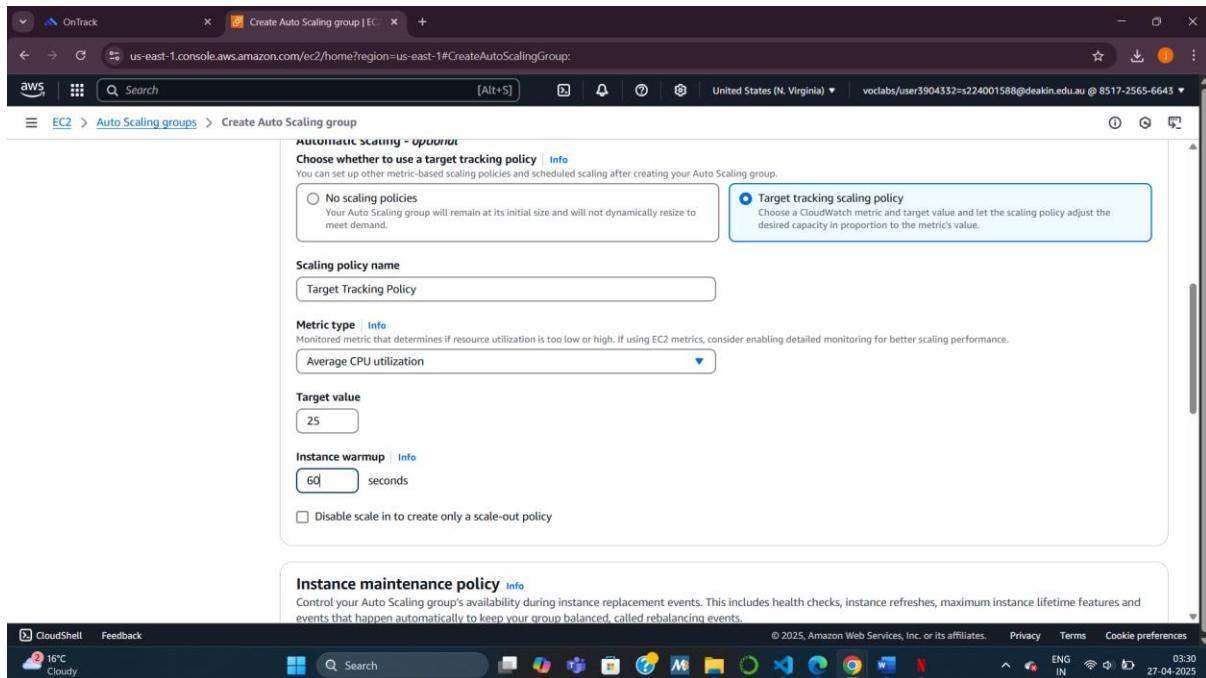
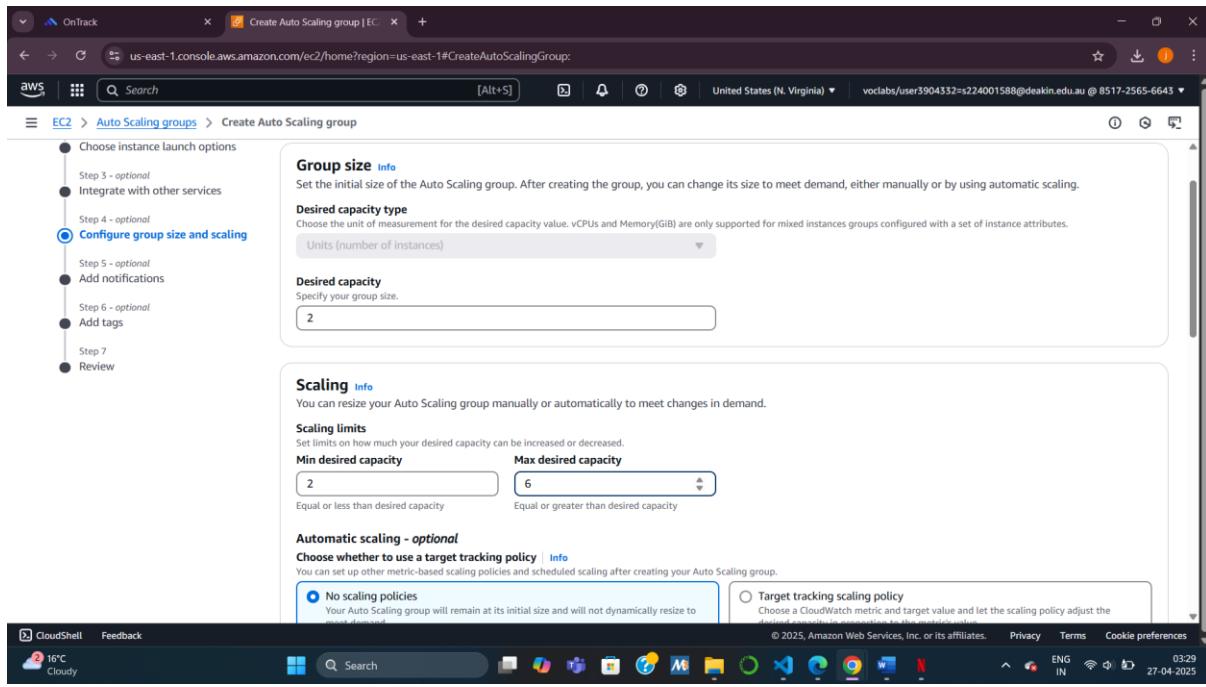
- Desired capacity: 2
- Minimum: 2
- Maximum: 6

I selected both Private Subnet 1 and 2 to ensure high availability. Then, I set a target tracking

policy with a CPU threshold of 25% and a warmup period of 60 seconds to trigger scaling events.

The screenshot shows the 'Create Auto Scaling group' wizard on the AWS console. The current step is 'Step 1 Choose launch template'. A sidebar on the left lists optional steps: Step 2 Choose instance launch options, Step 3 - optional Integrate with other services, Step 4 - optional Configure group size and scaling, Step 5 - optional Add notifications, Step 6 - optional Add tags, and Step 7 Review. The main area is titled 'Choose launch template' and contains a 'Name' field with 'Auto Scaling group name' and a value 'lab-ASG'. Below it is a note: 'For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.' A 'Launch template' dropdown is set to 'webservice', with a note: 'Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.' A 'Version' dropdown is set to 'Default (1)'. The bottom of the screen shows the Windows taskbar with various pinned icons.

The screenshot shows the 'Create Auto Scaling group' wizard on the AWS console. The current step is 'Step 7 Review'. The previous step was 'Network'. The 'Network' section includes a note: 'For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.' It has sections for 'VPC' (selected 'vpc-0444652a32ec39cde (Lab VPC)'), 'Availability Zones and subnets' (selected 'us-east-1a | subnet-07ac369c991ca83ab (Private Subnet 1)' and 'us-east-1b | subnet-0526efc6cc526c1b8 (Private Subnet 2)'), and 'Availability Zone distribution - new' (radio button selected for 'Balanced best effort'). The bottom of the screen shows the Windows taskbar with various pinned icons.



Task5 Creating an Application Load Balancer

What I did:

I made the web application accessible to the internet by creating an Application Load Balancer (ALB).

How I did it:

I created an HTTP ALB, attached it to both Public Subnet 1 and 2, and created a new security group that allowed HTTP (port 80) traffic from anywhere. Then I set up a target group and linked it to my ALB. Finally, I modified the Auto Scaling group to register instances to the target group automatically so the load balancer could distribute traffic evenly.

OnTrack Compare and select load balancer type us-east-1.console.aws.amazon.com/e2/home?region=us-east-1#SelectCreateELBWizard: aws Search [Alt+S] United States (N. Virginia) vocabs/user3904332=s224001588@deakin.edu.au @ 8517-2565-6643

EC2 > Load balancers > Compare and select load balancer type

Compare and select load balancer type

A complete feature-by-feature comparison along with detailed highlights is also available. [Learn more](#)

Load balancer types

Application Load Balancer Info

Choose an Application Load Balancer when you need a flexible feature set for your.

Network Load Balancer Info

Choose a Network Load Balancer when you need ultra-high performance, TLS offloading.

Gateway Load Balancer Info

Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 03:31 27-04-2025

OnTrack Create application load balancer us-east-1.console.aws.amazon.com/e2/home?region=us-east-1#CreateALBWizard: aws Search [Alt+S] United States (N. Virginia) vocabs/user3904332=s224001588@deakin.edu.au @ 8517-2565-6643

EC2 > Load balancers > Create Application Load Balancer

IP pools - new Info

You can optionally choose to configure an IPAM pool as the preferred source for your load balancers IP addresses. Create or view Pools in [Amazon VPC IP Address Manager console](#)

Use IPAM pool for public IPv4 addresses

The IPAM pool you choose will be the preferred source of public IPv4 addresses. If the pool is depleted IPv4 addresses will be assigned by AWS.

Availability Zones and subnets Info

Select at least two Availability Zones and a subnet for each zone. A load balancer node will be placed in each selected zone and will automatically scale in response to traffic. The load balancer routes traffic to targets in the selected Availability Zones only.

us-east-1a (use1-az4)

Subnet

Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.

subnet-0bfeeadc21bd477e Public Subnet 1

IPv4 subnet CIDR: 10.0.0.0/24

us-east-1b (use1-az6)

Subnet

Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.

subnet-03623fc27d2397f2b Public Subnet 2

IPv4 subnet CIDR: 10.0.1.0/24

us-east-1c (use1-az1)

us-east-1d (use1-az2)

Security groups Info

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 03:38 27-04-2025

The screenshot shows the AWS EC2 Security Groups creation interface. In the 'Basic details' section, a security group named 'lb-sg' is being created. It is described as allowing only HTTP traffic and is associated with a specific VPC. The 'Inbound rules' section indicates no rules have been added yet, with a prominent 'Add rule' button.

Basic details

Security group name Info
lb-sg
Name cannot be edited after creation.

Description Info
allows only HTTP traffic

VPC Info
vpc-0444632a32ec39cde (Lab VPC)

Inbound rules Info

This security group has no inbound rules.

Add rule

Inbound rules Info

Type Info Protocol Info Port range Info Source Info Description - optional Info

HTTP TCP 80 Anyw... 0.0.0.0/0 Delete

Add rule

⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Outbound rules Info

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 16°C Cloudy ENG IN 03:40 27-04-2025

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 16°C Cloudy ENG IN 03:40 27-04-2025

Create target group

Facilitates the use of [Amazon EC2 Auto Scaling](#) to manage and scale your EC2 capacity.

IP addresses

- Supports load balancing to VPC and on-premises resources.
- Facilitates routing to multiple IP addresses and network interfaces on the same instance.
- Offers flexibility with microservice based architectures, simplifying inter-application communication.
- Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.

Lambda function

- Facilitates routing to a single Lambda function.
- Accessible to Application Load Balancers only.

Application Load Balancer

- Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
- Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

Target group name
lb-tg

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol : Port
Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation

HTTP 80

CloudShell Feedback
16°C Cloudy © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 03:42 27-04-2025

Create application load balancer

Security groups [Info](#)
A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

Security groups
Select up to 5 security groups

lb-sg sg-0578083752afce14d VPC: vpc-0444632a32ec39cd

Listeners and routing [Info](#)
A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

Listener HTTP:80

Protocol	Port	Default action
HTTP	: 80 1-65535	Forward to lb-tg Target type: Instance, IPv4 HTTP

Listener tags - optional
Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add listener tag
You can add up to 50 more tags.

CloudShell Feedback
16°C Cloudy © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 03:43 27-04-2025

Successfully created load balancer: lab-lb
It might take a few minutes for your load balancer to fully set up and route traffic. Targets will also take a few minutes to complete the registration process and pass initial health checks.

Application Load Balancers now support public IPv4 IP Address Management (IPAM)
You can get started with this feature by configuring IP pools in the Network mapping section.

lab-lb

Details

Load balancer type	Status	VPC	Load balancer IP address type
Application	Active	vpc-0444632a32ec39cde	IPv4
Scheme	Internet-facing	Hosted zone Z355XD0TRQ7X7K	Availability Zones
			subnet-03623fc27d2397f2b us-east-1b (use1-az6)
			subnet-0bfeeddc21bd477e us-east-1a (use1-az4)

Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:851725656643:loadbalancer/app/lab-lb/7dca4338373640c4

DNS name Info
lab-lb-262200281.us-east-1.elb.amazonaws.com (A Record)

Edit lab-ASG Info

Load balancing - optional

Application, Network or Gateway Load Balancer target groups
Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

Select target groups
lb-tg | HTTP Application Load Balancer: lab-lb

Classic Load Balancers

Create and attach new load balancers
Add a new load balancer

Cancel Update

Task 6 : Testing the Web Application Without Load

What I did:

I verified whether the web application was properly deployed and accessible through the load balancer.

How I did it:

I visited the DNS URL of the ALB, appending /cafe at the end. The café application loaded successfully, confirming that the infrastructure was correctly set up and instances were receiving traffic via the load balancer.

If the page didn't load, I knew to troubleshoot:

- NAT gateway routes
- Security group rules
- Launch template settings
- IAM role
- Subnet selection for Auto Scaling
- Load balancer target group mapping

Café

Our café offers an assortment of delicious and delectable pastries and coffees that will put a smile on your face. From cookies to croissants, tarts and cakes, each treat is especially prepared to excite your tastebuds and brighten your day!

Frank bakes a rich variety of cookies. Try them all!



Tea, Coffee, Lattes, and Hot Chocolate. Yes, we have it!



Our tarts are always a customer favorite!



About Us



Frank and Martha have been adding sweetness to their customers' lives.

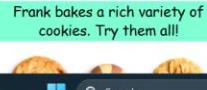
Café

Home About Us Contact Us Menu Order History




Our café offers an assortment of delicious and delectable pastries and coffees that will put a smile on your face. From cookies to croissants, tarts and cakes, each treat is especially prepared to excite your tastebuds and brighten your day!

Frank bakes a rich variety of cookies. Try them all!



Tea, Coffee,

Our tarts are always a customer favorite!



OnTrack Challenge (Café) lab: Creating a Scalable and Highly Available Environment for the Café c144539a3736925f0101091031 subnets | VPC Console

aws academy Home Modules Discussions Grades 17 Lucid (Whiteboard) Courses Calendar Inbox History Help

Challenge (Café) lab: Creating a Scalable and Highly Available Environment for the Café

Due No Due Date Points 56 Submitting an external tool

AWS EN_US 02:07 Start Lab End Lab AWS Details Details

Submission Apr 26 at 6:01pm Submission Details Grade: 46.96774193548387 (56 pts possible) Graded Anonymously: no

Comments: No Comments

EN_US NAT Gateway exists 5/5 Launch Template 5/5 Auto Scaling Group 5/5 Load Balancer 5/5 Testing the web application 5/5

Submit Submission Report Grades

Challenge Lab: Creating a Scalable and Highly Available Environment for the Café

Scenario

8°C Partly cloudy ENG IN 02:10 01-05-2025

This screenshot shows the challenge lab interface. On the right, a sidebar displays submission details: grade 46.96774193548387, due date April 26 at 6:01pm, and anonymous grading. Below this are comments and a navigation bar with 'Submit', 'Submission Report', and 'Grades' buttons. The main area contains a list of tasks with their respective grades: NAT Gateway exists (5/5), Launch Template (5/5), Auto Scaling Group (5/5), Load Balancer (5/5), and Testing the web application (5/5). At the bottom, there's a large heading for the challenge lab and a section for the scenario.

OnTrack Challenge (Café) lab: Creating a Scalable and Highly Available Environment for the Café c144539a3736925f0101091031 subnets | VPC Console

aws academy Home Modules Discussions Grades 17 Lucid (Whiteboard) Courses Calendar Inbox History Help

Challenge (Café) lab: Creating a Scalable and Highly Available Environment for the Café

Due No Due Date Points 56 Submitting an external tool

AWS EN_US 02:07 Start Lab End Lab AWS Details Details

Submission Apr 26 at 6:01pm Submission Details Grade: 46.96774193548387 (56 pts possible) Graded Anonymously: no

Comments: No Comments

I score 31/31

wer 01]	1/1
wer 02]	1/1
wer 03]	1/1
wer 04]	1/1
wer 05]	1/1

Submit Submission Report Grades

Challenge Lab: Creating a Scalable and Highly Available Environment for the Café

Scenario

8°C Partly cloudy ENG IN 02:10 01-05-2025

This screenshot shows the challenge lab interface. It features a summary table at the top with a total score of 31/31. Below this is a table showing individual student scores for five entries labeled wer 01 through wer 05, each with a score of 1/1. The rest of the interface is identical to the first screenshot, including the sidebar with submission details and the main content area with the challenge lab title and scenario section.