EX NO: 1A CREATION OF CLOUD ORGANIZATION IN GCP
DATE:

AIM:

To Create a Cloud Organization in Google Cloud platform with Role-based access control.

PROCEDURE:

Google Cloud Platform (GCP), offered by Google, is a suite of cloud computing services that runs on the same infrastructure that Google uses internally for its end-user products, such as Google Search, Gmail, Google Drive, and YouTube Alongside a set of management tools, it provides a series of modular cloud services including computing, data storage, data analytics and machine learning. Registration requires a credit card or bank account details.

Google Cloud Platform provides infrastructure as a service, platform as a service, and server less computing environments.

In April 2008, Google announced App Engine, a platform for developing and hosting web applications in Google-managed data centers, which was the first cloud computing service from the company. The service became generally available in November 2011. Since the announcement of App Engine, Google added multiple cloud services to the platform.

Google Cloud Platform is a part of Google Cloud, which includes the Google Cloud Platform public cloud infrastructure, as well as Google Workspace (G Suite), enterprise versions of Android and ChromeOS, and application programming interfaces (APIs) for machine learning and enterprise mapping services.

Creating your project

To deploy your app on App Engine, you must create a Google Cloud project, which is a top level container that holds your App Engine application resources as well as other Google Cloud resources.
In this task, you create a Google Cloud project and an App Engine application to store settings, computing resources, credentials, and metadata for your app.
If you already have a Google Cloud project with App Engine and the Cloud Build API enabled, continue to Writing Your Web Service.

Creating a Google Cloud project

1. If you're new to Google Cloud, create an account to evaluate how our products perform in real-world scenarios. New customers get $300 in free credits to run, test, and deploy workloads.

2. In the Google Cloud console, on the project selector page, select or create a Google Cloud project.

Note: If you don't plan to keep the resources that you create in this procedure, create a project instead of selecting an existing project. After you finish these steps, you can delete the project, removing all resources associated with the project.

Go to project selector

3. Make sure that billing is enabled for your Google Cloud project.

4. Enable the Cloud Build API.
Enable the API

5. Install the Google Cloud CLI.

6. To initialize the gcloud CLI, run the following command:
gcloud init

7. Create an App Engine application for your Google Cloud project in the Google Cloud console.
Open app creation

8. Select a region where you want your app's computing resources located.
Note: After you create your App Engine app, you cannot change the region. To reduce latency, choose the region closest to your app's intended users. For more information on the available regions, see App Engine Locations.

Next step
Now that your Google Cloud project is set up, you're ready to write a basic web service with Node.js.
Setting up your development environment
bookmark_border
Go Java Node.js PHP Python Ruby

Use the following steps to set up your local environment for developing and deploying your App Engine services:

1. Install the latest release of Python 3.
See Python3 Runtime Environment for a list of the supported versions.

2. Install and initialize the gcloud CLI for deploying and managing your apps. If you already have the gcloud CLI installed and initialized, run the gcloud components update command to update to the latest release. By downloading, you agree to be bound by the Terms that govern use of the gcloud CLI for App Engine.

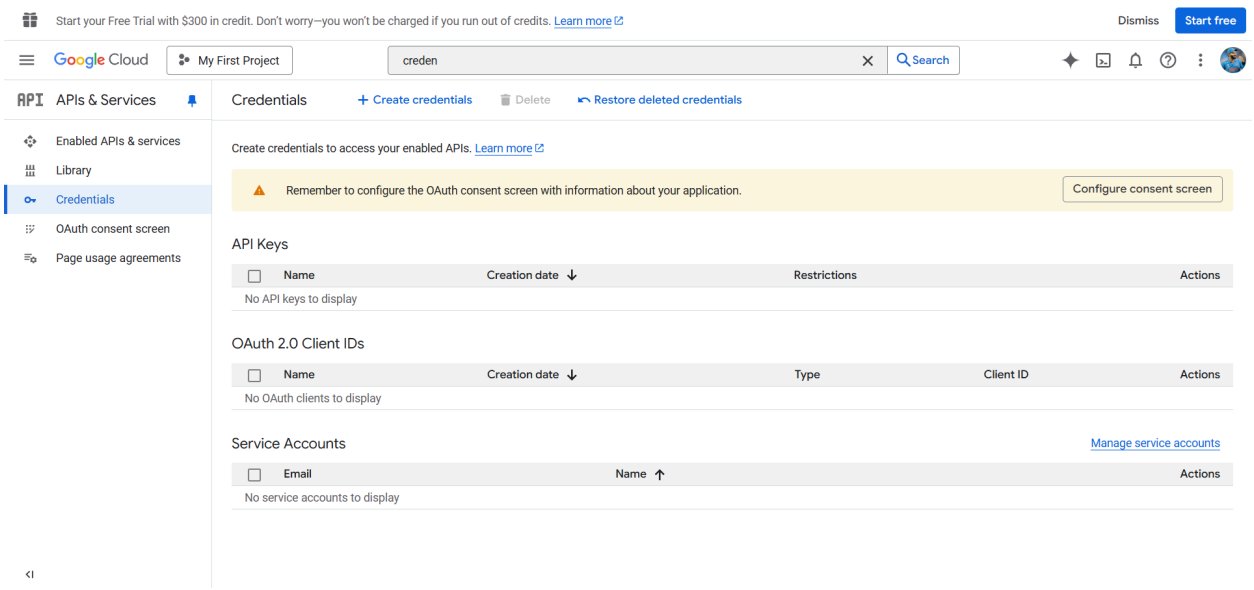
Optional tools:

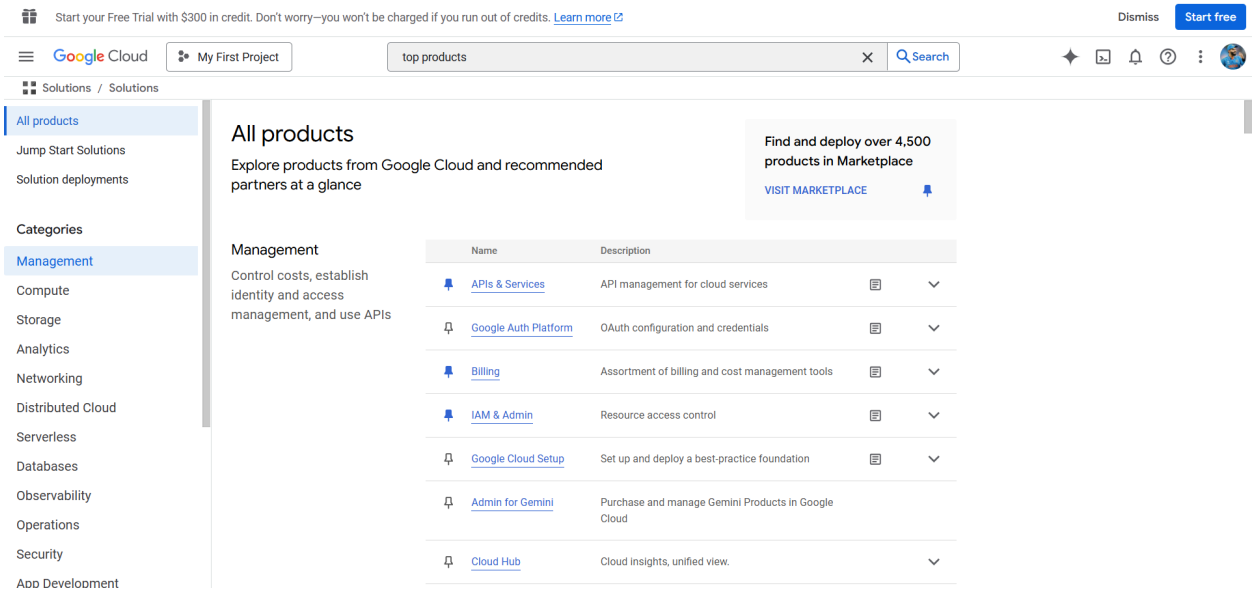
• Install Git for access to code, samples, libraries, and tools in the Google Cloud GitHub repository.

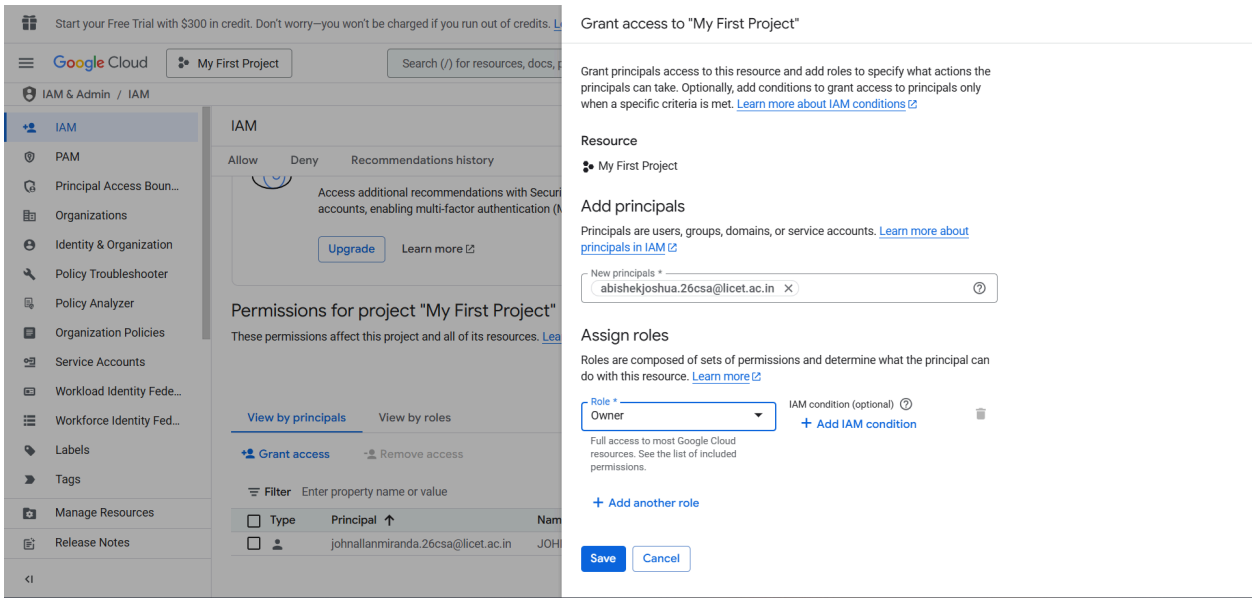
• Install your preferred tooling or framework, for example you can use any of the following frameworks

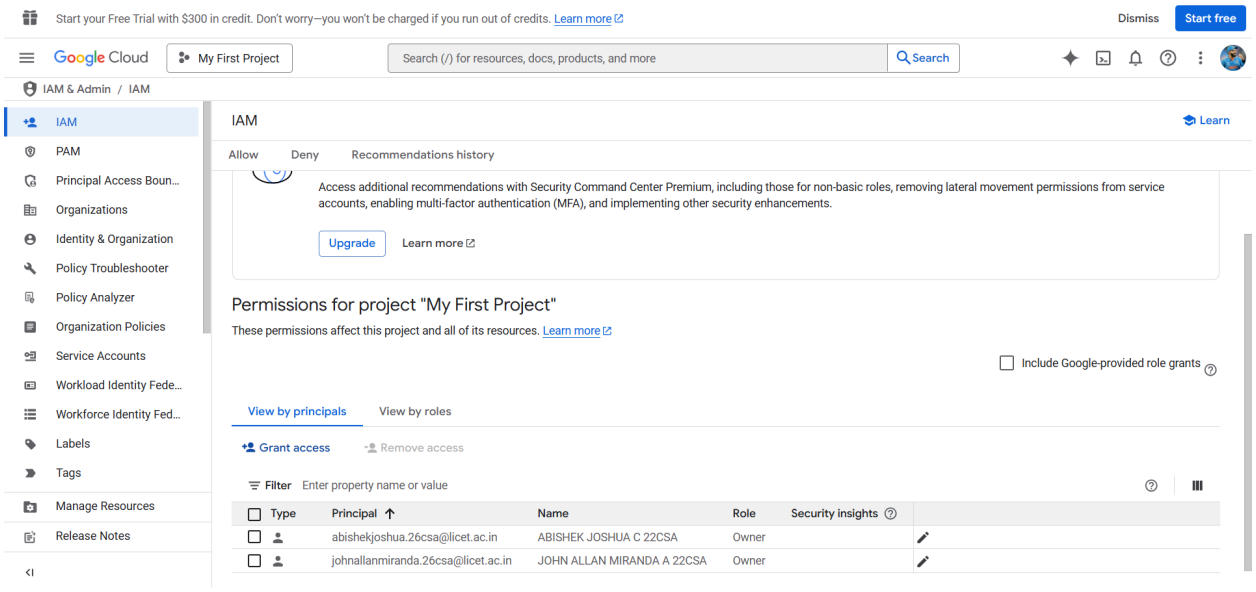
to developyour Python 3 app:

• Flask
• Django
• Pyramid
• Bottle
• web.py
• Tornado




Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits. Learn more
Dismiss
Start free
Google Cloud
My First Project
top products
Search
Solutions / Solutions
All products
Jump Start Solutions
Solution deployments
Categories
Management
Compute
Storage
Analytics
Networking
Distributed Cloud
Serverless
Databases
Observability
Operations
Security
App Development
All products
Explore products from Google Cloud and recommended partners at a glance
Find and deploy over 4,500 products in Marketplace
VISIT MARKETPLACE
Management
Control costs, establish identity and access management, and use APIs
Name
Description
APIs & Services
API management for cloud services
Google Auth Platform
OAuth configuration and credentials
Billing
Assortment of billing and cost management tools
IAM & Admin
Resource access control
Google Cloud Setup
Set up and deploy a best-practice foundation
Admin for Gemini
Purchase and manage Gemini Products in Google Cloud
Cloud Hub
Cloud insights, unified view.


Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits.
Google Cloud
My First Project
Search (/) for resources, docs,
IAM & Admin / IAM
IAM
PAM
Principal Access Boun...
Organizations
Identity & Organization
Policy Troubleshooter
Policy Analyzer
Organization Policies
Service Accounts
Workload Identity Fede...
Workforce Identity Fed...
Labels
Tags
Manage Resources
Release Notes
IAM
Allow
Deny
Recommendations history
Access additional recommendations with Securi
accounts, enabling multi-factor authentication (M
Upgrade
Learn more
Permissions for project "My First Project"
These permissions affect this project and all of its resources.
View by principals
View by roles
Grant access
Remove access
Filter
Enter property name or value
Type
Principal
Nam
johnallanmiranda.26csa@licet.ac.in
JOH
Grant access to "My First Project"
Grant principals access to this resource and add roles to specify what actions the principals can take. Optionally, add conditions to grant access to principals only when a specific criteria is met. Learn more about IAM conditions
Resource
My First Project
Add principals
Principals are users, groups, domains, or service accounts. Learn more about principals in IAM
New principals *
abishekjoshua.26csa@licet.ac.in
Assign roles
Roles are composed of sets of permissions and determine what the principal can do with this resource. Learn more
Role *
Owner
Full access to most Google Cloud resources. See the list of included permissions.
IAM condition (optional)
Add IAM condition
Add another role
Save
Cancel



RESULT:

Thus Cloud Organization in Google Cloud platform with Role-based access control is created.

EX NO: 1B CREATION OF CLOUD ORGANIZATION IN AWS
DATE:

AIM:

To Create a Cloud Organization in Amazon Web Services (AWS) with Role-based Access Control.

PROCEDURE:

Amazon Web Services (AWS), offered by Amazon, is a comprehensive cloud computing platform that provides infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS). AWS provides scalable cloud computing services, including computing power, storage, databases, machine learning, and analytics, to customers across various industries.

Creating an AWS Organization allows businesses to centrally manage multiple AWS accounts, enforce security policies, and optimize resource utilization. AWS Organizations enable Role-Based Access Control (RBAC), ensuring secure and efficient access management.

Creating Your AWS Organization and Setting Up an Account

1. Sign Up for AWS:
   Go to the AWS website and create an AWS account.

2. Access the AWS Management Console:
   Log in to the AWS Management Console.
   Navigate to the AWS Organizations service.

3. Create an AWS Organization:
   Click on "Create an Organization."

4. Create AWS Accounts within the Organization:
   In AWS Organizations, click Add an AWS Account to create or invite existing AWS accounts.

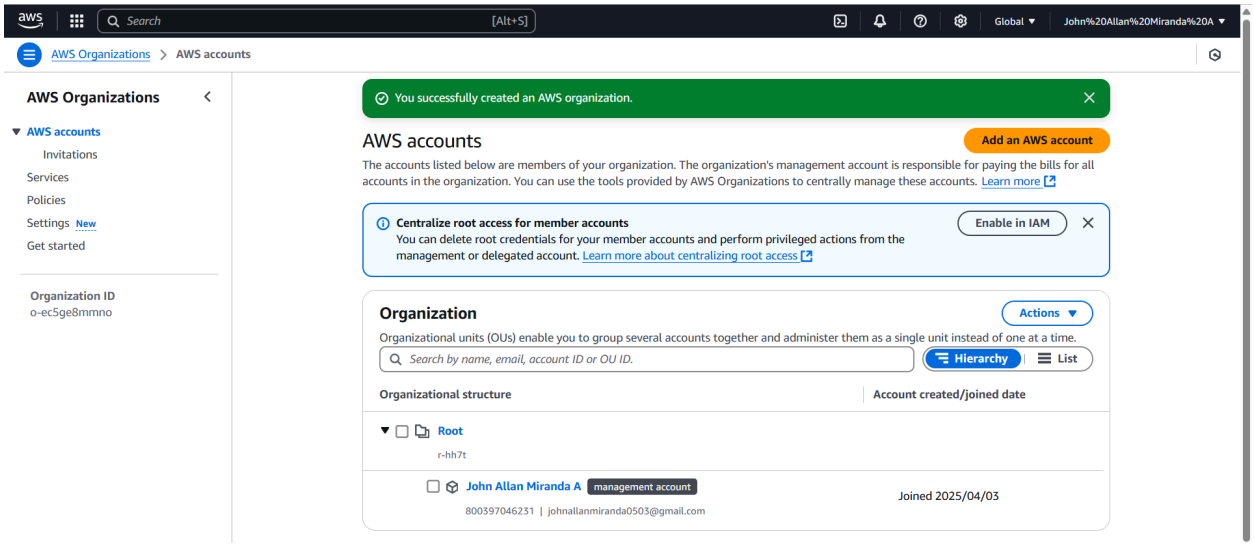
5. Enable and Configure IAM Policies for Role-Based Access Control (RBAC):
   Navigate to IAM (Identity and Access Management).
   Create Users, Groups, and Roles with specific permissions.
   Attach predefined or custom IAM Policies to enforce access control.

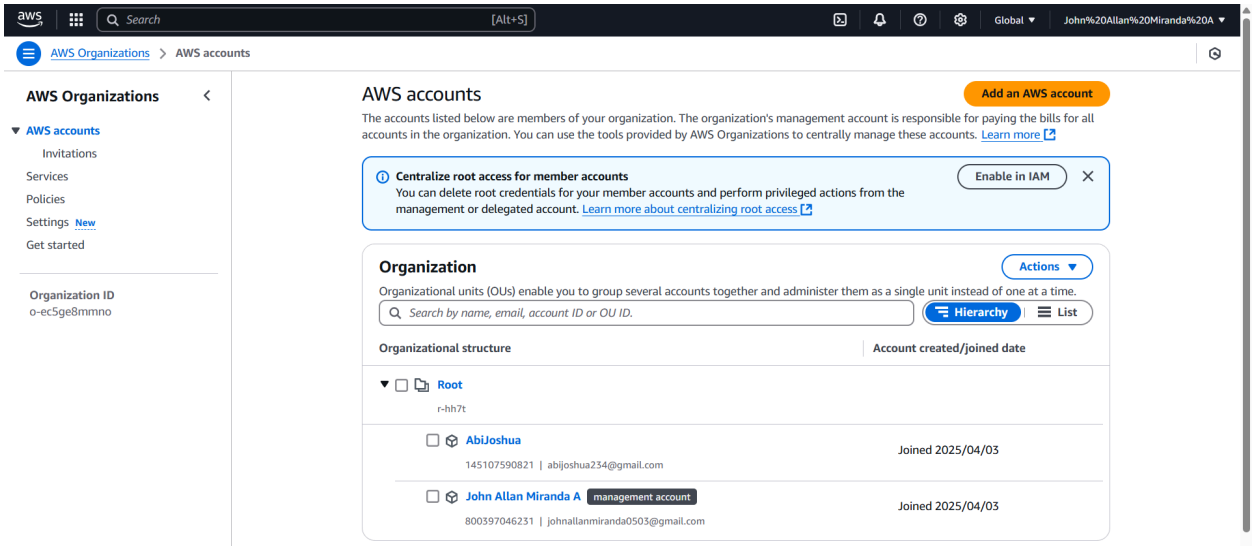
6. Enable AWS CLI for Command-Line Access:
   Download and install the AWS Command Line Interface (CLI).

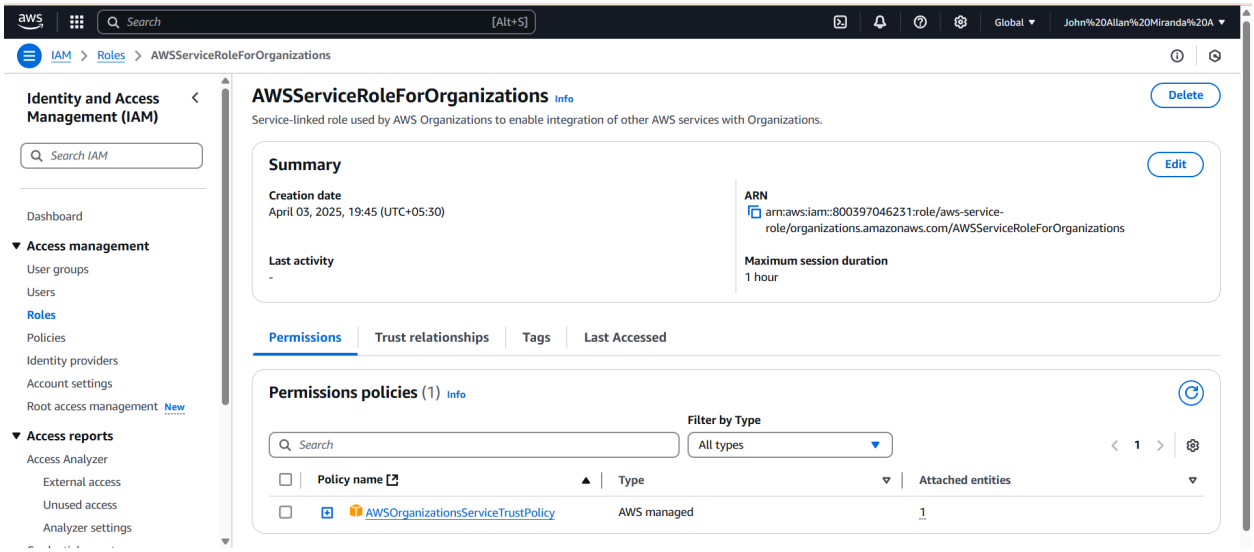
7. Configure the CLI using the following command:
   aws configure
   Enter your AWS Access Key, Secret Access Key, Region, and Output format.

8. Create and Deploy an Application in AWS:
   Choose a computing service such as Amazon EC2, AWS Lambda, or Elastic Beanstalk.
   Set up the required storage services like Amazon S3 or Amazon RDS.
   Deploy the application using AWS Management Console, AWS CLI, or Infrastructure as Code (IaC) tools like AWS CloudFormation or Terraform.

9. Select an AWS Region for Your Application:
   Choose a region closest to your users to minimize latency.
   Ensure compliance with data residency requirements.







RESULT :

Thus to Create a Cloud Organization in Amazon Web Services (AWS) with Role-based Access Control is created.

EX NO: 1C CREATION OF CLOUD ORGANIZATION IN AZURE
DATE:

AIM:

To Create a Cloud Organization in Azure with Role-based Access Control.

PROCEDURE:

Microsoft Azure, offered by Microsoft, is a cloud computing platform providing IaaS, PaaS, and SaaS solutions. Azure enables businesses to manage resources efficiently with secure role-based access control.

Creating an Azure Organization and Setting Up an Account

1. Sign Up for Azure:
   Go to the Azure website and create an account.
   New users get free credits for testing services.

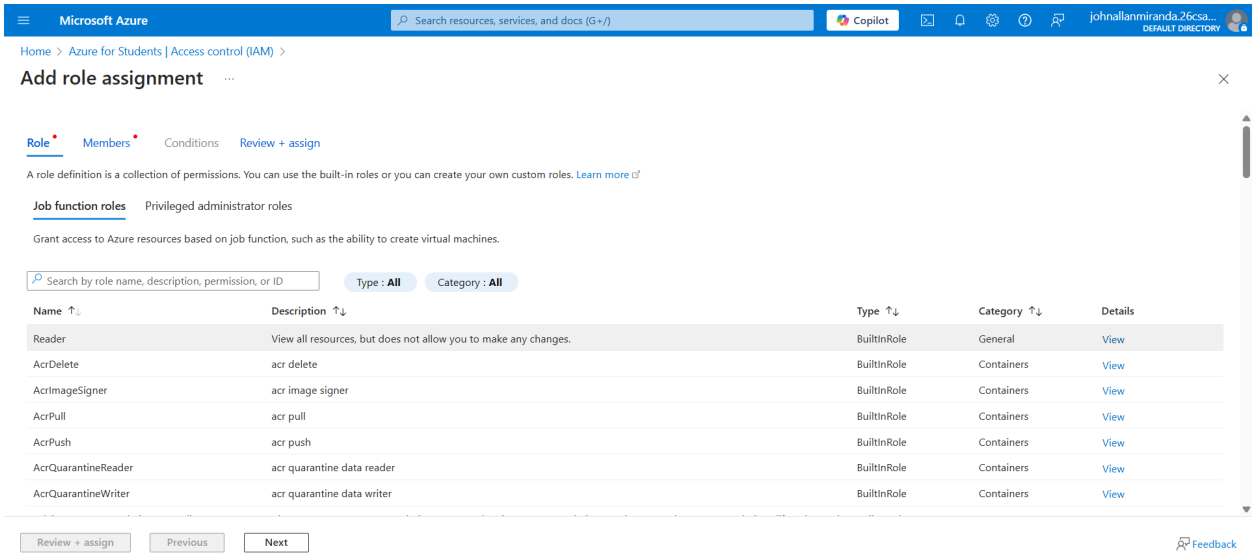
2. Access Azure Portal:
   Log in to the Azure Portal.
   Navigate to Azure Active Directory (Azure AD).

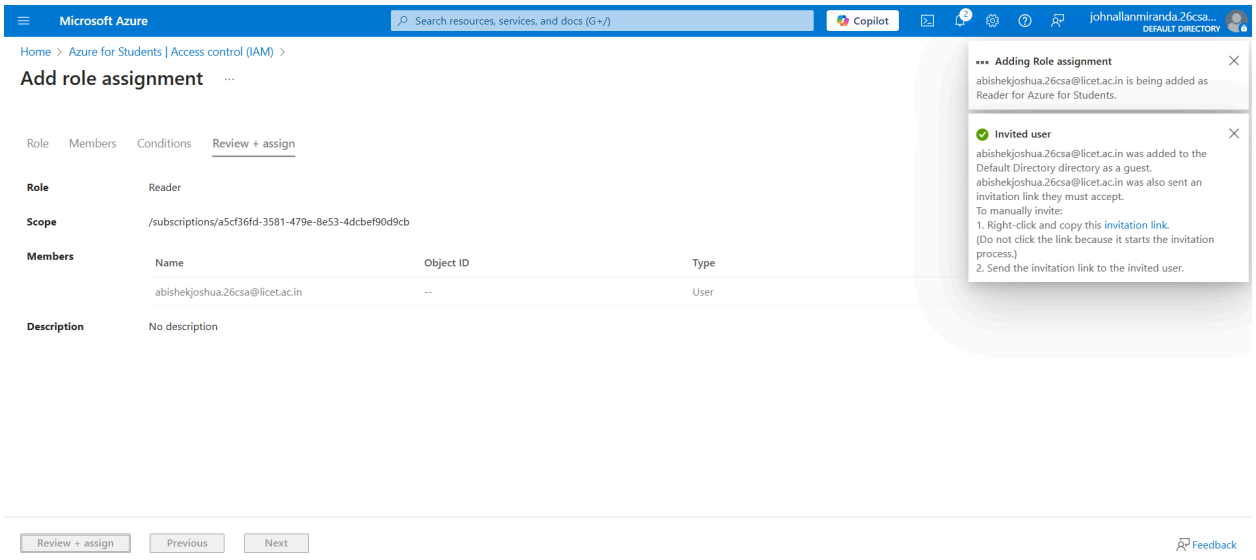
3. Create an Azure Tenant:
   Go to Azure AD > Manage Tenants > Create a Tenant.
   Choose an organization name and set up the domain.

4. Create and Manage Azure Subscriptions:
   Go to Subscriptions in the Azure Portal.
   Click Add Subscription to create a new one.

5. Enable and Configure RBAC (Role-Based Access Control):
   Navigate to Azure AD > Users and Groups.
   Create Users, Groups, and Roles with specific permissions.
   Assign RBAC roles using Azure IAM (Identity and Access Management).

6. Create and Deploy an Application in Azure:
   Choose a service like Azure Virtual Machines (VMs), Azure Functions, or App Services.
   Set up storage using Azure Blob Storage or Azure SQL Database.
   Deploy using the Azure Portal, CLI, or Infrastructure as Code (Bicep, ARM templates, Terraform).





RESULT:

Thus to Create a Cloud Organization in Azure with Role-based Access Control is created.