

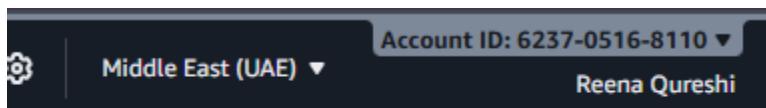
Name : Reena Qureshi

Reg No: 2023-BSE-052

Section: V-B

LAB 8

Task 1 — Create an AWS account and enable UAE (me-central-1)



Task 2 — Create IAM Admin and Lab8User with console access

Open IAM via Console search (Alt+S → "IAM").

Create the Admin user: IAM → Users → Create user. Fill:

User details

User name
The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ _ - (hyphen)

Provide user access to the AWS Management Console - optional
In addition to console access, users with SigninLocalDevelopmentAccess permissions can use the same console credentials for programmatic access without the need for access keys.

Console password

Autogenerated password
You can view the password after you create the user.

Custom password
Enter a custom password for the user.

- Must be at least 8 characters long
- Must include at least three of the following mix of character types: uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), and symbols ! @ # \$ % ^ & * () _ + -

Show password

Users must create a new password at next sign-in - Recommended
Users automatically get the [IAMUserChangePassword](#) policy to allow them to change their own password.

Info If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can you create this IAM user. [Learn more](#)

Attach policies directly → AdministratorAccess

Add user to group

Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

Copy permissions

Copy all group memberships, attached managed policies, and inline policies from an existing user.

Attach policies directly

Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Permissions policies (1/1434)



Create policy ↗

Choose one or more policies to attach to your new user.

Filter by Type

Search

All types

< 1 2 3 4 5 6 7 ... 72 > ⚙

Policy name ↗

▲ Type

▼ Attached entities

[AccessAnalyzerServiceRolePolicy](#)

AWS managed

0

[AdministratorAccess](#)

AWS managed - job function

0

[AdministratorAccess-Amplify](#)

AWS managed

0

Review and create

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details

User name
Admin

Console password type
Custom password

Require password reset
Yes

Permissions summary

Name	Type	Used as
AdministratorAccess	AWS managed - job function	Permissions policy
IAMUserChangePassword	AWS managed	Permissions policy

Tags - optional

Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags.

[Cancel](#)

[Previous](#)

[Create user](#)

[Alt+S]



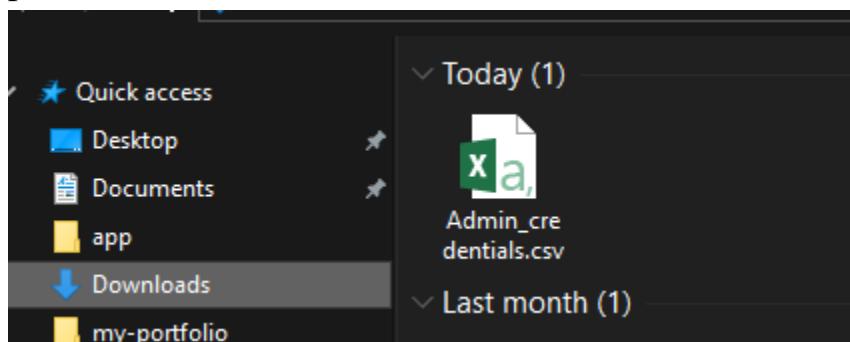
[View user](#)

Users (1) [Info](#)

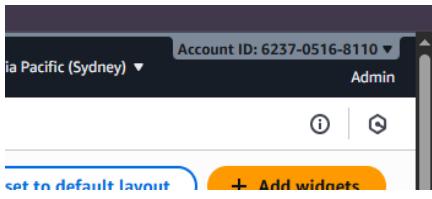
An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

User name	Path	Groups	Last activity	MFA	Password age	Console last sign-in	Access key ID	Active key age	Access key last used
Admin	/	0	Now	-	-	-	-	-	-

Download the Admin .csv and show its presence on your Windows host (do not display the password text):



Sign out of root, then sign in using the Admin account (use the signin URL from the .csv). Capture after successful Admin login:



While logged in as Admin, create Lab8User:

IAM → Users → Create user

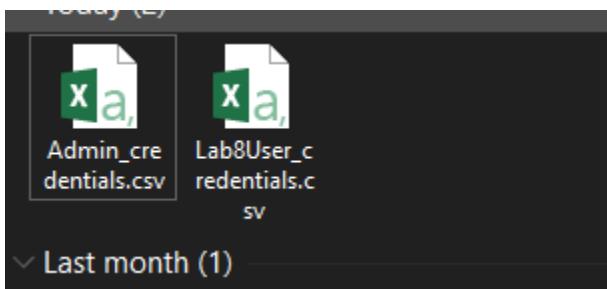
Username: Lab8User

Provide user access to the AWS Management Console

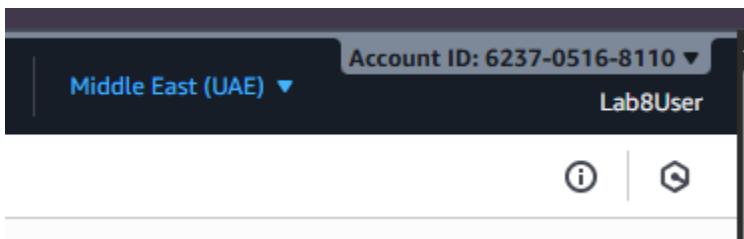
Attach AdministratorAccess policy

Capture the create-user success screen:

User name	Path	Groups	Last activity	MFA	Password age	Console last sign-in	Access key ID	Active key age	Access key last
Admin	/	0	6 minutes ago	-	4 minutes	6 minutes ago	-	-	-
Lab8User	/	0	-	-	Now	-	-	-	-



Logout Admin and login as Lab8User (use the Lab8User signin URL and credentials). Capture after login:



Task 3 — Inspect VPC resources (in UAE me-central-1)

View VPCs list.

Your VPCs						
VPCs		VPC encryption controls - new				
Your VPCs (1) Info						
<input type="checkbox"/>	Name	VPC ID	State	Encryption c...	Encryption control ...	Block Public
<input type="checkbox"/>	-	vpc-04a44d49b9440a0d6	Available	-	-	

Select a VPC above

View Subnets list. Capture

Find Subnets by attribute or tag						
<input type="checkbox"/>	Name	Subnet ID	State	VPC	Block Public.	Actions
<input type="checkbox"/>	-	subnet-036b24d524a4a8b41	Available	vpc-04a44d49b9440a0d6		
<input type="checkbox"/>	-	subnet-01a6b802f642a6c6f	Available	vpc-04a44d49b9440a0d6		
<input type="checkbox"/>	-	subnet-055e50e87b27fd1b0	Available	vpc-04a44d49b9440a0d6		

View Route Tables list.

Route tables (1) Info						Last updated 2 minutes ago	Actions	Create route table
<input type="checkbox"/>	Name	Route table ID	Explicit subnet associ...	Edge associations	Main			
<input type="checkbox"/>	-	rtb-06b58d9a83772335f	-	-	Yes			

View Network ACLs list.

Network ACLs (1) Info						Actions	Create network ACL
<input type="checkbox"/>	Name	Network ACL ID	Associated with	Default	VPC ID		
<input type="checkbox"/>	-	acl-0d2a88a318b05ec	3 Subnets	Yes	vpc-04a44d49b9440a0d6		

Task 3 summary (combine evidence):

The screenshot shows the AWS VPC Dashboard for the Middle East (UAE) region. The left sidebar includes sections for VPC dashboard, AWS Global View, and Virtual private cloud (with options for Your VPCs, Subnets, Route tables, Internet gateways, Egress-only Internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, and Endpoints). The main content area displays 'Resources by Region' with counts for UAE: VPCs (1), Subnets (3), Route Tables (1), NAT Gateways (0), VPC Peering Connections (0), and Network ACLs (1). It also features a 'Service Health' section with a link to view complete service health details, and 'Settings' and 'Additional Information' sections.

Task 4 — Launch EC2, SSH, install Docker & Docker Compose, deploy Gitea

Launch instance

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

[Launch instance](#) ▾

[Migrate a server](#) ↗

Note: Your instances will launch in the Middle East (UAE) Region

Service health

[AWS Health Dashboard](#)

Region

Middle East (UAE)

Status

✓ This service is open

Zones

Name and tags Info

Name

Lab8Machine

[Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) Info

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable image below, use the search field or choose [Browse more AMIs](#).

 *Search our full catalog including 1000s of application and OS images*

Quick Start

Amazon Linux

Ubuntu

Windows

Red Hat

SUSE Linux

Debian

 [Browse](#)

Instance type

t3.micro

Family: t3 2 vCPU 1 GiB Memory Current generation: true
On-Demand Ubuntu Pro base pricing: 0.016 USD per Hour
On-Demand Linux base pricing: 0.0125 USD per Hour
On-Demand RHEL base pricing: 0.0413 USD per Hour
On-Demand SUSE base pricing: 0.0125 USD per Hour
On-Demand Windows base pricing: 0.0217 USD per Hour

All generations

[Compare instance types](#)

Additional costs apply for AMIs with pre-installed software

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

The screenshot shows the AWS EC2 Launch Instance wizard. On the left, there's a sidebar with options like Subnet, Auto-assign public IP, Firewall (security groups), and Configure storage. The main area is titled 'Create new key pair' and shows a 'Key pair name' field containing 'Lab8Key'. It also displays two radio button options for 'Private key file format': '.pem' (selected) and '.ppk'. A note below says, 'When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance.' At the bottom right of this section are 'Cancel' and 'Launch instance' buttons. To the right, a 'Summary' panel shows the selected instance type (t3.micro), AMI (Amazon Linux 2023.9.2), and storage (1 volume(s) - 8 GiB). At the very bottom right is a 'Preview code' button.

After launch, EC2 Instances list showing Lab8Machine in "running" state and public IPv4 visible.

i-053fedbb1ef70e556 (Lab8Machine)

Instance ID
[i-053fedbb1ef70e556](#)

IPv6 address

-

Hostname type
IP name: ip-172-31-9-125.me-central-1.compute.internal

Answer private resource DNS name
IPv4 (A)

Public IPv4 address
[3.28.199.134](#) | open address

Instance state
[Running](#)

Private IP DNS name (IPv4 only)
[ip-172-31-9-125.me-central-1.compute.internal](#)

Instance type
t3.micro

Private IPv4 addresses
[172.31.9.125](#)

Public DNS
[ec2-3-28-199-134.me-central-1.compute.amazonaws.com](#)

Elastic IP addresses

On Windows host, run SSH using the downloaded .pem (PowerShell/Git Bash/Windows Terminal):

```
[ec2-user@ip-172-31-9-125 ~]$ Are you sure you want to continue connecting (yes/no)? [fingerprint]: yes
Warning: Permanently added '3.28.199.134' (ED25519) to the list of known hosts.

,      #
~\_\_ #####_      Amazon Linux 2023
~~ \_\#####\
~~  \###|
~~    \#/ ____ https://aws.amazon.com/linux/amazon-linux-2023
~~     V~' '-->
~~     /
~~.._. _/
~/ _/
~/m/
[ec2-user@ip-172-31-9-125 ~]$
```

Run the install commands on the EC2 shell:

Verified Compose.yml:

```
[ec2-user@ip-172-31-9-125 ~]$ ls -l
total 8
-rw-r--r--. 1 root root 7125 Dec 11 14:26 compose.yaml
[ec2-user@ip-172-31-9-125 ~]$
```

Add ec2-user to docker group, show groups before re-login, exit and reconnect, show groups after reconnect:

```
[ec2-user@ip-172-31-9-125 ~]$ groups
ec2-user adm wheel systemd-journal docker
[ec2-user@ip-172-31-9-125 ~]$
```

Run docker compose up -d from the directory with compose.yaml:

```
name: ec2-user_gitea_postgres
[ec2-user@ip-172-31-9-125 ~]$ docker compose up -d
WARN[0000] /home/ec2-user/compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] up 23/23
  Image postgres:alpine Pulled
  Image gitea/gitea:latest Pulled
  Network ec2-user_webnet Created
  Volume ec2-user_gitea_data Created
  Volume ec2-user_gitea_postgres Created
  Container gitea Created
  Container gitea_db Created
[ec2-user@ip-172-31-9-125 ~]$
```

Edit the security group Lab8SecurityGroup inbound rules in the EC2 console: add Custom TCP rule port 3000 source 0.0.0.0/0 and save. Capture the inbound rules after saving

The screenshot shows the AWS EC2 Security Groups page. The security group 'sg-007976612c11bc3b0 - launch-wizard-1' is selected. The 'Details' section shows the security group name, ID, owner, and various counts. The 'Inbound rules' tab is selected, displaying two rules:

Name	Security group rule ID	IP version	Type	Protocol	Port range	Source
-	sgr-0381846209fdd71a6	IPv4	Custom TCP	TCP	3000	0.0.0.0/0
-	sgr-024943661d63ad67c	IPv4	SSH	TCP	22	0.0.0.0/0

Open Gitea in your browser

The screenshot shows the Gitea Initial Configuration page. It starts with a note: "If you run Gitea inside Docker, please read the [documentation](#) before changing any settings." The 'Database Settings' section requires configuration for PostgreSQL:

- Database Type ***: PostgreSQL
- Host ***: db:5432
- Username ***: gitea
- Password ***: (redacted)
- Database Name ***: gitea
- SSL ***: Disable
- Schema**: (Leave blank for database default ("public").)

Complete initial Gitea setup (create admin user, create a repo) and capture Gitea showing the created repository

The screenshot shows a GitHub repository page. At the top, there's a navigation bar with tabs for Issues, Pull Requests, Milestones, and Explore. Below the navigation bar, the repository name 'reena/reenaqureshi_052' is displayed, along with a star icon, an 'Unwatch' button (with a count of 1), and a 'Star' button. A 'Settings' link is also visible. The main content area includes a 'Quick Guide' section with instructions for cloning the repository, a code editor with a command-line script for creating a new repository, and a 'Creating a new repository on the command line' section.

```
touch README.md
git init
git checkout -b main
git add README.md
git commit -m "first commit"
git remote add origin http://3.28.199.134:3000/reena/reenaqureshi_052.git
git push -u origin main
```

Cleanup — Remove resources to avoid charges

This screenshot shows the AWS Lambda console. It displays a single instance named 'Lab8Machine' with the ID 'i-053fedbb1ef70e556'. The instance is listed as 'Terminated'. The search bar at the top is set to 'Find Instance by attribute or tag (case-sensitive)'.

This screenshot shows the AWS Security Groups console. A green success message at the top states 'Security group (sg-007976612c11bc3b0 | launch-wizard-1) successfully deleted'. Below this, the heading 'Security Groups (2)' is followed by a table with columns for Name, Security group ID, and Security group name. The search bar at the top is set to 'Find security groups by attribute or tag'.

Reena Qureshi | Port... Keena Qureshi | Port... General | Project co...

Search [Alt+S]

Key pairs

: Store Successfully deleted 1 key pairs

Key pairs Info

Find Key Pair by attribute or tag

Name Type Created Fingerprint

No key pairs to display

[Alt+S]

This screenshot shows the AWS IAM Key Pairs page. A green success message at the top right says "Successfully deleted 1 key pairs". Below it, the heading "Key pairs Info" is followed by a search bar and a table header with columns for Name, Type, Created, and Fingerprint. A message below the table states "No key pairs to display".

User "Lab8User" deleted.

Users (1/2) Info

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

Search

This screenshot shows the AWS IAM Users page. A green success message at the top right says "User "Lab8User" deleted.". Below it, the heading "Users (1/2) Info" is followed by a search bar and a message stating "An IAM user is an identity with long-term credentials that is used to interact with AWS in an account."

Reena Qureshi | Port... Keena Qureshi | Port... General | Project co... Account ID:...

aws Search [Alt+S]

AWS Resource Groups

Resources Create Resource Group Saved Resource Groups Settings

Tagging Tag Editor Tag Policies

MANAGEMENT TOOLS

Resource Groups

Find and group your AWS resources by using queries.

You can create unlimited, single-region groups in your account, use your groups to view group-related insights, and automate tasks on group resources. Groups can be based on resource types and tag queries, or AWS CloudFormation stacks.

Start to use Resource Groups

Create a resource group

How it works

Find AWS resources in a selected region.

Create a group based on tag queries or an AWS CloudFormation stack.

View resource group specific insights.

More resources

Documentation

This screenshot shows the AWS Resource Groups landing page. It features a main heading "Resource Groups" with the sub-instruction "Find and group your AWS resources by using queries.". Below this is a detailed description of what Resource Groups are and how they can be used. To the right, there's a call-to-action button "Create a resource group" and a section titled "How it works" with three sub-points: "Find AWS resources in a selected region.", "Create a group based on tag queries or an AWS CloudFormation stack.", and "View resource group specific insights.". At the bottom right, there's a "More resources" section with a "Documentation" link.