

## Assignment 9 – API Gateway and Cognito

4 endpoints:

API Gateway > APIs > Resources - Course\_API (8563oqbpa)

API actions **Deploy API**

Create resource

/

- /courses
  - GET
  - PATCH
  - POST
- /{courseCode}
  - GET

/courses - POST - Method execution

Update documentation Delete

ARN: arn:aws:execute-api:us-east-1:009448477287:8563oqbpa/\*POST/courses

Resource ID: mifn6s

Client → Method request → Integration request → Lambda integration → Integration response → Method response → Client

Method request Integration request Integration response Method response Test

Method request settings

Edit

Authorization: API key required

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

UserPool:

Amazon Cognito > User pools

Integrate Amazon Cognito with Amazon Verified Permissions

Amazon Verified Permissions is a fine-grained authorization service for role- and attribute-based access control in apps that authenticate with Amazon Cognito. From a user's identity or access token, Verified Permissions compares their attributes to access rules for your resource. The response is an authorization decision: allow or deny. Verified Permissions can consolidate authorization for all of your apps and resources into a central policy store. Your policies are written in Cedar, an open-source language built for access control. Learn more

Go to Amazon Verified Permissions

User pools (1) Info

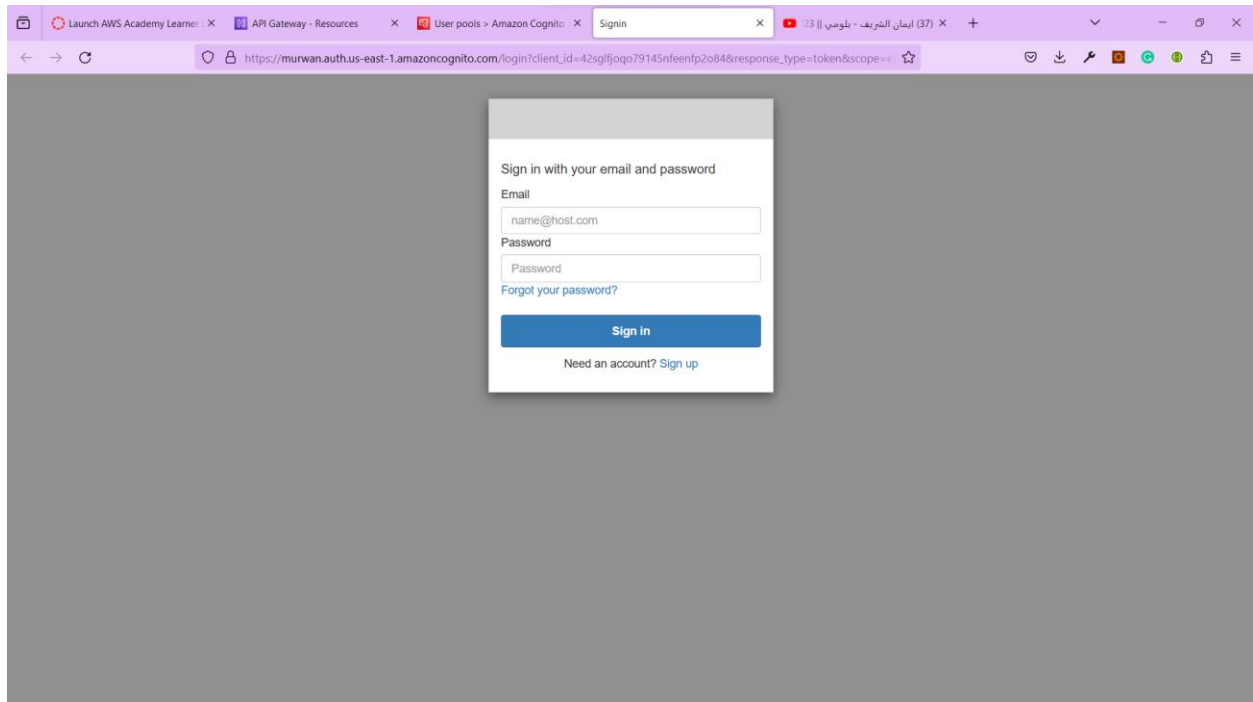
View and configure your user pools. User pools are directories of federated and local user profiles. They provide authentication options for your users.

Search user pools by name or ID

User pool name	User pool ID	Created time	Last updated time
MyPool	us-east-1_PWoe79n6L	3 minutes ago	3 minutes ago

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## Signin page:



The screenshot shows a web browser window with the URL `https://murwan.auth.us-east-1.amazonaws.com/login?client_id=42sgljfoqo79145nfeefp2o84&response_type=token&scope=...`. The page displays a "Sign in with your email and password" form. The form includes an "Email" field with the value "name@host.com", a "Password" field with the value "Password", and a "Forgot your password?" link. A blue "Sign In" button is at the bottom of the form, and a link "Need an account? Sign up" is below it.

Sign in with your email and password

Email

name@host.com

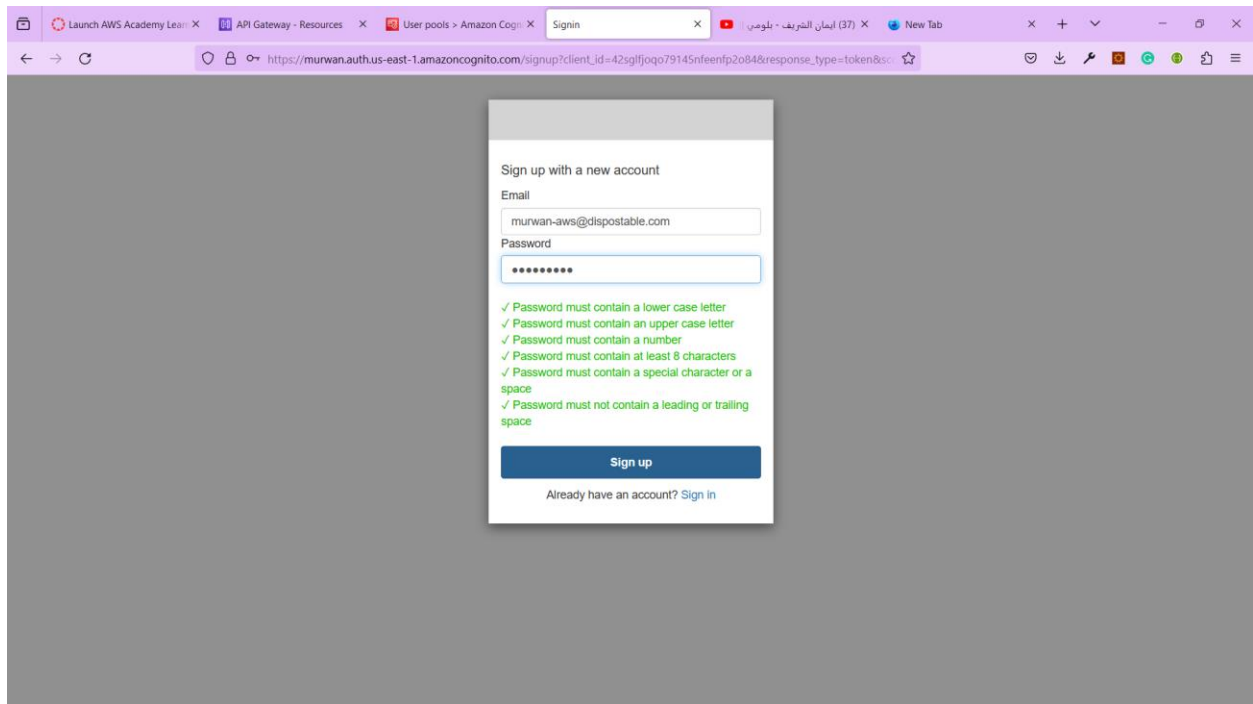
Password

Password

[Forgot your password?](#)

[Sign In](#)

[Need an account? Sign up](#)



The screenshot shows a web browser window with the URL `https://murwan.auth.us-east-1.amazonaws.com/signup?client_id=42sgljfoqo79145nfeefp2o84&response_type=token&scope=...`. The page displays a "Sign up with a new account" form. The form includes an "Email" field with the value "murwan-aws@dispostable.com", a "Password" field with the value "\*\*\*\*\*", and a list of password requirements. A blue "Sign up" button is at the bottom of the form, and a link "Already have an account? Sign in" is below it.

Sign up with a new account

Email

murwan-aws@dispostable.com

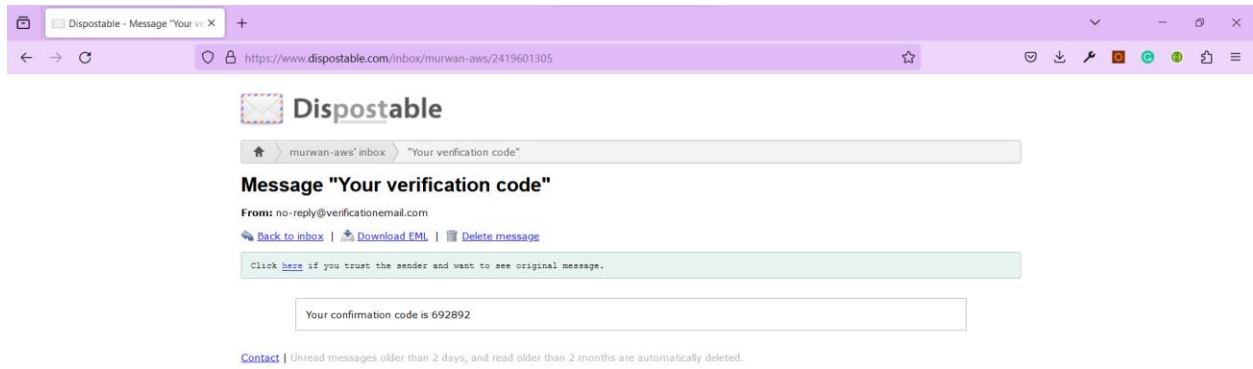
Password

\*\*\*\*\*

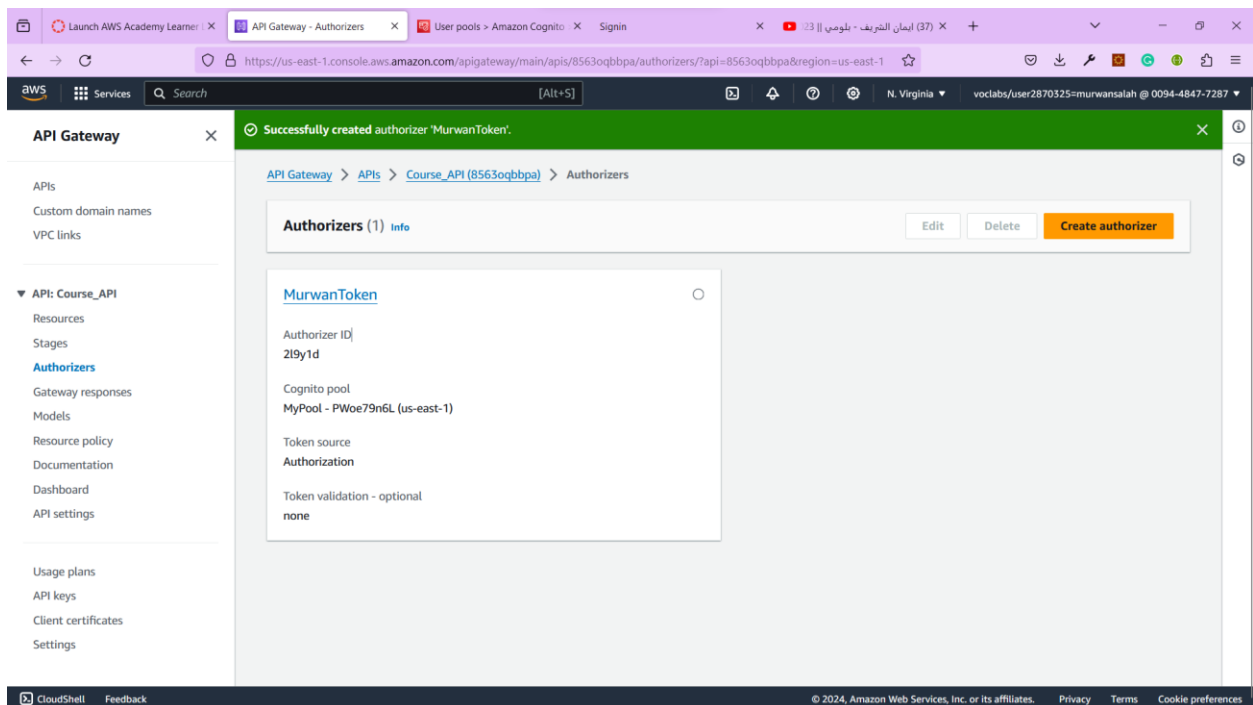
- ✓ Password must contain a lower case letter
- ✓ Password must contain an upper case letter
- ✓ Password must contain a number
- ✓ Password must contain at least 8 characters
- ✓ Password must contain a special character or a space
- ✓ Password must not contain a leading or trailing space

[Sign up](#)

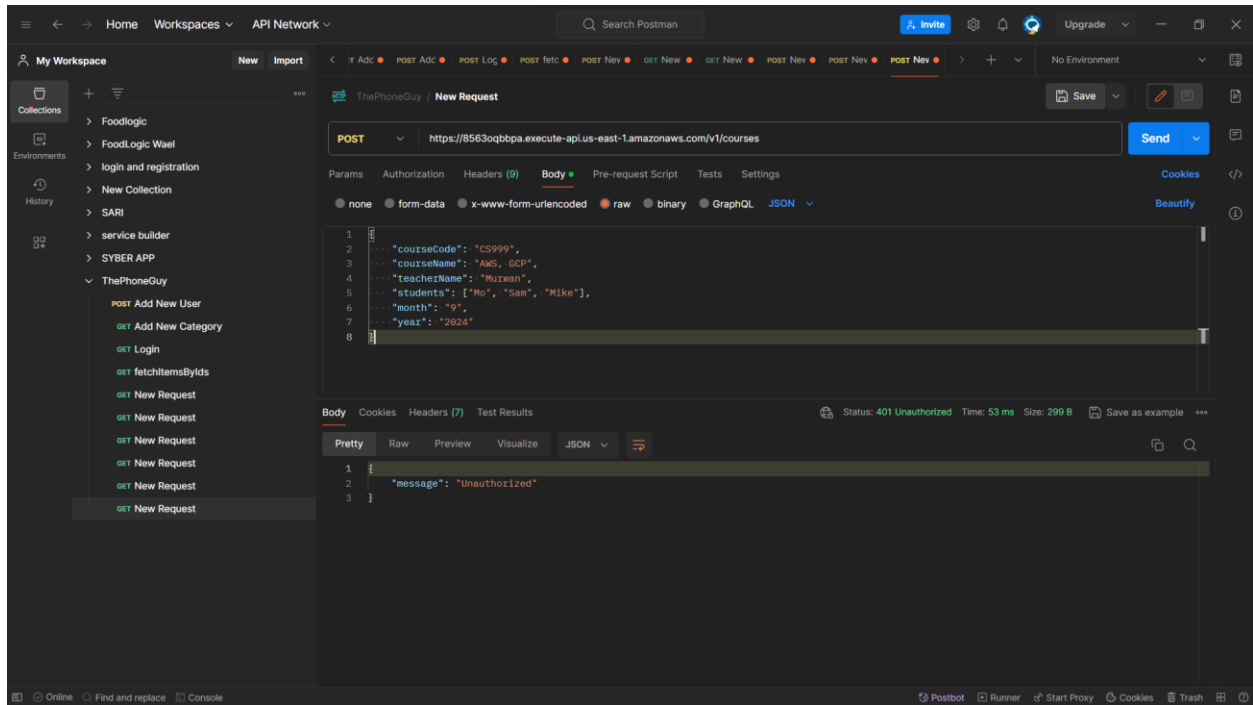
[Already have an account? Sign in](#)



## Authorization Token:



Adding course without token:



It worked with token:

