2. Creating a Serverless API

Objective: Develop a serverless API using AWS Lambda and API Gateway.

Approach:

● **Define API:** Design a simple RESTful API (e.g., for a todo list application).

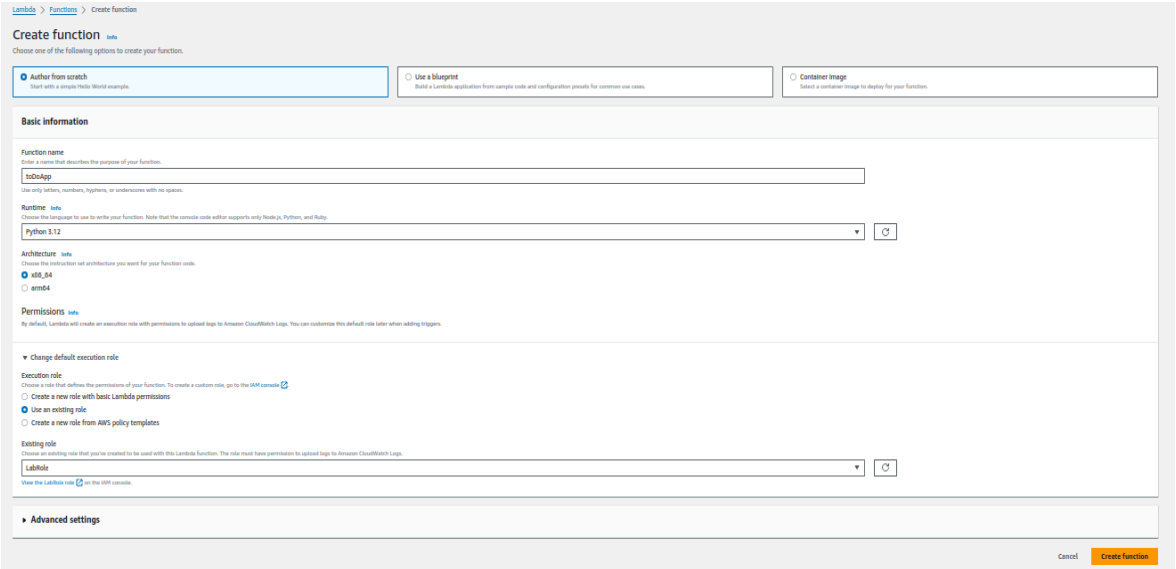
● **Lambda Functions:** Create Lambda functions for each API method (GET, POST, PUT, DELETE).

● **API Gateway Setup:** Use API Gateway to set up the API endpoints, connecting each endpoint to the corresponding Lambda function.

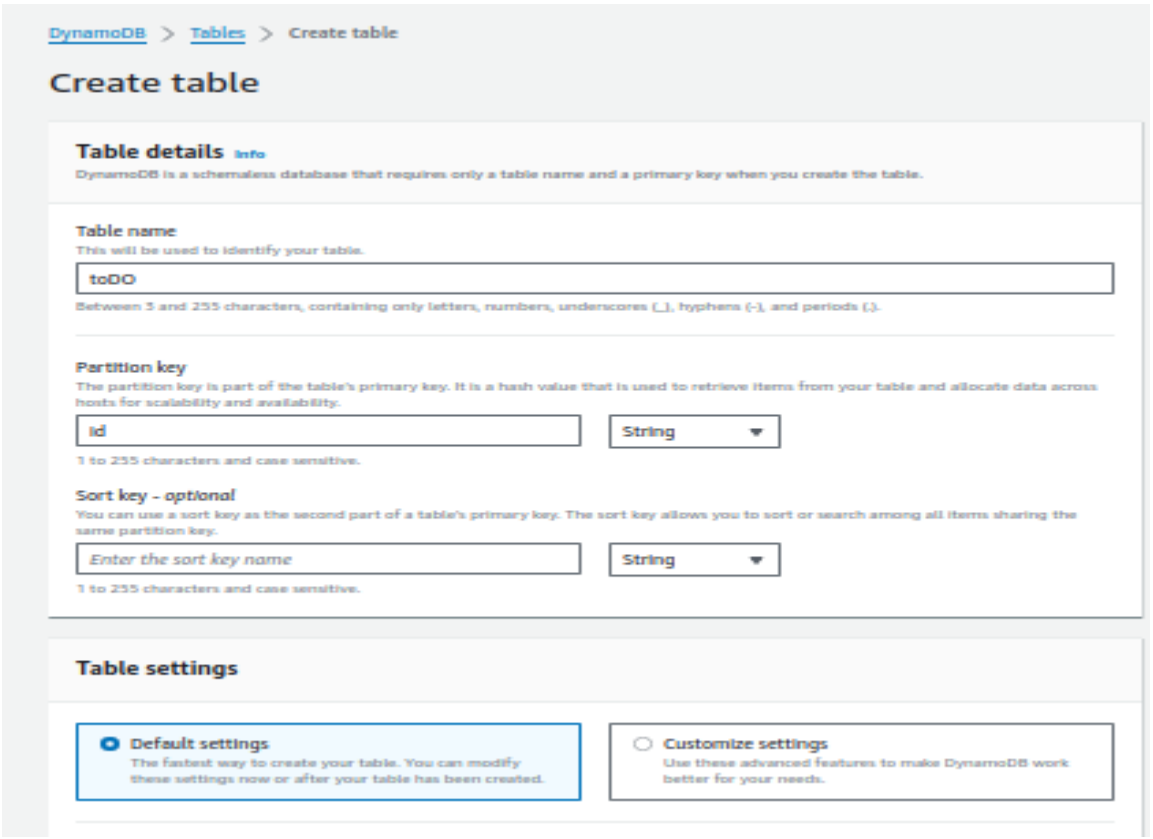
● **Testing**: Test the API using tools like Postman or AWS API Gateway test functionality.

Goal: Gain hands-on experience in building and deploying a serverless API, understanding the integration between Lambda and API Gateway.

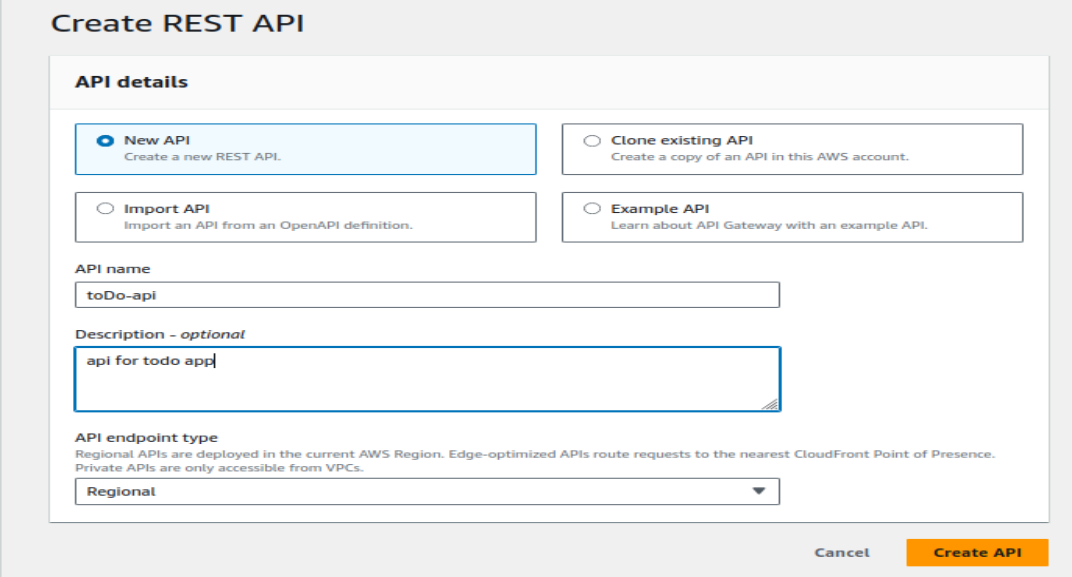
- Create lambda Function



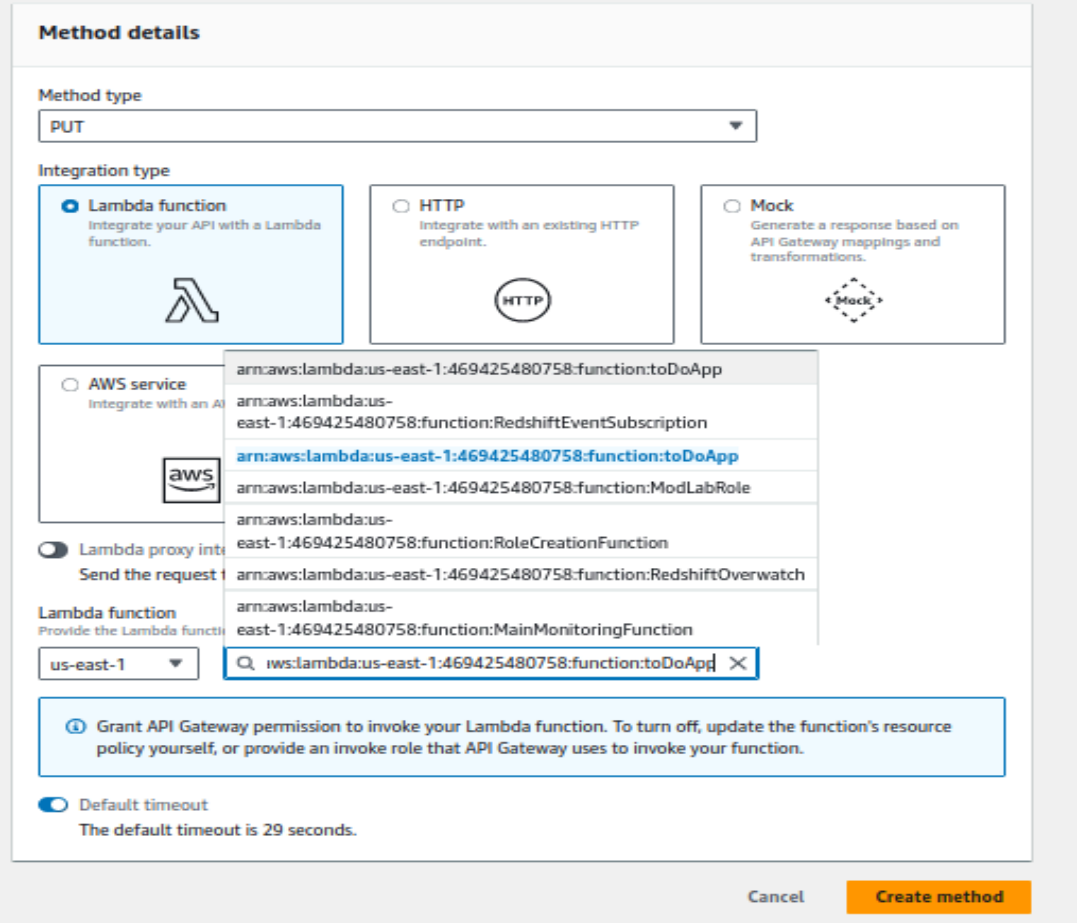
- Create DynamoDB table



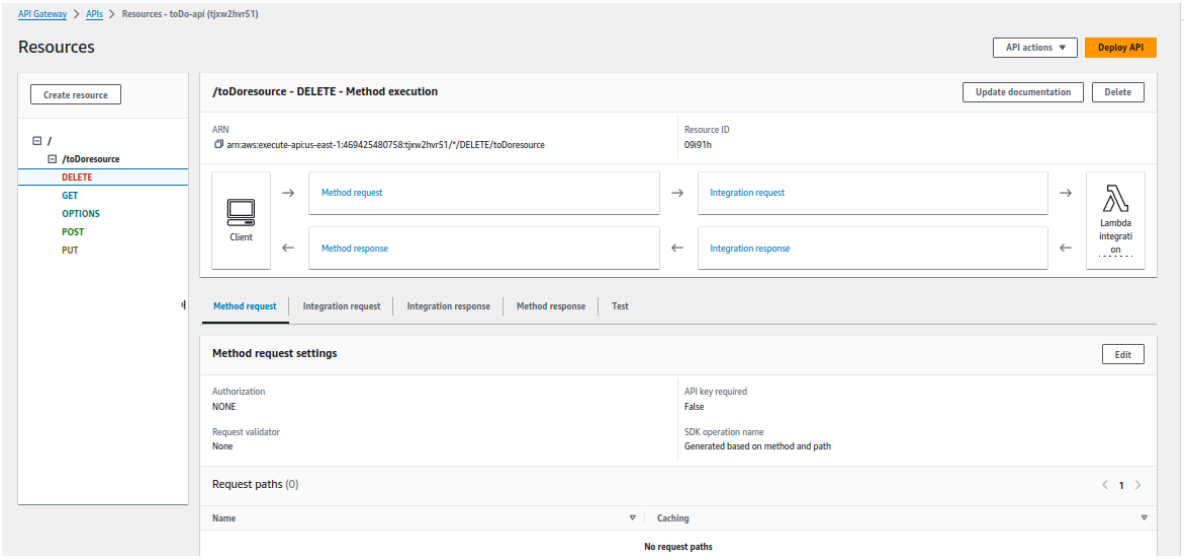
- Create REST api



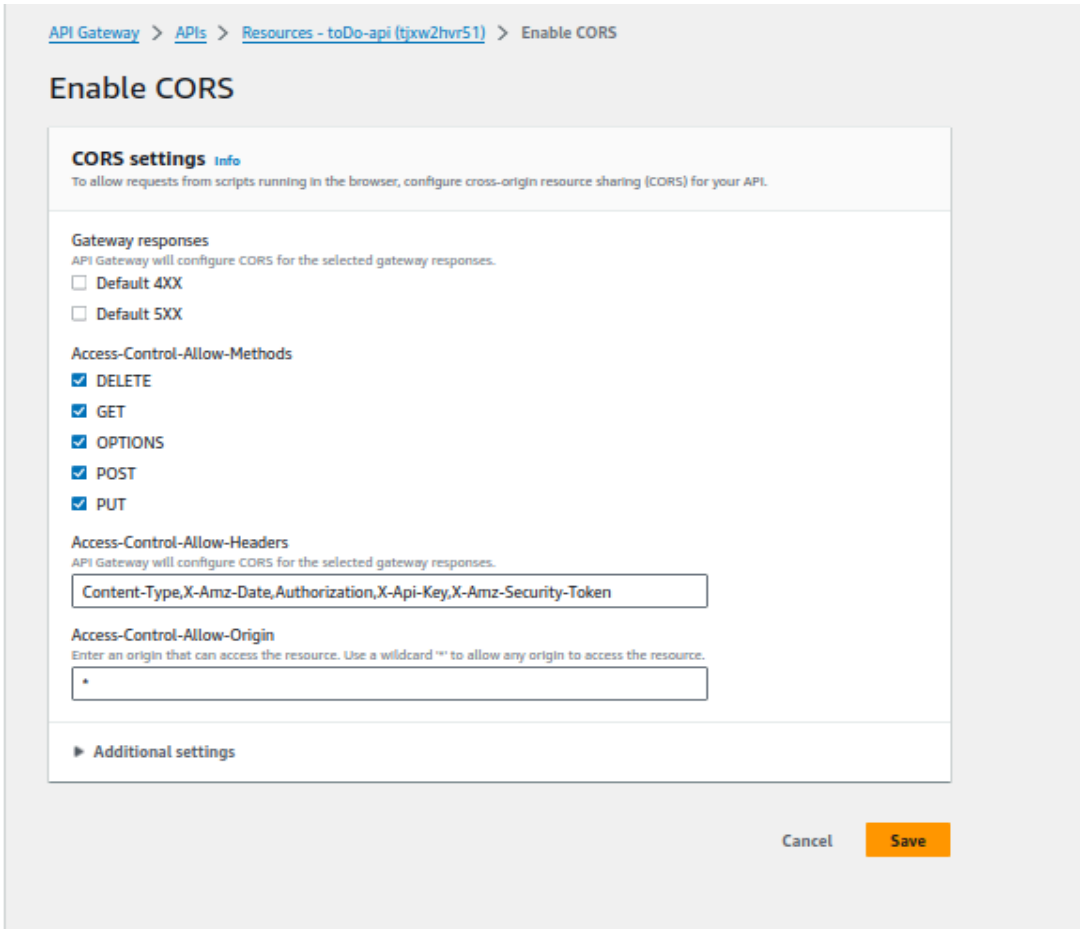
- Create Resource and create PUT, GET, POST, DELETE method



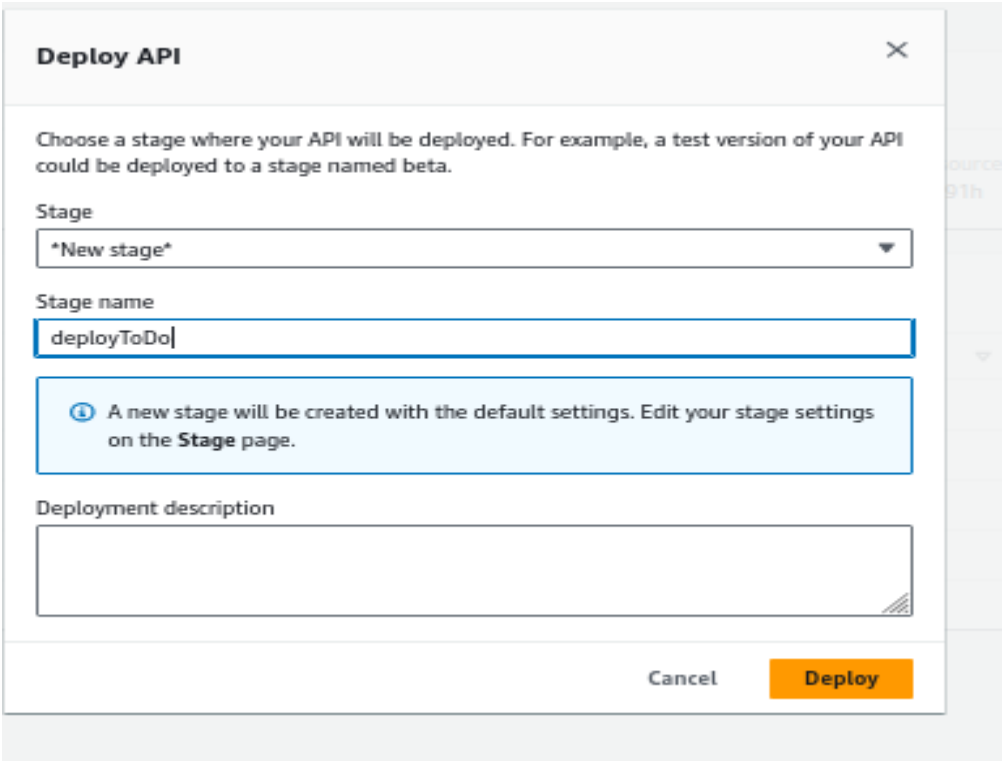
- Method list


API Gateway > APIs > Resources - toDo-api (tjxw2hvr51)
Resources
API actions
Deploy API
Create resource
/toDoresource - DELETE - Method execution
Update documentation
Delete
/toDoresource
DELETE
GET
OPTIONS
POST
PUT
ARN
arn:aws:execute-api:us-east-1:469425480758:tjxw2hvr51/*/DELETE/toDoresource
Resource ID
09i91h
Client
Method request
Integration request
Method response
Integration response
Lambda integration
Method request
Integration request
Integration response
Method response
Test
Method request settings
Edit
Authorization
NONE
API key required
False
Request validator
None
SDK operation name
Generated based on method and path
Request paths (0)
Name
Caching
No request paths

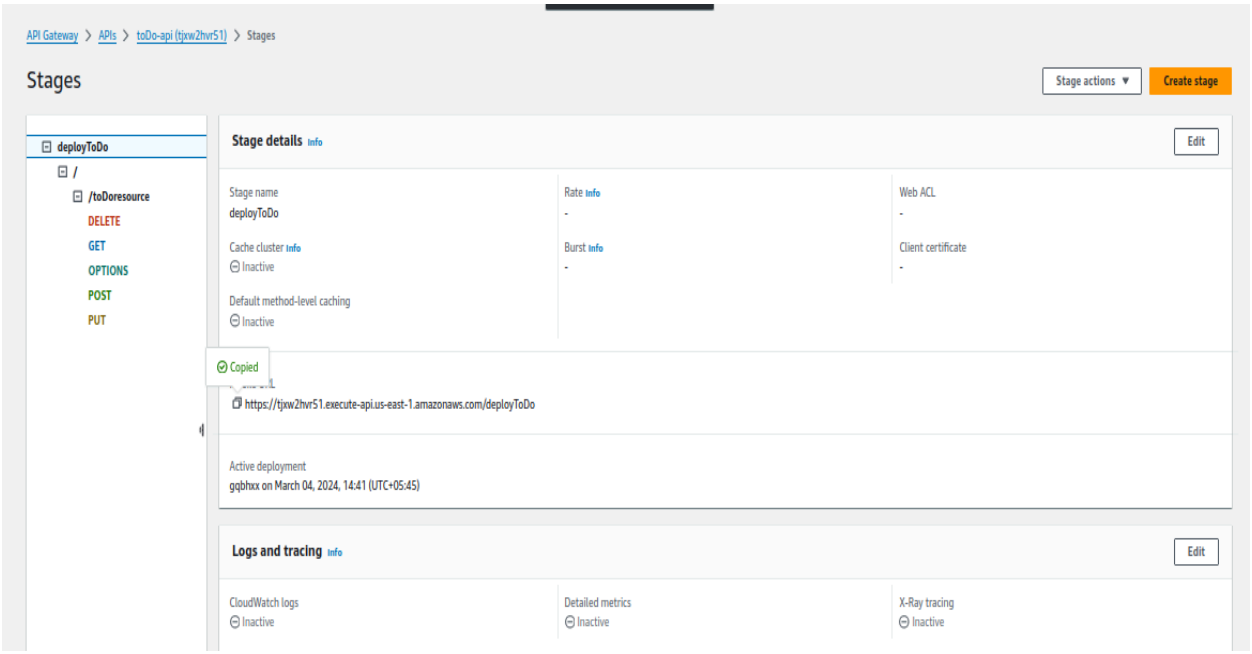
- Enable CORS


API Gateway > APIs > Resources - toDo-api (tjxw2hvr51) > Enable CORS
Enable CORS
CORS settings Info
To allow requests from scripts running in the browser, configure cross-origin resource sharing (CORS) for your API.
Gateway responses
API Gateway will configure CORS for the selected gateway responses.
Default 4XX
Default 5XX
Access-Control-Allow-Methods
DELETE
GET
OPTIONS
POST
PUT
Access-Control-Allow-Headers
API Gateway will configure CORS for the selected gateway responses.
Content-Type,X-Amz-Date,Authorization,X-Api-Key,X-Amz-Security-Token
Access-Control-Allow-Origin
Enter an origin that can access the resource. Use a wildcard '*' to allow any origin to access the resource.
*
Additional settings
Cancel
Save

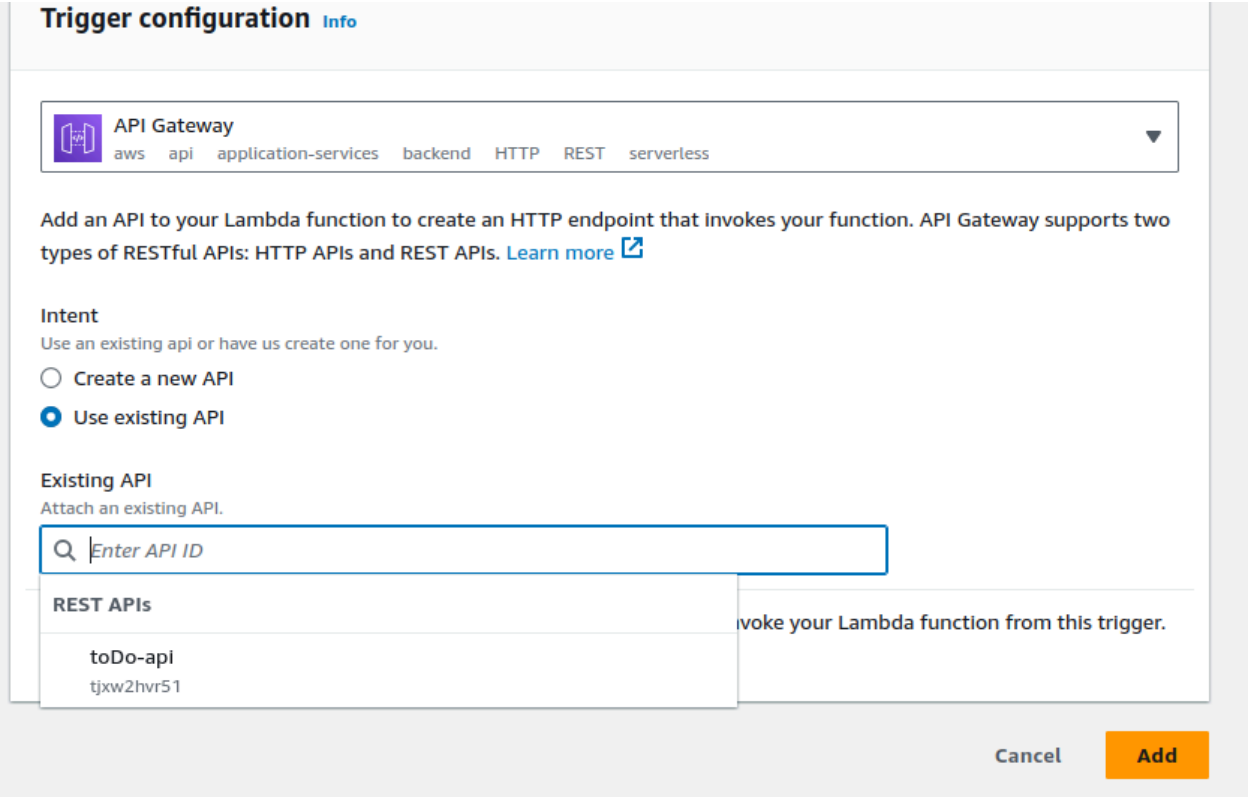
- Deploy API by creating new stage



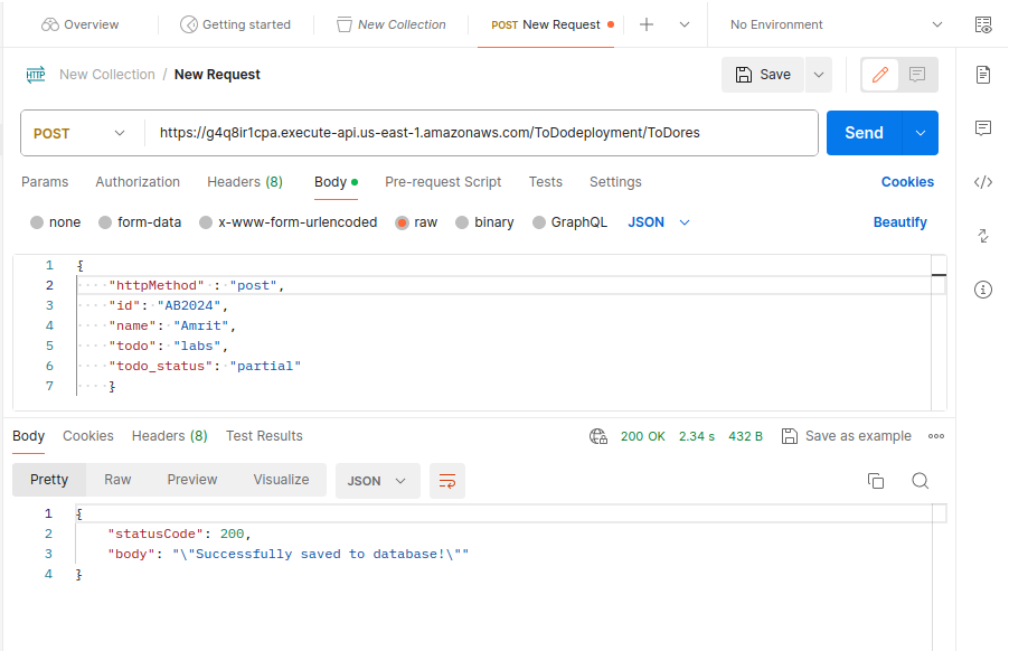
- From resource copy api Invoke url for particular method



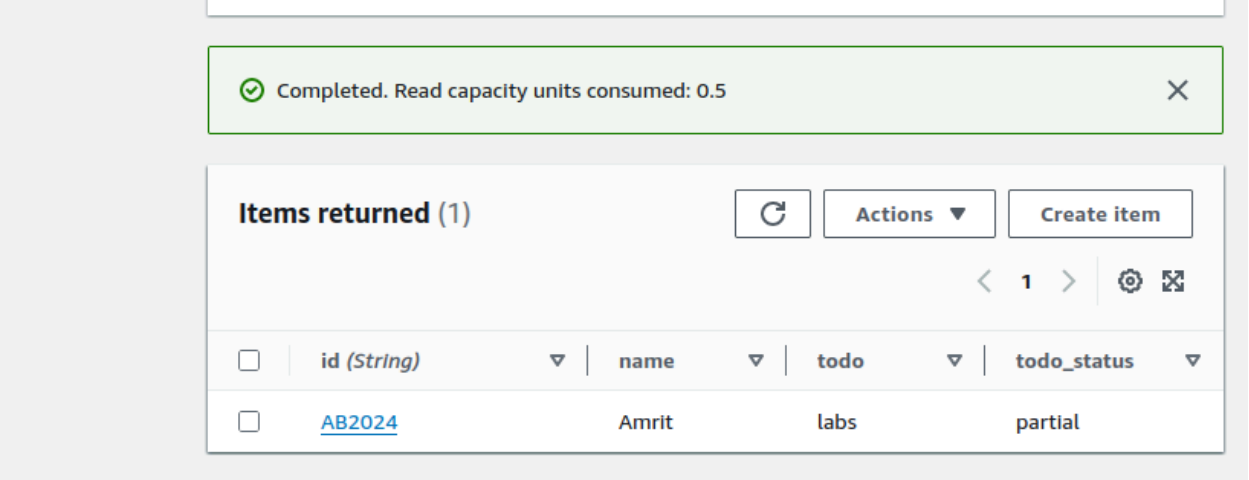
- Add trigger



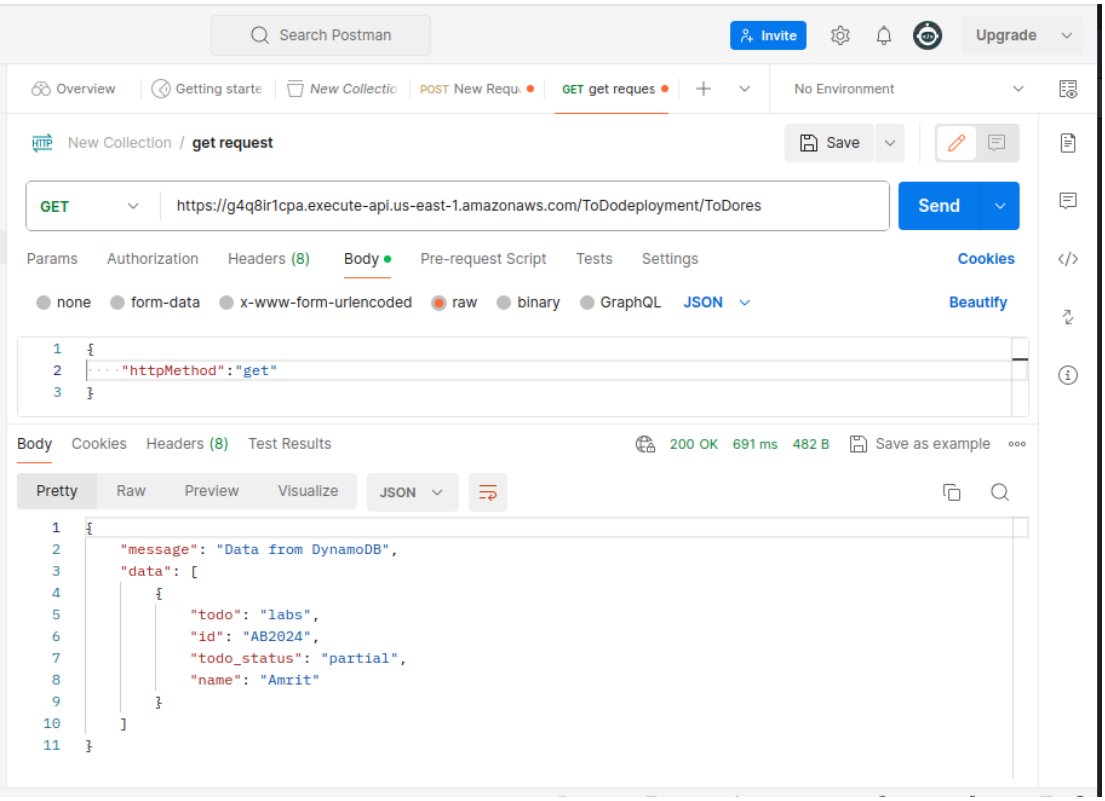
- Test POST method using Postman



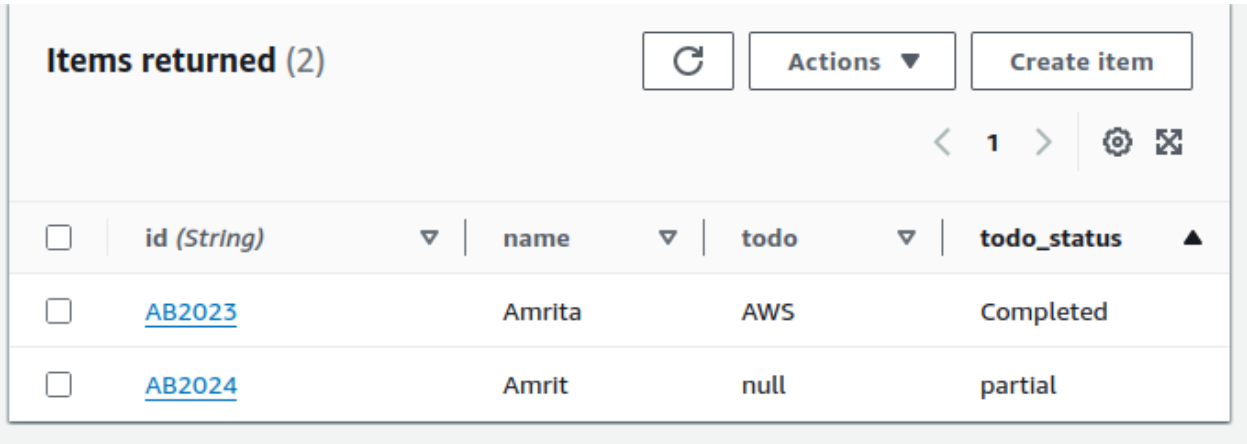
- Table after post



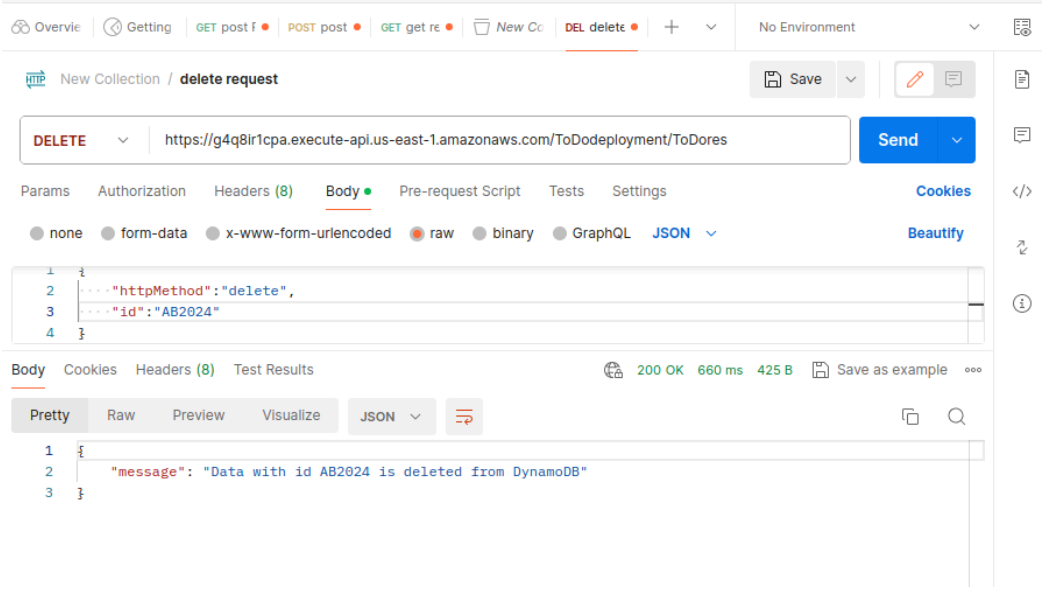
- Test GET method



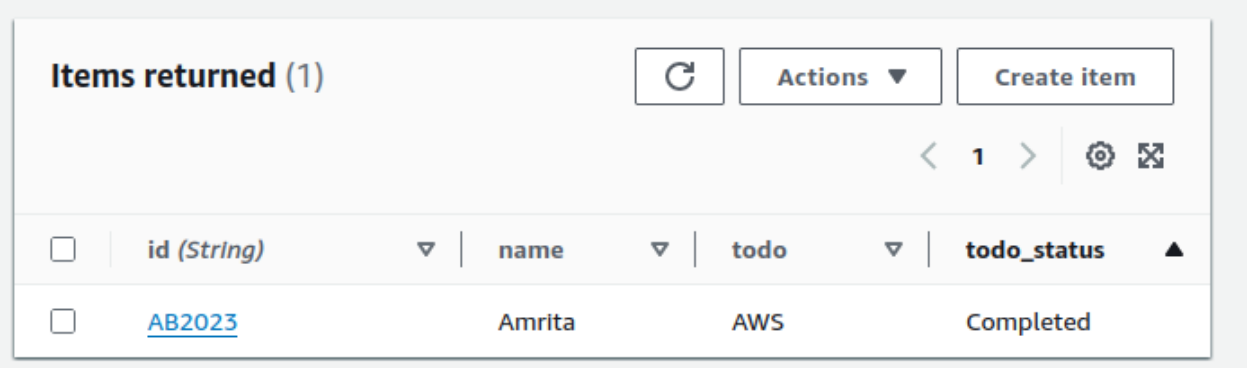
- Table Before DELETE request



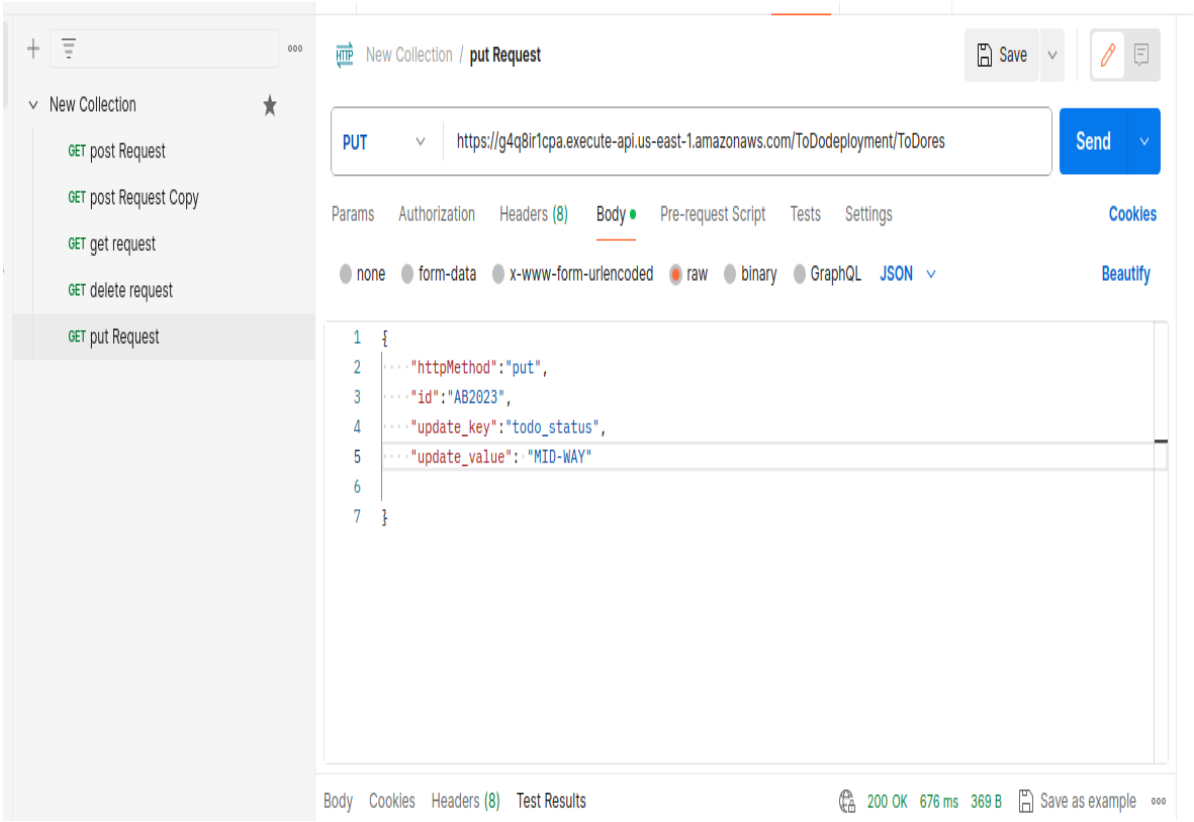
- Test DELETE request



- Table after DELETE request



- Test PUT method



- After PUT method /Updated table