

Cloud Computing Lab

Nayab Khazin

BSE-V B

2023-BSE-046

LAB 13

Task 0 Lab Setup (Codespace & GH CLI)

- task0_codespace_create_and_list.png

```
C:\Users\sweng>gh codespace create --repo NayabKhazin653/CC_
  Codespaces usage for this repository is paid for by SafaJ
? Choose Machine Type: 2 cores, 8 GB RAM, 32 GB storage
super-space-enigma-jrrpj9qpwxwhqxxj

C:\Users\Laptop>gh codespace list
NAME          DISPLAY NAME      REPOSIT
---          BRANCH STATE   CREATED AT
curly-goldfish-jjpj5j6w7vgjcqp5x  curly goldfish    NayabKh...
  main*     Shutdown  about 13 days ago
super-space-enigma-jrrpj9qpwx...  super space enigma  NayabKh...
  ... main     Available less than a minute ago
```

- task0_codespace_ssh_connected.png

```
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-1030-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-1030-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

Task 1 — Create IAM Group and Output Details

- task1_project_directory.png

```
@NayabKhazin653 → /workspaces/CC_nayab_046 $ cd Lab_13
```

- task1_file_created.png

```
@NayabKhazin653 → /workspaces/CC_nayab_046/Lab_13 (main) $ touch main.tf  
@NayabKhazin653 → /workspaces/CC_nayab_046/Lab_13 (main) $ ls  
EADME.md main.tf
```

- task1_main_tf.png

```
@NayabKhazin653 → /workspaces/CC_nayab_046/Lab_13 (main) $ cat main.tf  
provider "aws" {  
    shared_config_files      = ["~/.aws/config"]  
    shared_credentials_files = ["~/.aws/credentials"]  
  
resource "aws_iam_group" "developers" {  
    name = "developers"  
    path = "/groups/"  
  
    output "group_details" {  
        value = {  
            group_name = aws_iam_group.developers.name  
            group_arn  = aws_iam_group.developers.arn  
            unique_id = aws_iam_group.developers.unique_id  
        }  
    }  
}
```

- task1_terraform_init.png

```
@NayabKhazin653 → /workspaces/CC_nayab_046/Lab_13 (main) $ terraform init  
Initializing the backend...  
Initializing provider plugins...  
- Installing hashicorp/aws v6.28.0...  
- Installed hashicorp/aws v6.28.0 (signed by HashiCorp)  
Terraform has created a lock file .terraform.lock.hcl to record the provider  
selections it made above. Include this file in your version control repository  
so that Terraform can guarantee to make the same selections by default when  
you run "terraform init" in the future.  
  
Terraform has been successfully initialized!  
  
You may now begin working with Terraform. Try running "terraform plan" to see  
any changes that are required for your infrastructure. All Terraform commands  
should now work.  
  
If you ever set or change modules or backend configuration for Terraform, rerun this command to reinitialize your work-  
space. If you forget, other  
commands will detect it and remind you to do so if necessary.
```

- task1_terraform_apply.png

```

        ]
        Plan: 1 to add, 0 to change, 0 to destroy.

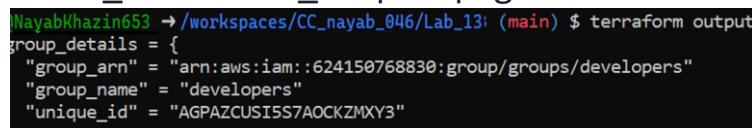
Changes to Outputs:
+ group_details = {
  + group_arn  = (known after apply)
  + group_name = "developers"
  + unique_id  = (known after apply)
}
aws_iam_group.developers: Creating...
aws_iam_group.developers: Creation complete after 2s [id=developers]
[1]
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

Outputs:

group_details = {
  "group_arn" = "arn:aws:iam::624150768830:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAZCUSI5S7AOCKZMXY3"
}

```

- task1_terraform_output.png

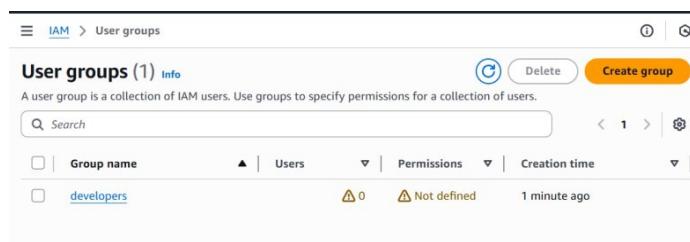


```

NayabKhazin653 → /workspaces/CC_nayab_046/Lab_13: (main) $ terraform output
group_details = {
  "group_arn" = "arn:aws:iam::624150768830:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAZCUSI5S7AOCKZMXY3"
}

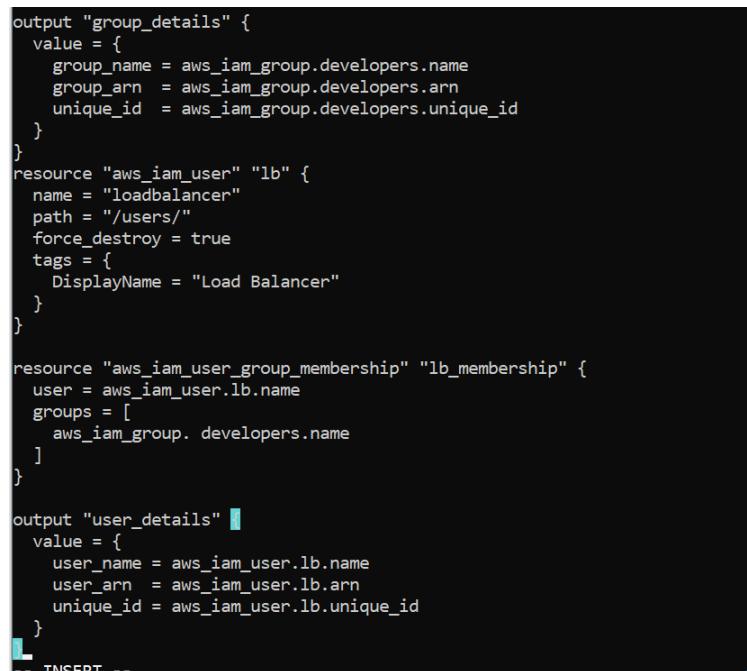
```

- task1_aws_console_group.png



Task 2 — Create IAM User with Group Membership

- task2_main_tf_user.png



```

output "group_details" {
  value = {
    group_name = aws_iam_group.developers.name
    group_arn  = aws_iam_group.developers.arn
    unique_id  = aws_iam_group.developers.unique_id
  }
}

resource "aws_iam_user" "lb" {
  name = "loadbalancer"
  path = "/users/"
  force_destroy = true
  tags = {
    DisplayName = "Load Balancer"
  }
}

resource "aws_iam_user_group_membership" "lb_membership" {
  user = aws_iam_user.lb.name
  groups = [
    aws_iam_group.developers.name
  ]
}

output "user_details" {
  value = {
    user_name = aws_iam_user.lb.name
    user_arn  = aws_iam_user.lb.arn
    unique_id = aws_iam_user.lb.unique_id
  }
}

-- INSERT --

```

- task2_terraform_apply.png

```

        }
aws_iam_user.lb: Creating...
aws_iam_user.lb: Creation complete after 1s [id=loadbalancer]
aws_iam_user_group_membership.lb_membership: Creating...
aws_iam_user_group_membership.lb_membership: Creation complete after 1s [i
d=terraform-20260112140940444400000001]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
Outputs:

```

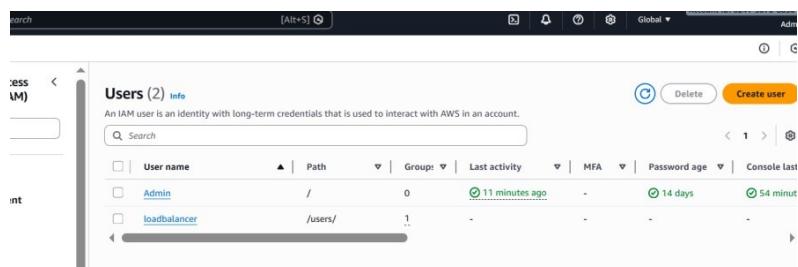
- task2_terraform_output.png

```

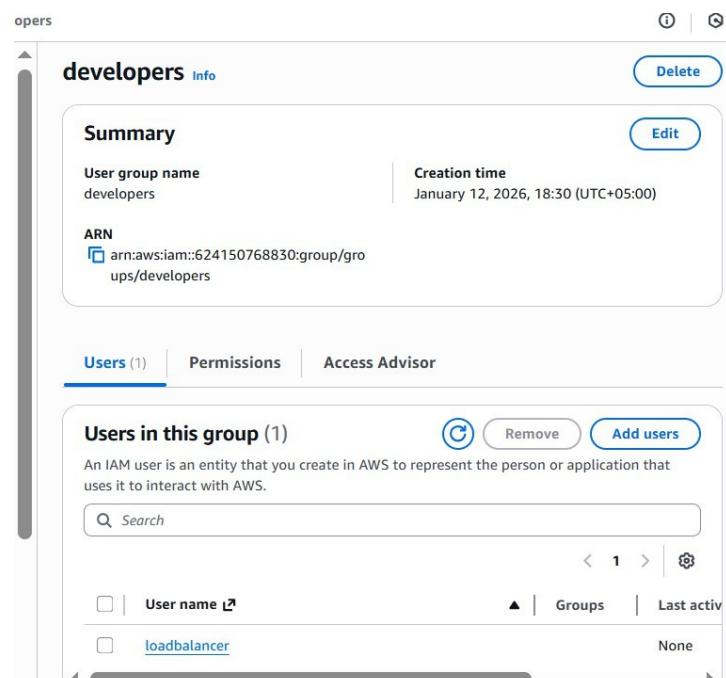
@NayabKhasin653 ~ /workspaces/CC_nayab_046/Lab_13 (main) $ terraform output
group_details = {
  "group_arn" = "arn:aws:iam::624150768830:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAZCUSI5S7AOCKZMXY3"
}
user_details = {
  "unique_id" = "AIDAZCUSI5S7FQSZOE3YF"
  "user_arn" = "arn:aws:iam::624150768830:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}

```

- task2_aws_console_user.png



- task2_aws_console_user_groups.png



Task 3 — Attach Policies to IAM Group

- task3_main_tf_policies.png

```
    unique_id = aws_iam_user.id.unique_id
}

resource "aws_iam_group_policy_attachment" "developer_ec2_fullaccess" {
    group = aws_iam_group.developers.name
    policy_arn = "arn:aws:iam::aws:policy/AmazonEC2FullAccess"
}

resource "aws_iam_group_policy_attachment" "change_password" {
    group = aws_iam_group.developers.name
    policy_arn = "arn:aws:iam::aws:policy/IAMUserChangePassword"
}

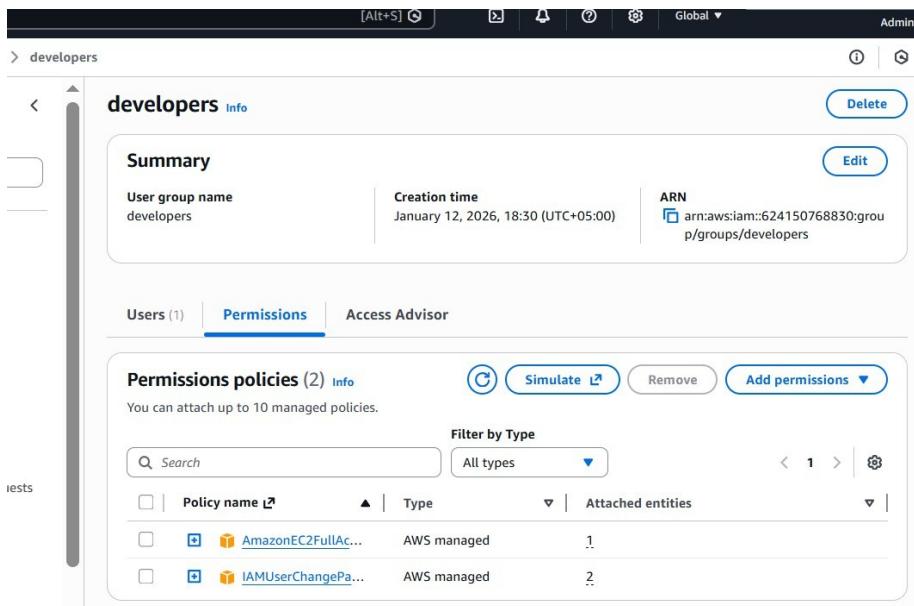
:wd
```

- task3_terraform_apply.png

```
Plan: 2 to add, 0 to change, 0 to destroy.
aws_iam_group_policy_attachment.change_password: Creating...
aws_iam_group_policy_attachment.developer_ec2_fullaccess: Creating...
aws_iam_group_policy_attachment.developer_ec2_fullaccess: Creation complete after
1s [id=developers-2026011214245453200000001]
aws_iam_group_policy_attachment.change_password: Creation complete after 1s [id=de
velopers-20260112142454646500000002]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
```

- task3_aws_console_policies.png



Task 4 — Create Login Profile for IAM User

- task4_variables_tf.png

```
NayabKhazin653 ② /workspaces/CC_nayab_046/Lab_13 (main) $ cat variables.tf
variable "iam_password" {
  description = "Temporary password for the IAM user"
  type        = string
  sensitive   = true
  default     = "1dontKnow"
```

- task4_create_login_script.png

```
GNU nano 7.2                                         create-login-profile.sh
#!/usr/bin/env bash
set -eou pipefail

USERNAME="$1"
PASSWORD="$2"

# Check if login profile already exists
if aws iam get-login-profile --user-name "$USERNAME" >/dev/null 2>&1; then
  echo "Login profile already exists for $USERNAME. Skipping."
else
  echo "Creating login profile for $USERNAME"
  aws iam create-login-profile \
    --user-name "$USERNAME" \
    --password "$PASSWORD" \
    --password-reset-required
fi
```

- task4_chmod_script.png

```
_056/Lab13 (main) $ @NayabKhazin653 ② /workspaces/CC_nayab_046/Lab_13 (main) $ chmod +x create-login-profile.sh
```

- task4_main_tf_login_profile.png

```
resource "null_resource" "create_login_profile" {
  triggers = {
    password_hash = sha256(var.iam_password)
    user         = aws_iam_user.lb.name
  }

  depends_on = [aws_iam_user.lb]

  provisioner "local-exec" {
    command = "${path.module}/create-login-profile.sh ${aws_iam_user.lb.name} '${var.iam_password}'"
  }
}
```

- task4_terraform_apply.png

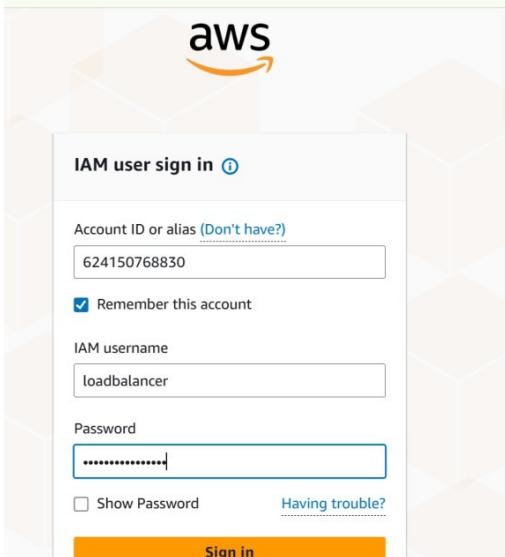
```
Plan: 1 to add, 0 to change, 0 to destroy.
null_resource.create_login_profile: Creating...
null_resource.create_login_profile: Provisioning with 'local-exec'...
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profile (local-exec): Creation complete after 7s [id=1355112466946589324]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
```

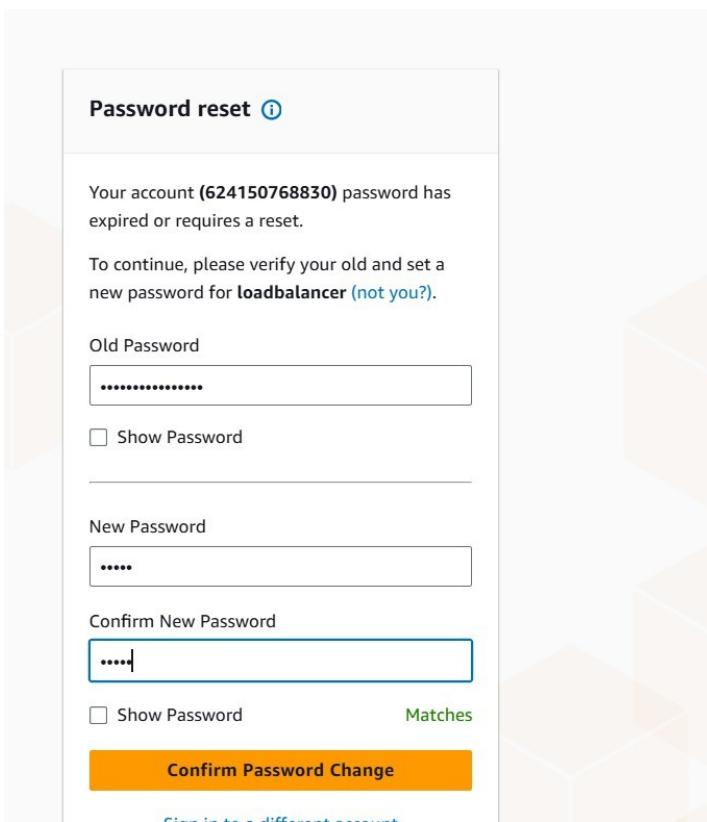
- task4_aws_cli_verify.png

```
@NayabKhazin653 eworkspaces/CC_nayab_046/Lab_13 (main) $ aws iam get-login-profile --user-name loadbalancer
{
  "LoginProfile": {
    "UserName": "loadbalancer",
    "CreateDate": "2026-01-13T04:20:43+00:00",
    "PasswordResetRequired": true
  }
}
```

- task4_aws_console_login.png



- task4_aws_console_password_reset.png



Task 5 — Generate Access Keys for IAM User

- task5_main_tf_access_keys.png

```

    command = "$[path.module]/create-login-profile.sh $aws_iam_
}
}

resource "aws_iam_access_key" "lb_access_key" {
  user = aws_iam_user.lb.name
}

output "access_key_id" {
  value = aws_iam_access_key.lb_access_key.id
}

output "access_key_secret" {
  value    = aws_iam_access_key.lb_access_key.secret
  sensitive = true
}

:wq_

```

- task5_terraform_apply.png

```

Plan: 1 to add, 0 to change, 0 to destroy.

Changes to Outputs:
+ access_key_id      = (known after apply)
+ access_key_secret = (sensitive value)
aws_iam_access_key.lb_access_key: Creating...
aws_iam_access_key.lb_access_key: Creation complete after 1s [id=AKIAZCUSI5S7HDDJJD6J]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

```

- task5_terraform_output.png

```

@NayabKhazin653 /workspaces/CC_nayab_046/Lab_13 (main) $ terraform output
access_key_id = "AKIAZCUSI5S7HDDJJD6J"
access_key_secret = <sensitive>
group_details = {
  "group_arn" = "arn:aws:iam::624150768830:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAZCUSI5S7AOCKZMXY3"
}

```

- task5_tfstate_secret. Png

```

@NayabKhazin653 eworkspaces/CC_nayab_046/Lab_13 (main) $ cat terraform.tfstate
{
  "access_key_secret": {
    "value": "6jTDuQi0350TPaRCwhzlwQwGUHBkenlIRjZeNJy9Q",
    "type": "string",
    "sensitive": true
  },
  "group_details": {
    "value": {
      "group_arn": "arn:aws:iam::624150768830:group/groups/developers",
      "group_name": "developers",
      "unique_id": "AGPAZCUSI5S7AOCKZMXY3"
    }
  }
}

```

- task5_aws_console_access_keys. Png

The screenshot shows the AWS IAM Access Management console. On the left, there's a sidebar with navigation links like 'Dashboard', 'Access Management', 'AWS Groups', 'AWS Roles', 'AWS Policies', 'AWS Identity Providers', 'AWS Lambda settings', 'AWS Temporary delegation requests', 'AWS Access Reports', 'AWS CloudTrail Analyzer', and 'Resource analysis'. The main area shows a table for 'Access keys (1)'. The table has columns for 'Type', 'Identifier', 'Certifications', and 'Created on'. A message at the top says 'No MFA devices. Assign an MFA device to improve the security of your AWS environment' with a 'Assign MFA device' button. Below the table, there's a section for the single access key, which is identified as 'AKIAZCUSI5S7HDDJJD6'. It shows details like 'Status: Active', 'Created: 2 minutes ago', 'Last used: None', 'Last used region: N/A', and 'Last used service: N/A'. There's also a 'Create access key' button.

Task 6 — Implement Terraform Remote State with S3

- task6_s3_bucket_create.png

The screenshot shows the AWS S3 'Create bucket' wizard. The first step, 'General configuration', is shown. It includes fields for 'Bucket name' (set to 'myapp-s3-bucket-05d'), 'AWS Region' (set to 'Middle East (UAE) me-central-1'), and a 'Copy settings from existing bucket - optional' section with a 'Choose bucket' button. The second step, 'Object Ownership', is shown next. It has two options: 'ACLs disabled (recommended)' (selected) and 'ACLs enabled'. The third step, 'Block Public Access settings for this bucket', is shown. It has a checked checkbox for 'Block all public access'. Below it are four sub-options under 'Block public access to buckets and objects granted through new access control lists (ACLs)': 'Block public access to buckets and objects granted through new access control lists (ACLs)', 'Block public access to buckets and objects granted through any access control lists (ACLs)', 'Block public access to buckets and objects granted through new public bucket or access point policies', and 'Block public and cross-account access to buckets and objects through any public bucket or access point policies'. The fourth step, 'Bucket Versioning', is shown. It has a checked checkbox for 'Enable'. At the bottom, there's a 'Next Step' button.

- task6_s3_bucket_versioning.png

- task6_main_tf_backend.png

```
terraform {
  backend "s3" {
    bucket = "myapp-s3-bucket-056"
    key    = "myapp/terraform.tfstate"
    region = "me-central-1"
    encrypt = true
    use_lockfile = true
  }
}

provider "aws" {
```

- task6_terraform_init_migrate.png

```
NayabKhazin653 ~/workspaces/CC_nayab_046/Lab_I3 (main) $ terraform init -migrate-state
initializing the backend...
Do you want to copy existing state to the new backend?
Pre-existing state was found while migrating the previous "local" backend to the
newly configured "s3" backend. No existing state was found in the newly
configured "s3" backend. Do you want to copy this state to the new "s3"
backend? Enter "yes" to copy and "no" to start with an empty state.

Enter a value: yes

Successfully configured the backend "s3"! Terraform will automatically
use this backend unless the backend configuration changes.
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Reusing previous version of hashicorp/null from the dependency lock file
- Using previously-installed hashicorp/aws v6.28.0
- Using previously-installed hashicorp/null v3.2.4

Terraform has been successfully initialized!
```

- task6_terraform_apply.png

```
aws_iam_user_group_membership.lb_membership: Refreshing state... [id=terraform-20260112140940444400000001]
aws_iam_access_key.lb_access_key: Refreshing state... [id=AKIAZCUSI5S7HDDJJJD6J]

No changes. Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no differences, so no
changes are needed.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.
```

- task6_s3_tfstate_file.png

The screenshot shows the AWS S3 console interface. At the top, there's a navigation bar with 'Search' (Alt+S), a refresh icon, a bell icon, a gear icon, 'Middle East (UAE)', and 'Admin'. Below the navigation bar, the path 'Buckets > myapp-s3-bucket-056 > myapp/' is shown. The main area is titled 'myapp/' and has tabs for 'Objects' (selected) and 'Properties'. Under 'Objects (1)', there's a table with one item:

Name	Type	Last modified	Size	Storage class
terraform.tfstate	tfstate	January 13, 2026, 09:51:59 (UTC+00:00)	6.7 KB	Standard

- task6_local_state_backup.png

```
@NayabKhazin653 ~/workspaces/CC_nayab_046/Lab_13 (main) $ ls -la terraform.tfstate*
-rw-rw-rw- 1 codespace codespace    0 Jan 13 04:51 terraform.tfstate
-rw-rw-rw- 1 codespace codespace 6882 Jan 13 04:51 terraform.tfstate.backup
```

- task6_terraform_destroy.png

```
aws_iam_group_policy_attachment.change_password: Destruction complete after 1s
aws_iam_group_policy_attachment.developer_ec2_fullaccess: Destruction complete after 1s
aws_iam_user_group_membership.lb_membership: Destruction complete after 1s
aws_iam_user.lb: Destroying... [id=loadbalancer]
aws_iam_group.developers: Destroying... [id=developers]
aws_iam_group.developers: Destruction complete after 0s
aws_iam_user.lb: Destruction complete after 3s

Destroy complete! Resources: 7 destroyed.
```

- task6_s3_tfstate_destroyed. Png

The screenshot shows a browser window with the URL 'myapp-s3-bucket-056.s3.me-central-1.amazonaws.com/m...'. Below the address bar, there's a 'pretty-print' checkbox. The main content area displays the JSON structure of the destroyed Terraform state file:

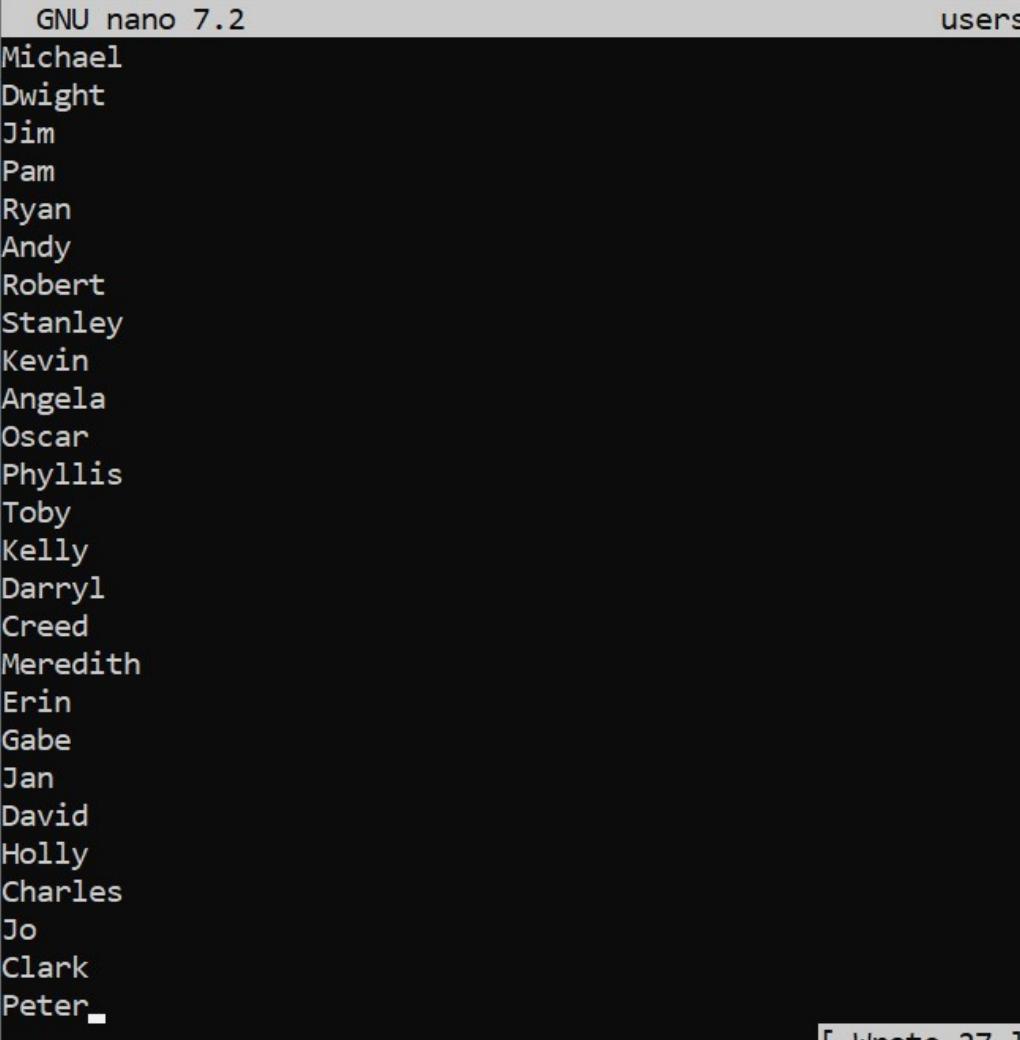
```
[
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 2,
  "lineage": "46f2a50a-befd-1786-5486-911df1a43971",
  "outputs": {},
  "resources": [],
  "check_results": null
]
```

Task 7 — Create Multiple Users from CSV File

- task7_locals_tf.png

```
@NayabKhazin653 ② /workspaces/CC_nayab_046/Lab_13 (main) $ vim locals.tf
@NayabKhazin653 ② /workspaces/CC_nayab_046/Lab_13 (main) $ cat locals.tf
locals {
    users = csvdecode(file("users.csv"))
}
```

- task7_users_csv.png



```
GNU nano 7.2                               users
Michael
Dwight
Jim
Pam
Ryan
Andy
Robert
Stanley
Kevin
Angela
Oscar
Phyllis
Toby
Kelly
Darryl
Creed
Meredith
Erin
Gabe
Jan
David
Holly
Charles
Jo
Clark
Peter
```

- task7_main_tf_multiple_users.png

```
@NayabKhazin653 ✘ /workspaces/CC_nayab_046/Lab_13 (main) $ cat main.tf
terraform {
  backend "s3" {
    bucket = "myapp-s3-bucket-056"
    key    = "myapp/terraform.tfstate"
    region = "me-central-1"
    encrypt = true
    use_lockfile = true
  }
}

provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

resource "aws_iam_group" "developers" {
  name = "developers"
  path = "/groups/"
}

output "group_details" {
  value = {
    group_name = aws_iam_group.developers.name
    group_arn  = aws_iam_group.developers.arn
    unique_id  = aws_iam_group.developers.unique_id
  }
}

# Create multiple IAM users from CSV
resource "aws_iam_user" "users" {
  for_each = { for user in local.users : user.user_name => user }

  name        = each.value.user_name
  path        = "/users/"
  force_destroy = true
```

- task7_terraform_init.png

```
@NayabKhazin653 ✘ /workspaces/CC_nayab_046/Lab_13 (main) $ terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/null from the dependency lock file
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/null v3.2.4
- Using previously-installed hashicorp/aws v6.28.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget,
commands will detect it and remind you to do so if necessary.
```

- task7_terraform_apply.png

```

null_resource.create_login_profiles["Meredith"] (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profiles["Meredith"] (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profiles["Meredith"] (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profiles["Meredith"] (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profiles["Meredith"] (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profiles["Meredith"] (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profiles["Meredith"] (local-exec): (output suppressed due to sensitive value in configuration)
null_resource.create_login_profiles["Meredith"] (local-exec): Creation complete after 10s [id=8521261764303816648]

Apply complete! Resources: 26 added, 0 changed, 26 destroyed.

```

- task7_terraform_output.png

```

@NayabKhazin653 eworkspaces/CC_nayab_046/Lab_13 (main) $ terraform output
all_access_key_secrets = <sensitive>
all_users_details = {
    "Andy" = {
        "access_key_id" = "AKIAZCUSI5S7DTWGSAXQ"
        "user_arn" = "arn:aws:iam::624150768830:user/users/Andy"
        "user_unique_id" = "AIDAZCUSI5S7EU66BIZMN"
    }
    "Angela" = {
        "access_key_id" = "AKIAZCUSI5S7JFJMRYV6"
        "user_arn" = "arn:aws:iam::624150768830:user/users/Angela"
        "user_unique_id" = "AIDAZCUSI5S7MJJHZDLXE"
    }
    "Charles" = {
        "access_key_id" = "AKIAZCUSI5S7DCMWLCW2"
        "user_arn" = "arn:aws:iam::624150768830:user/users/Charles"
        "user_unique_id" = "AIDAZCUSI5S7II2DWPH4H"
    }
    "Clark" = {
        "access_key_id" = "AKIAZCUSI5S7CF4EMTBO"
        "user_arn" = "arn:aws:iam::624150768830:user/users/Clark"
        "user_unique_id" = "AIDAZCUSI5S7DOBR3VI4X"
    }
}

```

- task7_tfstate_secrets. Png

```

@NayabKhazin653 eworkspaces/CC_nayab_046/Lab_13 (main) $ terraform state pull | grep -A 5 "all_access_key_secrets"
"all_access_key_secrets": {
    "value": {
        "Andy": "xSH56b/f9rl28TKyZkVHLqBx80+HCCJSAt+bDB/1A",
        "Angela": "EQVFQJetu7K1EdaVdNb21wZ8xdm73f1aXug7dP",
        "Charles": "vNbQV1th16w3qsLVsqLSXuL8nLvJQtbeIneNcise",
        "Clark": "bY8ye5Ya762nEcnuEK2Z8UmCrgGa2q6NuOExsTTz",
    }
}

```

- task7_aws_console_all_users. Png

User name	Path	Group	Last activity	MFA	Password age	Console last sign-in	Access key ID	Active
Admin	/	0	5 minutes ago	-	15 days	33 minutes ago	Active - AKIAZCUSI557...	7
Andy	/users/	1	-	-	4 minutes	-	Active - AKIAZCUSI557...	7
Angela	/users/	1	-	-	4 minutes	-	Active - AKIAZCUSI557...	7
Charles	/users/	1	-	-	4 minutes	-	Active - AKIAZCUSI557...	7
Clark	/users/	1	-	-	4 minutes	-	Active - AKIAZCUSI557...	6
Creed	/users/	1	-	-	4 minutes	-	Active - AKIAZCUSI557...	6
Darryl	/users/	1	-	-	4 minutes	-	Active - AKIAZCUSI557...	7
David	/users/	1	-	-	4 minutes	-	Active - AKIAZCUSI557...	6
Dwight	/users/	1	-	-	4 minutes	-	Active - AKIAZCUSI557...	6
Erin	/users/	1	-	-	4 minutes	-	Active - AKIAZCUSI557...	7
Gabe	/users/	1	-	-	4 minutes	-	Active - AKIAZCUSI557...	6
Holly	/users/	1	-	-	4 minutes	-	Active - AKIAZCUSI557...	6

- task7_aws_console_group_members.Png

Summary																															
User group name	developers	Creation time	January 13, 2026, 10:09 (UTC+05:00)																												
		ARN	arn:aws:iam::624150768830:group/groups/developers																												
Users in this group (26)																															
An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.																															
<table border="1"> <thead> <tr> <th>User name</th> <th>Groups</th> <th>Last activity</th> <th>Creation time</th> </tr> </thead> <tbody> <tr><td>Andy</td><td></td><td>None</td><td>7 minutes ago</td></tr> <tr><td>Angela</td><td></td><td>None</td><td>7 minutes ago</td></tr> <tr><td>Charles</td><td></td><td>None</td><td>7 minutes ago</td></tr> <tr><td>Clark</td><td></td><td>None</td><td>7 minutes ago</td></tr> <tr><td>Creed</td><td></td><td>None</td><td>7 minutes ago</td></tr> <tr><td>Darryl</td><td></td><td>None</td><td>7 minutes ago</td></tr> </tbody> </table>				User name	Groups	Last activity	Creation time	Andy		None	7 minutes ago	Angela		None	7 minutes ago	Charles		None	7 minutes ago	Clark		None	7 minutes ago	Creed		None	7 minutes ago	Darryl		None	7 minutes ago
User name	Groups	Last activity	Creation time																												
Andy		None	7 minutes ago																												
Angela		None	7 minutes ago																												
Charles		None	7 minutes ago																												
Clark		None	7 minutes ago																												
Creed		None	7 minutes ago																												
Darryl		None	7 minutes ago																												

- task7_aws_console_user_access_key.png

Access keys (1)	
Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active: more)	
AKIAZCUSI557K3BWG7FZ	Status
Description	Active
Last used	None
Last used region	N/A
	Created
	10 minutes ago
	Last used service
	N/A

- task7_s3_tfstate_multiple_users.png

```
{
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 5,
  "lineage": "46f2a50a-befd-1786-5486-911df1a43971",
  "outputs": {
    "all_access_key_secrets": {
      "value": {
        "Andy": "xSH56b/f9r128TKyZkVHLqBx80+HCCJSA+bDB/1A",
        "Angela": "EQVFQJetu7FKIEdaVdNb72wZ8xdm73flaXug7dP",
        "Charles": "vNbQvIth16w3qsLVSq1SXWL8nLvJQIBEineNCise",
        "Clark": "bY8ye5Ya762nEcnuEK2Z8UmCrgGa2q6NuOExsTTz",
        "Creed": "LhNNTtlgCCIHWToBUP/w8Y7WRDRubMGzvDf3JOLiQ",
        "Darryl": "+TIBY0xlo4VJaA54JESfsqsMG+l68fun2EODLzR+",
        "David": "/NjUr5QyVoxBRUYc0yvJf+HCXYx+uLI1AU040mTb",
        "Dwight": "1Ca+g3uuSA+Pb7T3ubaEnDeEElkjTJ1G90eNgsJge",
        "Erin": "DPK9wN03vios1Un/Gu+MSoE7tXmhcpwsq1sSFqy",
        "Gabe": "omPpczXRMWqN0rTpifxYRsZ8u1/RYUXk4nkH/F5L",
        "Holly": "tsRO0ybcsVx4nD/5MXotSwNnhvIM/Uw3bwNZUNVq",
        "Jan": "K6SN+yFTygBK9qtvrBm9Iav2MhU0by2WTcv1bQQ2",
        "Jim": "Xe9oyCaW+ebleUEqD3cDZC4NXEM3aMQMcyDt/Hbx",
        "Jo": "YVe4Ljr2hkFx6dmjxtzkxj2pshkF4jwgmk0vdAOA",
        "Kelly": "iaQ3HM1RZkyIFRKByNucoGiL2dqeqBmFN0oVJGcT",
        "Kevin": "Aedm3PKPd1EBQGe1C5kgRrUyRF1oTvyl0B1z4vp",
        "Meredith": "a61ZiBMxF20gMeI2Z1C50ThLur6UMXUt8AxRI7lh",
        "Michael": "PV5sO2R+oP8B190enn0uShc3HyNGkQupmpUDq0gn",
        "Oscar": "z3VtCwkSVtO3k23XZA5hmt0gZbsgWiuboY5Xmk55",
        "Pam": "Miw9nmV1XmMUfxQxwmmQc6FwH+XcwIFINCgs4P05",
        "Peter": "YkNKnpcolu4kk04UR2iv25ZGIQ6SlgyswQHGN1lw",
        "Phyllis": "f86kJun03jR0hpYu+0Femj0V/4Q8/DT9+FBF2sG5",
        "Robert": "vxYOK0beJ7hP9mGHTdXpZXOUS/9JazR+K2ROCR5a",
        "Ryan": "10pFHRRparpOvgveJUG4fxtvWGgiPRvL54pdLmio".
      }
    }
}
```

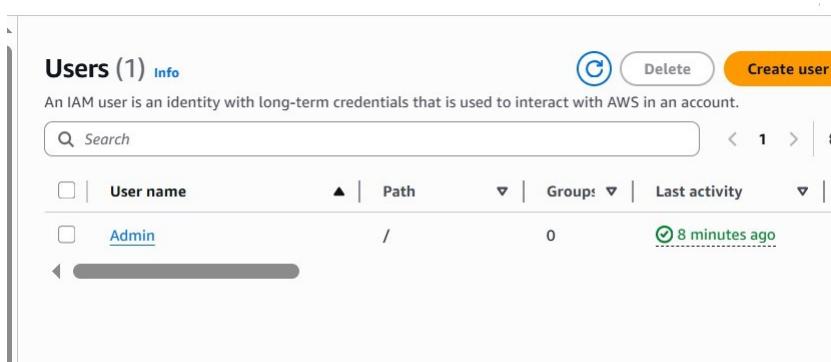
Cleanup

- cleanup_destroy_complete.png

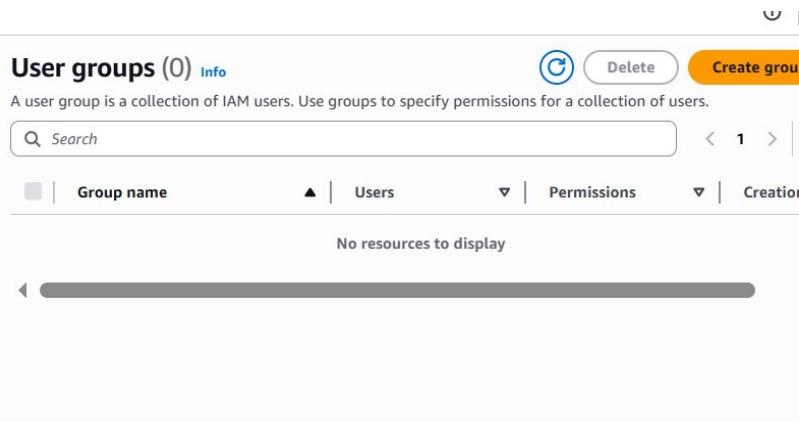
```
aws_iam_user.users["Clark"].Destroying... [id=Clark]
aws_iam_user.users["Jan"]: Destruction complete after 8s
aws_iam_user.users["Andy"]: Destroying... [id=Andy]
aws_iam_user.users["Jo"]: Destruction complete after 2s
aws_iam_user.users["Gabe"]: Destruction complete after 8s
aws_iam_user.users["David"]: Destruction complete after 3s
aws_iam_user.users["Creed"]: Destruction complete after 4s
aws_iam_user.users["Michael"]: Destruction complete after 5s
aws_iam_user.users["Holly"]: Destruction complete after 4s
aws_iam_user.users["Andy"]: Destruction complete after 2s
aws_iam_user.users["Oscar"]: Destruction complete after 5s
aws_iam_user.users["Pam"]: Destruction complete after 4s
aws_iam_user.users["Clark"]: Destruction complete after 5s

Destroy complete! Resources: 107 destroyed.
```

- cleanup_aws_console_users_deleted.png



- cleanup_aws_console_group_deleted.png



- cleanup_s3_empty_state.png

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Error>
<Code>AccessDenied</Code>
<Message>Access Denied</Message>
<RequestId>F3JH1VJEQ1ZCP910</RequestId>
<HostId>XpGVya1jGgR52SDDhZi6HTAbco2VmEVZ6u5pTQbk080MB525CPjZM9E3iuJbj7/9ACP7c2fN+A3oas4z66vSgI+N9+fTIaES</HostId>
</Error>
```

- cleanup_final_files.png

```
@NayabKhazin653  /workspaces/CC_nayab_046/Lab_13 (main) $ ls -la
total 65324
drwxrwxrwx+  4 codespace root      4096 Jan 13 05:10 .
drwxrwxrwx+ 17 codespace root      4096 Jan 12 14:05 ..
drwxr-xr-x+  3 codespace codespace  4096 Jan 13 04:51 .terraform
-rw-r--r--  1 codespace codespace  2422 Jan 13 04:20 .terraform.lock.hcl
-rw-rw-rw-  1 codespace root       0 Jan 12 13:05 README.md
drwxr-xr-x+  3 codespace codespace  4096 Jan  9 19:14 aws
-rw-rw-rw-  1 codespace codespace 66842323 Jan 12 13:21 awscliv2.zip
-rwxrwxrwx  1 codespace codespace   423 Jan 13 04:17 create-login-profile.sh
-rw-rw-rw-  1 codespace codespace    50 Jan 13 05:00 locals.tf
-rw-rw-rw-  1 codespace codespace  2509 Jan 13 05:10 main.tf
-rw-rw-rw-  1 codespace codespace      0 Jan 13 04:51 terraform.tfstate
-rw-rw-rw-  1 codespace codespace  6882 Jan 13 04:51 terraform.tfstate.backup
-rw-rw-rw-  1 codespace codespace   167 Jan 13 05:02 users.csv
-rw-rw-rw-  1 codespace codespace   150 Jan 13 04:15 variables.tf
```