



**CLOUD COMPUTING LAB**  
**BSE ( V-B )**

**LAB 13**

**Submitted By:** Urwa Zahra

**Roll No:** 2023-BSE-068

**Submitted To:** Sir Muhammad Shoaib

## Task 0 Lab Setup (Codespace & GH CLI)

```
@Urwa012 → /workspaces/cc-urwazahra-2023-BSE-068 (main) $ cd Lab11
@Urwa012 → /workspaces/cc-urwazahra-2023-BSE-068/Lab11 (main) $ gh codespace list
NAME          DISPLAY NAME      REPOSITORY      BRANCH STATE      CREATED AT
studious-guide-r45rj7gx6699f56wg  studious guide  Urwa012/cc-urwazahra-2...  main*  Shutdown  about 2 days ago
turbo-space-yodel-q759wvx4gj...   turbo space yodel  Urwa012/cc-urwazahra-2...  main  Shutdown  about 1 day ago
bookish-goldfish-pj5gx7r6w6pg...  bookish goldfish  Urwa012/cc-urwazahra-2...  main  Available  about 3 minutes ago
o

@Urwa012 → /workspaces/cc-urwazahra-2023-BSE-068/Lab11 (main) $ gh codespace ssh -c bookish-goldfish-pj5gx7r6w6pgc795
W
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.

@Urwa012 → /workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

## Task 1 — Create IAM Group and Output Details

```
@Urwa012 → /workspaces/cc-urwazahra-2023-BSE-068 (main) $ mkdir -p ~/Lab13
@Urwa012 → /workspaces/cc-urwazahra-2023-BSE-068 (main) $ cd ~/Lab13
@Urwa012 → ~/Lab13 $ touch 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
  }
}
```

```
► @Urwa012 →~/Lab13 $ terraform init
  Initializing the backend...
  Initializing provider plugins...
    - Reusing previous version of hashicorp/aws from the dependency lock file
    - Using previously-installed hashicorp/aws v6.27.0
```

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, other commands will detect it and remind you to do so if necessary.

```
► @Urwa012 →~/Lab13 $ terraform apply -auto-approve
```

commands will detect it and remind you to do so if necessary.

```
● @Urwa012 →~/Lab13 $ terraform apply -auto-approve
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

+ create

Terraform will perform the following actions:

```
# aws_iam_group.developers will be created
+ resource "aws_iam_group" "developers" {
  + arn      = (known after apply)
  + id       = (known after apply)
  + name     = "developers"
  + path     = "/groups/"
  + unique_id = (known after apply)
}
```

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 1s [id=developers]
```

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

Outputs:

```
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAWOW3ND6CDXUOKYFRS"
}
```

```
}
```

@Urwa012 ~/Lab13 \$ terraform output

```
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAWOW3ND6CDXUOKYFRS"
}
```

@Urwa012 ~/Lab13 \$

<input type="checkbox"/>	Group name	▲	Users	▼
<input type="checkbox"/>	<a href="#">admin</a>		1	
<input type="checkbox"/>	<a href="#">developers</a>			⚠ 0

## Task 2 — Create IAM User with Group Membership

```
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
  }
}

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
  }
}
```

```

@Urwa012 ~/Lab13 $ terraform apply -auto-approve
+ tags_all      = {
+   + "DisplayName" = "Load Balancer"
}
+ unique_id     = (known after apply)
}

# aws_iam_user_group_membership.lb_membership will be created
+ resource "aws_iam_user_group_membership" "lb_membership" {
+   groups = [
+     + "developers",
]
+   id     = (known after apply)
+   user   = "loadbalancer"
}

Plan: 2 to add, 0 to change, 0 to destroy.

Changes to Outputs:
+ user_details = {
+   + unique_id = (known after apply)
+   + user_arn  = (known after apply)
+   + user_name = "loadbalancer"
}
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 [id=AGPAWOW3ND6CDXUOKYFRS]

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

Outputs:

group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAWOW3ND6CDXUOKYFRS"
}
user_details = {
  "unique_id" = "AIDAWOW3ND6CPYXAZ2SBZ"
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}

}

● @Urwa012 ~/Lab13 $ terraform output
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAWOW3ND6CDXUOKYFRS"
}
user_details = {
  "unique_id" = "AIDAWOW3ND6CPYXAZ2SBZ"
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}

```

## loadbalancer [Info](#)



### Summary

#### ARN

arn:aws:iam::443915509636:user/users/loadbala

#### Console access

Disabled

#### Access key 1

[Create access key](#)

#### Created

January 06, 2026, 02:11 (UTC+05:00)

## developers [Info](#)

[Delete](#)

### Summary

[Edit](#)

#### User group name

developers

#### Creation time

January 06, 2026, 02:07 (UTC+05:00)

#### ARN

arn:aws:iam::443915509636:group/groups/developers

[Users \(1\)](#)

[Permissions](#)

[Access Advisor](#)

[Users in this group \(1\)](#)

## Task 3 — Attach Policies to IAM Group

```

    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
    }
}

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
    }
}

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"
}

```

```

@Urwa012 ~/Lab13 $ terraform apply -auto-approve
aws_iam_user.lb: Refreshing state... [id=loadbalancer]
aws_iam_user_group_membership.lb_membership: Refreshing state... [id=terraform-20260105211133401200000001]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the
following symbols:
+ create

Terraform will perform the following actions:

# aws_iam_group_policy_attachment.change_password will be created
+ resource "aws_iam_group_policy_attachment" "change_password" {
    + group      = "developers"
    + id         = (known after apply)
    + policy_arn = "arn:aws:iam::aws:policy/IAMUserChangePassword"
}

# aws_iam_group_policy_attachment.developer_ec2_fullaccess will be created
+ resource "aws_iam_group_policy_attachment" "developer_ec2_fullaccess" {
    + group      = "developers"
    + id         = (known after apply)
    + policy_arn = "arn:aws:iam::aws:policy/AmazonEC2FullAccess"
}

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.change_password: Creation complete after 1s [id=developers-20260105211538735400000001]
aws_iam_group_policy_attachment.developer_ec2_fullaccess: Creation complete after 1s [id=developers-20260105211538751600000002]

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

Outputs:
```

**developers** [Info](#) [Delete](#)

Summary		<a href="#">Edit</a>
User group name developers	Creation time January 06, 2026, 02:07 (UTC+05:00)	ARN <a href="#">arn:aws:iam:443915509636:group/groups/developers</a>

[Users \(1\)](#) [Permissions](#) [Access Advisor](#)

### Permissions policies (2) [Info](#)

You can attach up to 10 managed policies.

Filter by Type		
<input type="text"/> Search	All types	
<input type="checkbox"/> Policy name <a href="#">L</a>	<a href="#">▲</a> Type	<a href="#">▼</a> Attached entities
<input type="checkbox"/> <a href="#">AmazonEC2FullAccess</a>	AWS managed	1
<input type="checkbox"/> <a href="#">IAMUserChangePassword</a>	AWS managed	3

## Task 4 — Create Login Profile for IAM User

```
variable "iam_password" {
  description = "Temporary password for the IAM user"
  type        = string
  sensitive   = true
  default     = "1dontKnow"
}

#!/usr/bin/env bash
set -euo 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
~
```

- @Urwa012 →~/Lab13 \$ vim create-login-profile.sh
- @Urwa012 →~/Lab13 \$ chmod +x create-login-profile.sh
- @Urwa012 →~/Lab13 \$ █

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

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}'"
  }
}
```

```

commands will detect it and remind you to do so if necessary.
● @Urwa012 →~/Lab13 $ terraform apply -auto-approve -var="iam_password=MySecurePass123!"
aws_iam_group.developers: Refreshing state... [id=developers]
aws_iam_user.lb: Refreshing state... [id=loadbalancer]
aws_iam_group_policy_attachment.change_password: Refreshing state... [id=developers-20260105211538735400000001]
aws_iam_user_group_membership.lb_membership: Refreshing state... [id=terraform-2026010521133401200000001]
aws_iam_group_policy_attachment.developer_ec2_fullaccess: Refreshing state... [id=developers-20260105211538751600000002]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following s
+ create

Terraform will perform the following actions:

# null_resource.create_login_profile will be created
+ resource "null_resource" "create_login_profile" {
    + id      = (known after apply)
    + triggers = {
        + "password_hash" = (sensitive value)
        + "user"          = "loadbalancer"
    }
}

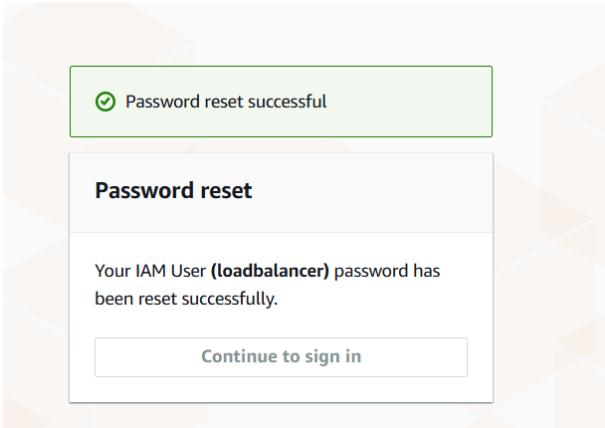
Plan: 1 to add, 0 to change, 0 to destroy.
null_resource.create_login_profile: Creating...
null_resource.create_login_profile: Provisioning with 'local-exec'...
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): Creation complete after 4s [id=3996121946308949874]

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

Outputs:

group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAWOW3ND6CDXUOKYFRS"
}
user_details = {
  "unique_id" = "AIDAWOW3ND6CPYXAZ2SBZ"
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}
}
● @Urwa012 →~/Lab13 $ aws iam get-login-profile --user-name loadbalancer
{
  "LoginProfile": {
    "UserName": "loadbalancer",
    "CreateDate": "2026-01-05T21:21:06+00:00",
    "PasswordResetRequired": true
  }
}

```



## Task 5 — Generate Access Keys for IAM User

```
}

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
}
-- INSERT --
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=AKIAWOW3ND6CEA4YPS26]

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

Outputs:

access_key_id = 'AKIAWOW3ND6CEA4YPS26'
access_key_secret = <sensitive>
group_details = {
    "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
    "group_name" = "developers"
    "unique_id" = "AGPDAWV...VVFP$"
}
user_details = {
    "unique_id" = "...-...-...-...-8Z"
    "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
    "user_name" = "loadbalancer"
}
@Urwa012 ~/Lab13 $
```

```

}
● @Urwa012 →~/Lab13 $ terraform output
access_key_id = "AKIAWV...26"
access_key_secret = <sensitive>
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AC...YFRS"
}
user_details = {
  "unique_id" = "ATP...SBZ"
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}
}
● @Urwa012 →~/Lab13 $ cat terraform.tfstate | grep -A 10 "access_key_secret"
"access_key_secret": {
  "value": "1...8F1KJURguo33eI219y1+9s4YuAqt4",
  "type": "string",
  "sensitive": true
},
"group_details": {
  "value": {
    "group_arn": "arn:aws:iam::443915509636:group/groups/developers",
    "group_name": "developers"
    "unique_id": "AC...YFRS"
  },
}
○ @Urwa012 →~/Lab13 $

```

**loadbalancer** [Info](#) [Delete](#)

Summary		Console access	Access key 1
ARN	<a href="#">arn:aws:iam::443915509636:user/users/loadbalancer</a>	<span style="color: red;">⚠ Enabled without MFA</span>	AKIAWV...26 - Active Never used. Created today.
Created	January 06, 2026, 02:11 (UTC+05:00)	Last console sign-in <a href="#">Today</a>	Access key 2 <a href="#">Create access key</a>
Permissions	Groups (1)	Tags (1)	<a href="#">Security credentials</a> <a href="#">Last Accessed</a>

**Console sign-in** [Manage console access](#)

Console sign-in link <a href="https://443915509636.signin.aws.amazon.com/console">https://443915509636.signin.aws.amazon.com/console</a>	Console password Updated 8 minutes ago (2026-01-06 02:28 GMT+5)
Last console sign-in <a href="#">View details</a>	

Amazon S3 > Buckets > Create bucket

## Create bucket Info

Buckets are containers for data stored in S3.

### General configuration

AWS Region  
Canada (Central) ca-central-1

Bucket name Info  
 Bucket names must be 3 to 63 characters and unique within the global namespace. Bucket names must also begin and end with a letter or number. Valid characters are a-z, 0-9, periods (.), and hyphens (-). [Learn more](#)

Copy settings from existing bucket - optional  
Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

Format: s3://bucket/prefix

### Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

Object Ownership

ACLs disabled (recommended)  
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

ACLs enabled  
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership  
Bucket owner enforced

### Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is disabled, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your use case does not require public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use case.

Block all public access  
Turning this setting on is the same as turning on all four settings below. Each of the following settings is independent of one another.

- Block public access to buckets and objects granted through **new** access control lists (ACLs)  
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing resources using ACLs.
- Block public access to buckets and objects granted through **any** access control lists (ACLs)  
S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through **new** public bucket or access point policies  
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through **any** public bucket or access point policies  
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

### Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your bucket. A bucket with versioning enabled can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning

Disable

Enable

### Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your bucket. A bucket with versioning enabled can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning

Disable

Enable

```

terraform {
  backend "s3" {
    bucket = "myapp-s3-bucket-demonstration"
    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 = {

```

```

● @Urwa012 ~/Lab13 $ 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 v6.27.0
- Using previously-installed hashicorp/null v3.2.4

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, other
commands will detect it and remind you to do so if necessary.
○ @Urwa012 ~/Lab13 $ [REDACTED]
spaces: bookish goldfish  ↵ main ⌂  ⊗ 0 ▲ 0  ↵ 2

```

```
● @Urwa012 →~/Lab13 $ terraform apply -auto-approve -var="iam_password=MySecurePass123!"  
aws_iam_group.developers: Refreshing state... [id=developers]  
aws_iam_user.lb: Refreshing state... [id=loadbalancer]  
aws_iam_group_policy_attachment.developer_ec2_fullaccess: Refreshing state... [id=developers-20260105211538751600000002]  
aws_iam_group_policy_attachment.change_password: Refreshing state... [id=developers-20260105211538735400000001]  
aws_iam_access_key.lb_access_key: Refreshing state... [id=AKIAWOW3ND6CEA4YPS26]  
null_resource.create_login_profile: Refreshing state... [id=3996121946308949874]  
aws_iam_user_group_membership.lb_membership: Refreshing state... [id=terraform-202601052113340120000001]
```

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.

#### Outputs:

```
access_key_id = "AKIAWOW3ND6CEA4YPS26"  
access_key_secret = <sensitive>  
group_details = {  
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"  
  "group_name" = "developers"  
  "unique_id" = "A...T...C...P...K...Y...R...S"  
}  
user_details = {  
  "unique_id" = "A...T...C...P...K...Y...R...Z"  
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"  
  "user_name" = "loadbalancer"  
}
```

```
○ @Urwa012 →~/Lab13 $
```

> myapp-s3-bucket-demonstration > myapp/

myapp/

Objects Properties

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For

Find objects by prefix

Name	Type	Last modified	Size	Storage class
terraform.tfstate	tfstate	January 6, 2026, 02:46:05 (UTC+05:00)	6.7 KB	Standard

```
● @Urwa012 →~/Lab13 $ ls -la terraform.tfstate*  
-rw-r--r-- 1 codespace codespace 0 Jan  5 21:46 terraform.tfstate  
-rw-r--r-- 1 codespace codespace 6882 Jan  5 21:46 terraform.tfstate.backup  
○ @Urwa012 →~/Lab13 $ terraform destroy -auto-approve
```

```

@Urwa012 ➔~/Lab13 $ terraform destroy -auto-approve
    } -> null
- unique_id          = "AIDAWOW3ND6CPYXAZ2SBZ" -> null
# (1 unchanged attribute hidden)
}

# aws_iam_user_group_membership.lb_membership will be destroyed
- resource "aws_iam_user_group_membership" "lb_membership" {
    - groups = [
        - "developers",
    ] -> null
- id     = "terraform-20260105211133401200000001" -> null
- user   = "loadbalancer" -> null
}

# null_resource.create_login_profile will be destroyed
- resource "null_resource" "create_login_profile" {
    - id      = "3996121946308949874" -> null
    - triggers = {
        - "password_hash" = (sensitive value)
        - "user"           = "loadbalancer"
    } -> null
}

Plan: 0 to add, 0 to change, 7 to destroy.

Changes to Outputs:
- access_key_id      = "AKIAWOW3ND6CEA4YPS26" -> null
- access_key_secret = (sensitive value) -> null
- group_details     = {
    - group_arn  = "arn:aws:iam::443915509636:group/groups/developers"
    - group_name = "developers"
    - unique_id  = "AGPAWOW3ND6CDXUOKYFRS"
} -> null
- user_details       = {
    - unique_id = "AIDAWOW3ND6CPYXAZ2SBZ"
    - user_arn   = "arn:aws:iam::443915509636:user/users/loadbalancer"
    - user_name  = "loadbalancer"
} -> null
null_resource.create_login_profile: Destroying... [id=3996121946308949874]
null_resource.create_login_profile: Destruction complete after 0s
aws_iam_group_policy_attachment.developer_ec2_fullaccess: Destroying... [id=developers-20260105211538735400]
aws_iam_access_key.lb_access_key: Destroying... [id=AKIAWOW3ND6CEA4YPS26]
aws_iam_group_policy_attachment.change_password: Destroying... [id=developers-20260105211133401200000000]
aws_iam_user_group_membership.lb_membership: Destroying... [id=terraform-20260105211133401200000001]
aws_iam_user_group_membership.lb_membership: Destruction complete after 1s
aws_iam_group_policy_attachment.change_password: Destruction complete after 1s
aws_iam_access_key.lb_access_key: Destruction complete after 1s
aws_iam_user.lb: Destroying... [id=loadbalancer]
aws_iam_group_policy_attachment.developer_ec2_fullaccess: Destruction complete after 1s
aws_iam_group.developers: Destroying... [id=developers]
aws_iam_group.developers: Destruction complete after 0s
aws_iam_user.lb: Destruction complete after 2s

Destroy complete! Resources: 7 destroyed.

```

Properties	Permissions	Versions			
<b>Versions (1)</b> <span style="float: right;"> <a href="#">Download</a> <a href="#">Open</a> <a href="#">Delete</a> <a href="#">Actions ▾</a> </span>					
	Version ID	Type	Last modified	Size	Storage class
<input type="checkbox"/>	null (Current version)	tfstate	January 6, 2026...	181.0 B	Standard

## Task 7 — Create Multiple Users from CSV File

The screenshot shows two terminal windows side-by-side. The left terminal window has tabs: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is selected), and PORTS. It contains the following Python code:

```
locals()
users = csvdecode(file("users.csv"))
~
```

The right terminal window also has tabs: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (selected), and PORTS. It displays a list of user names:

```
user_name
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
~
```

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS 2

        CreatedBy  = "Terraform"
    }
}

# Add all users to developers group
resource "aws_iam_user_group_membership" "users_membership" {
    for_each = aws_iam_user.users

    user = each.value.name
    groups = [
        aws_iam_group.developers.name
    ]
}

# Create login profiles for all users
resource "null_resource" "create_login_profiles" {
    for_each = aws_iam_user.users

    triggers = {
        password_hash = sha256(var.iam_password)
        user          = each.value.name
    }

    depends_on = [aws_iam_user.users]

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

# Create access keys for all users
resource "aws_iam_access_key" "users_access_keys" {
    for_each = aws_iam_user.users

    user = each.value.name
}

# Output all user details
output "all_users_details" {
    value = {
        for user_name, user in aws_iam_user.users : user_name => {
            user_arn      = user.arn
            user_unique_id = user.unique_id
            access_key_id = aws_iam_access_key.users_access_keys[user_name].id
        }
    }
}

# Output all access key secrets (sensitive)
output "all_access_key_secrets" {
    value = {
        for user_name, key in aws_iam_access_key.users_access_keys : user_name => key.secret
    }
    sensitive = true
}
```

```
● @Urwa012 →~/Lab13 $ 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 v6.27.0
```

```
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, other
commands will detect it and remind you to do so if necessary.
```

```
@Urwa012 →~/Lab13 $ terraform apply -auto-approve -var="iam_password=MySecurePass123!"
null_resource.create_login_profiles["Darryl"]: Creation complete after 7s [id=8876531924476751998]
```

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

#### Outputs:

```
access_key_id = "AKIAWOW3ND6CHOZNUGE"
access_key_secret = <sensitive>
all_access_key_secrets = <sensitive>
all_users_details = {
  "Andy" = {
    "access_key_id" = "AKIAWOW3ND6CGHJ6QMHK"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Andy"
    "user_unique_id" = "AIDAWOW3ND6CLHNCXNK6V"
  }
  "Angela" = {
    "access_key_id" = "AKIAWOW3ND6CC7GBZWOF"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Angela"
    "user_unique_id" = "AIDAWOW3ND6CP3EW2Q0W4"
  }
  "Charles" = {
    "access_key_id" = "AKIAWOW3ND6COQFF4AJY"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Charles"
    "user_unique_id" = "AIDAWOW3ND6CKWDBSMPLK"
  }
  "Clark" = {
    "access_key_id" = "AKIAWOW3ND6CBSYF6NMA"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Clark"
    "user_unique_id" = "AIDAWOW3ND6CNVF6A3YF4"
  }
  "Creed" = {
    "access_key_id" = "AKIAWOW3ND6CCOF2L6ED"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Creed"
    "user_unique_id" = "AIDAWOW3ND6CDX5J45HZG"
  }
  "Darryl" = {
    "access_key_id" = "AKIAWOW3ND6CBHE66AJG"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Darryl"
    "user_unique_id" = "AIDAWOW3ND6CMFTNMXQ46"
  }
  "David" = {
    "access_key_id" = "AKIAWOW3ND6CNDK6MJVM"
    "user_arn" = "arn:aws:iam::443915509636:user/users/David"
    "user_unique_id" = "AIDAWOW3ND6CAGVVYYUH"
  }
  "Dwight" = {
    "access_key_id" = "AKIAWOW3ND6CFCSVF5XY"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Dwight"
    "user_unique_id" = "AIDAWOW3ND6CA2CFRUSHV"
  }
  "Erin" = {
    "access_key_id" = "AKIAWOW3ND6CPAM63A4H"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Erin"
    "user_unique_id" = "AIDAWOW3ND6CNUI5VRGP"
  }
```

```
● @Urwa012 → ~/Lab13 $ terraform output
access_key_id = "AKIAWOW3ND6CGHJ6QMHK"
access_key_secret = <sensitive>
all_access_key_secrets = <sensitive>
all_users_details = {
    "Andy" = {
        "access_key_id" = "AKIAWOW3ND6CC7GBZWF"
        "user_arn" = "arn:aws:iam::443915509636:user/users/Andy"
        "user_unique_id" = "AIDAWOW3ND6CLHNCXNK6V"
    }
    "Angela" = {
        "access_key_id" = "AKIAWOW3ND6CC7GBZWF"
        "user_arn" = "arn:aws:iam::443915509636:user/users/Angela"
        "user_unique_id" = "AIDAWOW3ND6CP3EW2QDW4"
    }
    "Charles" = {
        "access_key_id" = "AKIAWOW3ND6CQQFF4AJY"
        "user_arn" = "arn:aws:iam::443915509636:user/users/Charles"
        "user_unique_id" = "AIDAWOW3ND6CKWDBSMPLK"
    }
    "Clark" = {
        "access_key_id" = "AKIAWOW3ND6CBSYF6NMA"
        "user_arn" = "arn:aws:iam::443915509636:user/users/Clark"
        "user_unique_id" = "AIDAWOW3ND6CNVF6A3YF4"
    }
    "Creed" = {
        "access_key_id" = "AKIAWOW3ND6CCOF2L6ED"
        "user_arn" = "arn:aws:iam::443915509636:user/users/Creed"
        "user_unique_id" = "AIDAWOW3ND6CDX5J45HZG"
    }
    "Darryl" = {
        "access_key_id" = "AKIAWOW3ND6CBHE66AJG"
        "user_arn" = "arn:aws:iam::443915509636:user/users/Darryl"
        "user_unique_id" = "AIDAWOW3ND6CMFTNMXQ46"
    }
    "David" = {
        "access_key_id" = "AKIAWOW3ND6CNDK6MJVM"
        "user_arn" = "arn:aws:iam::443915509636:user/users/David"
        "user_unique_id" = "AIDAWOW3ND6CAGYVVVUH"
    }
    "Dwight" = {
        "access_key_id" = "AKIAWOW3ND6CFCSVF5XY"
        "user_arn" = "arn:aws:iam::443915509636:user/users/Dwight"
        "user_unique_id" = "AIDAWOW3ND6CA2CFRUSHV"
    }
    "Erin" = {
        "access_key_id" = "AKIAWOW3ND6CPAM63A4H"
        "user_arn" = "arn:aws:iam::443915509636:user/users/Erin"
        "user_unique_id" = "AIDAWOW3ND6CNUI5VRRGP"
    }
    "Gabe" = <redacted>
```

	Admin	/	0	4 days ago	-	4 days	4 days ago	-	-	arnaws:iam::443915509636:user/Admin	4 days ago	Enabled	-		
<input type="checkbox"/>	<a href="#">Andy</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Angela</a>	/users/	1	-	-	Yesterday	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">assignment</a>	/	1	-	-	11 days	-	-	Active - ANIAKROW3ND...	11 days	Yesterday	arnaws:iam::443915509636:user/assig...	11 days ago	Enabled	-
<input type="checkbox"/>	<a href="#">assignment2</a>	/	0	-	-	12 days	-	-	Active - ANIAKROW3ND...	11 days	-	arnaws:iam::443915509636:user/assig...	12 days ago	Enabled	-
<input type="checkbox"/>	<a href="#">CC</a>	/	0	-	-	-	-	-	Active - ANIAKROW3ND...	11 days	-	arnaws:iam::443915509636:user/CC...	11 days ago	Disabled	-
<input type="checkbox"/>	<a href="#">Charles</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Chris</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Cress</a>	/users/	1	-	-	3 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">David</a>	/users/	1	-	-	2 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">David</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Dwight</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Eric</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Gale</a>	/users/	1	-	-	2 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Holy</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Jan</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Jim</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Jo</a>	/users/	1	-	-	2 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Kelly</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Kenny</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Latifah</a>	/	0	4 days ago	-	4 days	4 days ago	-	-	-	-	arnaws:iam::443915509636:user/Latif...	4 days ago	Enabled	-
<input type="checkbox"/>	<a href="#">Leothar</a>	/users/	1	-	-	8 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Meredith</a>	/users/	1	-	-	3 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Michael</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Oscar</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Pam</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Peter</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Phyllis</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Robert</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Ryan</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Stanley</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Toby</a>	/users/	1	-	-	5 minutes	-	-	Active - ANIAKROW3ND...	8 minutes	-	arnaws:iam::443915509636:user/user...	8 minutes ago	Enabled	-
<input type="checkbox"/>	<a href="#">Ursula</a>	/	0	6 minutes ago	-	-	-	-	Active - ANIAKROW3ND...	2 days	6 minutes ago	arnaws:iam::443915509636:user/Urso...	11 days ago	Disabled	-

## Users in this group (27)

 Remove

An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.

Search

<input type="checkbox"/>	User name	Groups	Last activity	Creation time
<input type="checkbox"/>	<a href="#">Andy</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Angela</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Charles</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Clark</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Creed</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Darryl</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">David</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Dwight</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Erin</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Gabe</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Holly</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Jan</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Jim</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Jo</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Kelly</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Kevin</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">loadbalancer</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Meredith</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Michael</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Oscar</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Pam</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Peter</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Phyllis</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Robert</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Ryan</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Stanley</a>		None	10 minutes ago
<input type="checkbox"/>	<a href="#">Toby</a>		None	10 minutes ago

## Michael Info

[Delete](#)

### Summary

**ARN**[am:aws:iam::443915509636:user/users/Michael](#)**Created**

January 06, 2026, 02:55 (UTC+05:00)

**Console access**[Enabled without MFA](#)**Last console sign-in**[Never](#)**Access key 1**

AKIAWOW3ND6CA3F24SN4 - Active

[Never used. Created today.](#)**Access key 2**[Create access key](#)[Permissions](#)[Groups \(1\)](#)[Tags \(2\)](#)**Security credentials**[Last Accessed](#)

### Console sign-in

[Manage console access](#)**Console sign-in link**<https://443915509636.sigin.aws.amazon.com/console>**Console password**

Updated 5 minutes ago (2026-01-06 03:01 GMT+5)

**Last console sign-in**[Never](#)

### Multi-factor authentication (MFA) (0)

[Remove](#)[Resync](#)[Assign MFA device](#)

Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. [Learn more](#)

Type	Identifier	Certifications	Created on
------	------------	----------------	------------

No MFA devices. Assign an MFA device to improve the security of your AWS environment

[Assign MFA device](#)

### Access keys (1)

[Create access key](#)

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 or inactive) at a time. [Learn more](#)

**AKIAWOW3ND6CA3F24SN4**[Actions ▾](#)

```

        "type": "string"
    },
    "access_key_secret": {
        "value": "pvNw/9vCXfc10Yqm3+Nx/2aqXfGhi/C5CRJ2Cne+",
        "type": "string",
        "sensitive": true
    },
    "all_access_key_secrets": {
        "value": {
            "Andy": "NmrvAVcBSjV4fuE1kgbYrFFF/BHN243vidLBhJek1",
            "Angela": "ScBNJHPK5gzm08Qlw66h+NyOwXnmmmA1yQxXr",
            "Charles": "g0wLO1Juy1CkuvUVMx1vuSI2w8B5q7tJz1102GGR",
            "Clark": "p5QqbSXX4Y5Te4jqTtKs/dd9ms9RqnJvpbats2vj",
            "Creed": "2RdyDS3cS0AtVsD+IF7YcLKWecfDhDA6kRnq1Yze",
            "Darryl": "BfKFF1YBMybu0AP9xdmz4V19ELSrFM2Ny9W5oWk",
            "David": "Yyjw8VQW0PNW9fl1jAK1zG5qg/GENqt7Rc2y5/o",
            "Dwight": "mYkNanxM4h0k0qcccGBF9QhTg4/P7fh8CD0e+rri7",
            "Erin": "CGizONWrhed01FgoWraBnmxbax4W4rWC6p1G5fV",
            "Gabe": "@AdtNUCjWJ2TinDF35ruy0lXVszoOeGuIw4WkduB",
            "Holly": "Kvk13vvW/h2YTLgpj0WSb1mV/99x+2oY0vpXBdek",
            "Jan": "Mt48xThodv7XNlEwh93qgHk4y1D/a3VY7GnLPP44",
            "Jim": "UE1A69gUWTDbtbwogiBEgczT0pAU7wn3vjHK3Uex",
            "Jo": "sgczcZnd540WFQkthnoegeiKLP6QHXIHMKZH0h/",
            "Kelly": "BXFY3TfxD0yJcylqgguc3502o3FpFnX5CyKC6yae",
            "Kevin": "gqA2Uy6y7ZE6n+19d5iQpvsn617bbKxkr+a1l0+q",
            "Meredith": "jsfxlpVUiC10U909YZ/qFK4c7BRo/fAUOUKZXg7V",
            "Michael": "M/b+Y3rj0sGUyhckdo2gg3ZTVtD/wBQoThr7sA01",
            "Oscar": "1CjNUXLuF/8K7CYAYvgjSF7ukWgZjtGqVVUViWC",
            "Pam": "J0xJHSobznBIJTJrTOrx078GH7sjzf9s1qcq/0xj",
            "Peter": "NfqWq85U1WKjLIYdjtzv/0I10NcaN2gHa2MAj6Jr",
            "Phyllis": "qm/sallrS01dMzgeJKNw8r2TEILBg04A6c7RdACS",
            "Robert": "wmyql1mtNp9vrTAT7/8f2En4ssFIImmbOrm+WCsY",
            "Ryan": "FVpfTUVCXR9uYRa88djvIW4ntnBRjfZ8G7LfGlp1",
            "Stanley": "mopueFTbnGbyNOLj6G0x3mkn3amkm900nJNMEKO",
            "Toby": "c93hK2TzOsjghPpruZmNZ5j+ou4w/pgsReUTIJOm"
        }
    },
    "type": [
        "object",
        {
            "Andy": "string",
            "Angela": "string",
            "Charles": "string",
            "Clark": "string",
            "Creed": "string",
            "Darryl": "string",
            "David": "string",
            "Dwight": "string",
            "Erin": "string",
            "Gabe": "string",
            "Holly": "string",
            "Jan": "string",
            "Jim": "string",
            "Jo": "string",
            "Kelly": "string",
            "Kevin": "string",
            "Meredith": "string",
            "Michael": "string",
            "Oscar": "string",
            "Pam": "string",
            "Peter": "string",
            "Phyllis": "string",
            "Robert": "string",
            "Ryan": "string",
            "Stanley": "string",
            "Toby": "string"
        }
    ]
}

```

## Cleanup:

```
aws_iam_user.users["Stanley"]: Destruction complete after 5s
aws_iam_user.users["Charles"]: Destroying... [id=Charles]
aws_iam_user.users["Toby"]: Destruction complete after 5s
aws_iam_user.users["Robert"]: Destroying... [id=Robert]
aws_iam_user.users["Andy"]: Destruction complete after 2s
aws_iam_user.users["Kevin"]: Destroying... [id=Kevin]
aws_iam_user.users["Jan"]: Destruction complete after 2s
aws_iam_user.users["Erin"]: Destroying... [id=Erin]
aws_iam_user.users["Creed"]: Destruction complete after 2s
aws_iam_user.users["Phyllis"]: Destroying... [id=Phyllis]
aws_iam_user.users["Peter"]: Destruction complete after 2s
aws_iam_user.users["Jim"]: Destroying... [id=Jim]
aws_iam_user.users["David"]: Destruction complete after 4s
aws_iam_user.users["Gabe"]: Destroying... [id=Gabe]
aws_iam_user.users["Charles"]: Destruction complete after 2s
aws_iam_user.users["Kelly"]: Destroying... [id=Kelly]
aws_iam_user.users["Clark"]: Destruction complete after 4s
aws_iam_user.users["Ryan"]: Destroying... [id=Ryan]
aws_iam_user.users["Robert"]: Destruction complete after 2s
aws_iam_user.users["Darryl"]: Destruction complete after 7s
aws_iam_user.users["Kevin"]: Destruction complete after 2s
aws_iam_user.users["Erin"]: Destruction complete after 2s
aws_iam_user.users["Dwight"]: Destruction complete after 5s
aws_iam_user.users["Jim"]: Destruction complete after 3s
aws_iam_user.users["Kelly"]: Destruction complete after 2s
aws_iam_user.users["Ryan"]: Destruction complete after 3s
aws_iam_user.users["Gabe"]: Destruction complete after 4s
aws_iam_user.users["Phyllis"]: Destruction complete after 7s
```

```
Destroy complete! Resources: 111 destroyed.
```

```
@Urwa012 ~/Lab13 $
```

```
aws_iam_user.users["Stanley"]: Destruction complete after 5s
aws_iam_user.users["Charles"]: Destroying... [id=Charles]
aws_iam_user.users["Toby"]: Destruction complete after 5s
aws_iam_user.users["Robert"]: Destroying... [id=Robert]
aws_iam_user.users["Andy"]: Destruction complete after 2s
aws_iam_user.users["Kevin"]: Destroying... [id=Kevin]
aws_iam_user.users["Jan"]: Destruction complete after 2s
aws_iam_user.users["Erin"]: Destroying... [id=Erin]
aws_iam_user.users["Creed"]: Destruction complete after 2s
aws_iam_user.users["Phyllis"]: Destroying... [id=Phyllis]
aws_iam_user.users["Peter"]: Destruction complete after 2s
aws_iam_user.users["Jim"]: Destroying... [id=Jim]
aws_iam_user.users["David"]: Destruction complete after 4s
aws_iam_user.users["Gabe"]: Destroying... [id=Gabe]
aws_iam_user.users["Charles"]: Destruction complete after 2s
aws_iam_user.users["Kelly"]: Destroying... [id=Kelly]
aws_iam_user.users["Clark"]: Destruction complete after 4s
aws_iam_user.users["Ryan"]: Destroying... [id=Ryan]
aws_iam_user.users["Robert"]: Destruction complete after 2s
aws_iam_user.users["Darryl"]: Destruction complete after 7s
aws_iam_user.users["Kevin"]: Destruction complete after 2s
aws_iam_user.users["Erin"]: Destruction complete after 2s
aws_iam_user.users["Dwight"]: Destruction complete after 5s
aws_iam_user.users["Jim"]: Destruction complete after 3s
aws_iam_user.users["Kelly"]: Destruction complete after 2s
aws_iam_user.users["Ryan"]: Destruction complete after 3s
aws_iam_user.users["Gabe"]: Destruction complete after 4s
aws_iam_user.users["Phyllis"]: Destruction complete after 7s
```

```
Destroy complete! Resources: 111 destroyed.
```

```
@Urwa012 ~/Lab13 $
```

## Users (0) [Info](#)

[Delete](#)

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

 Search

<input type="checkbox"/>	User name	▲	Path	▼	Groups	▼	Last activity
--------------------------	-----------	---	------	---	--------	---	---------------

No resources to display

## User groups (0) [Info](#)

[Delete](#)

A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.

 Search

<input type="checkbox"/>	Group name	▲	Users	▼	Permissions
--------------------------	------------	---	-------	---	-------------

No resources to display

myapp-s3-bucket-demonstration [Info](#)

Objects    Metadata    Properties    Permissions    Metrics    Management    Access Points

**Objects (0)**

[Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix [Show versions](#)

Name [▲](#) Type [▼](#) Last modified [▼](#) Size [▼](#) Storage class [▼](#)

No objects  
You don't have any objects in this bucket.

[Upload](#)

```
Destroy complete! Resources: 111 destroyed.
● @Urwa012 → ~/Lab13 $ ls -la
total 61764
drwxr-xr-x 4 codespace codespace    4096 Jan  5 22:00 .
drwxr-x--- 1 codespace codespace    4096 Jan  5 22:00 ..
drwxr-xr-x 3 codespace codespace    4096 Jan  5 21:46 .terraform
-rw-r--r-- 1 codespace codespace   2422 Jan  5 21:20 .terraform.lock.hcl
drwxr-xr-x 3 codespace codespace    4096 Jan  5 19:11 aws
-rw-r--r-- 1 codespace codespace 63189840 Jan  5 20:57 awscliv2.zip
-rwxr-xr-x 1 codespace codespace     423 Jan  5 21:18 create-login-profile.sh
-rw-r--r-- 1 codespace codespace      50 Jan  5 21:52 locals.tf
-rw-r--r-- 1 codespace codespace   3566 Jan  5 22:00 main.tf
-rw-r--r-- 1 codespace codespace       0 Jan  5 21:46 terraform.tfstate
-rw-r--r-- 1 codespace codespace  6882 Jan  5 21:46 terraform.tfstate.backup
-rw-r--r-- 1 codespace codespace     167 Jan  5 21:53 users.csv
-rw-r--r-- 1 codespace codespace     150 Jan  5 21:18 variables.tf
```