

TASK - 1

Name : Ponvaishnavi MR

Location : Chennai

Serverless Web Application :

The screenshot shows the AWS DynamoDB Item Explorer interface. On the left, a sidebar menu includes 'Dashboard', 'Tables', 'Explore items' (which is selected), 'PartiQL editor', 'Backups', 'Exports to S3', 'Imports from S3', 'Integrations', 'Reserved capacity', and 'Settings'. Below this is a 'DAX' section with 'Clusters', 'Subnet groups', 'Parameter groups', and 'Events'. At the bottom of the sidebar are 'CloudShell' and 'Feedback' buttons. The main area displays a 'Tables (2)' section with a search bar and a 'Find tables' button. Two tables are listed: 'users' (selected) and 'UserSubmissions' (highlighted with a blue border). To the right, the 'UserSubmissions' table details are shown under the heading 'UserSubmissions'. It includes sections for 'Scan or query items' (with 'Scan' and 'Query' buttons), 'Select a table or index' (set to 'Table - UserSubmissions'), 'Select attribute projection' (set to 'All attributes'), and 'Filters - optional'. A 'Run' button is at the bottom. The status bar at the bottom right indicates '© 2025, Amazon Web Services, Inc. or its affiliates.' and links for 'Privacy', 'Terms', and 'Cookie preferences'.

This screenshot shows the results of a scan operation on the 'UserSubmissions' table. The interface is identical to the previous screenshot, but the 'Scan or query items' section now has a green success message: 'Completed · Items returned: 2 · Items scanned: 2 · Efficiency: 100% · RCUs consumed: 2'. Below this, a table titled 'Table: UserSubmissions - Items returned (2)' lists the results. The table has columns: submissionId (String), email, message, and name. Two items are listed:

submissionId (String)	email	message	name
9599b7ee-6fe7-4bff-abcc-...	vaishugane...	Hello, This i...	Ponvaishnavi
a2d71998-e37d-4737-a94...	john@exam...	Hello!	John Doe

The status bar at the bottom right indicates '© 2025, Amazon Web Services, Inc. or its affiliates.' and links for 'Privacy', 'Terms', and 'Cookie preferences'.

Screenshot of the AWS IAM Roles page:

The URL is <https://us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#/roles>

The page shows 10 roles listed in a table:

Role name	Trusted entities	Last activity
AWSserviceRoleForAPIGateway	AWS Service: ops.apigateway (Service)	-
AWSserviceRoleForResourceExplorer	AWS Service: resource-explorer-2 (Service)	15 minutes ago
AWSserviceRoleForSupport	AWS Service: support (Service-Linker)	2 hours ago
AWSserviceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service)	-
LambdaDynamoDBRole	AWS Service: lambda	-
QueryLambda-role-gzp8xfy0	AWS Service: lambda	-
SubmissionLambda-role-8p7h80r8	AWS Service: lambda	Yesterday
test-lambda-appln-py-role-ac0b8xq1	AWS Service: lambda	3 days ago
test_lambda_appln_role_6nw0022d	AWS Service: lambda	3 days ago

Navigation and other details:

- Left sidebar: Identity and Access Management (IAM) - User groups, Users, Roles, Policies, Identity providers, Account settings, Root access management.
- Top right: Account ID: 8115-7252-8734, Pонаишнави.
- Bottom: CloudShell, Feedback, © 2025, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, Cookie preferences.

Screenshot of the AWS Lambda Functions page:

The URL is <https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/functions>

The page shows 3 functions listed in a table:

Function name	Description	Package type	Runtime	Last modified
SubmissionLambda	-	Zip	Python 3.12	2 hours ago
QueryLambda	-	Zip	Python 3.12	2 hours ago
test-lambda-appln-py	-	Zip	Python 3.13	3 days ago

Success messages:

- Successfully updated the function QueryLambda.
- The test event "GetAllSubmissions" was successfully saved.

Tutorials section:

- Info: Learn how to implement common use cases in AWS Lambda.
- Tutorials:
 - Create a simple web app

Details for Create a simple web app:

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

Buttons: Learn more, Start tutorial.

Navigation and other details:

- Left sidebar: Lambda - Functions.
- Top right: Account ID: 8115-7252-8734, Pонаишнави.
- Bottom: CloudShell, Feedback, © 2025, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, Cookie preferences.

The screenshot shows the AWS Lambda Function Editor interface. The top navigation bar includes tabs for Items, Amazon D, Roles, IAM, Global, SubmissionLambda, API Gateway, Instance details, EC2 Instance Con, User Submission, and a search bar for https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#functions/SubmissionLambda?tab=code. The account ID is 8115-7252-8734, and the region is Asia Pacific (Mumbai). The user is Purnavishnavi.

The main area displays the function code in a code editor:

```
lambda_function.py
1 import json
2 import boto3
3 import uuid
4 from datetime import datetime
5
6 dynamodb = boto3.resource('dynamodb')
7 table = dynamodb.Table('UserSubmissions')
8
9 def lambda_handler(event, context):
10     try:
11         body = json.loads(event['body'])
12         name = body.get('name')
13         email = body.get('email')
14         message = body.get('message')
15
16         if not name or not email or not message:
17             return {
18                 "statusCode": 400,
19                 "body": "Missing required parameters: name, email, and message"
20             }
21
22         # Process the submission
23         # ...
24
25         response = table.put_item(
26             Item={
27                 "id": str(uuid.uuid4()),
28                 "name": name,
29                 "email": email,
30                 "message": message,
31                 "submitted_at": datetime.now().isoformat()
32             }
33         )
34
35         return {
36             "statusCode": 200,
37             "body": "Submission successful"
38         }
39     except Exception as e:
40         return {
41             "statusCode": 500,
42             "body": f"Error: {str(e)}"
43         }
```

The left sidebar shows the EXPLORER panel with a SUBMISSIONLAMBDA folder containing lambda_function.py, and a DEPLOY section with Deploy (Ctrl+Shift+U) and Test (Ctrl+Shift+) buttons. The TEST EVENTS section is expanded, showing options like Create new test event and Private saved events, with TestSubmission selected.

The bottom navigation bar includes PROBLEMS, OUTPUT (selected), CODE REFERENCE LOG, TERMINAL, and Tasks.

The right sidebar features an Info tab, a Tutorials tab (which is selected), and a Learn how to implement common use cases in AWS Lambda section. It also contains a Create a simple web app tutorial with a Start tutorial button.

The screenshot shows the AWS Lambda console interface. At the top, there are several tabs: Items | Amazon D... (highlighted), Roles | IAM | Glob... (disabled), SubmissionLambda (highlighted), API Gateway - Sta..., Instance details (disabled), EC2 Instance Con... (disabled), and User Submission (disabled). The URL in the browser is https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/functions/SubmissionLambda?tab=testing.

The main navigation bar includes the AWS logo, a search bar with the placeholder "Search" and an "Alt+S" keyboard shortcut, and various icons for account management and help. The region is set to "Asia Pacific (Mumbai)" and the account ID is 8115-7252-8734. The user's name, Ponavishnavi, is also visible.

The left sidebar shows the navigation path: Lambda > Functions > SubmissionLambda. The right sidebar has tabs for "Info" (selected) and "Tutorials". A green success message at the top states: "The test event "GetAllSubmissions" was successfully saved."

The main content area displays the "Test" tab selected. It shows a green box indicating the function executed successfully: "Executing function: succeeded (logs [2])". Below this is a "Details" section with a collapsible "Logs" panel containing JSON log output:

```
{  
  "statusCode": 200,  
  "headers": {  
    "Access-Control-Allow-Origin": "*"  
  },  
  "body": "{\"message\": \"Submission successful\", \"id\": \"a2d71998-e37d-4737-a943-c271ee983ca3\""}  
}
```

Below the logs is a "Summary" section with the following details:

Code SHA-256	Execution time
NMNBAacu917fTvvCjCDWPwyBncktg2afDw3GcZ+cmM=	2 hours ago
Function version	Request ID
\$LATEST	b523f07e-2fc4-4501-8ba2-3b67ca737cb9

To the right, a "Create a simple web app" tutorial is shown, with a summary and a "Start tutorial" button.

The screenshot shows the AWS Lambda console interface. The URL in the browser is <https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/functions/QueryLambda?tab=code>. The page title is "Lambda > Functions > QueryLambda". A green success message at the top says "The test event "GetAllSubmissions" was successfully saved." Below it, there are tabs for "Code", "Test", "Monitor", "Configuration", "Aliases", and "Versions". The "Code" tab is selected. On the left, the "Code source" sidebar shows the "EXPLORER" view with a file named "lambda_function.py" selected. The main area displays the Python code:

```
lambda_function.py
1 import json
2 import boto3
3 from boto3.dynamodb.conditions import Attr
4
5 dynamodb = boto3.resource('dynamodb')
6 table = dynamodb.Table('UserSubmissions')
7
8 def lambda_handler(event, context):
9     try:
10         params = event.get('queryStringParameters') or {}
11         email = params.get('email')
12
13         if email:
```

The screenshot shows the AWS Lambda console interface after testing the function. The URL in the browser is <https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/functions/QueryLambda?tab=testing>. The page title is "Lambda > Functions > QueryLambda". A green success message at the top says "The test event "GetAllSubmissions" was successfully saved." Below it, there are tabs for "Code", "Test", "Monitor", "Configuration", "Aliases", and "Versions". The "Test" tab is selected. The main area displays the test results under "Executing function: succeeded (logs [?])". It shows the following log output:

```
{"statusCode": 200,
"headers": {
    "Access-Control-Allow-Origin": "*"
},
"body": "[{"submissionDate": "\u00222025-11-01T11:45:59.856003\u0022, "message": "\u0022Hello!\u0022, "email": "\u0022john@example.com\u0022, "name": "\u0022John Doe\u0022, "submissionId": "\u0022a2d71998-e37d-4737-a943-c271ee983ca3\u0022, "status": "\u0022new\u0022}]"
}
```

Below the logs, there is a "Details" section showing the response body and execution time. The "Summary" section includes the "Code SHA-256" (4FXpv7UXc8Tu3grDvN4JRRsW4dblwOl7H+DUN3HIYK4=), "Function version" (\$LATEST), and "Request ID" (66ca892d-8cba-4a15-a7a6-5809dd3d1a58). The right side of the screen features a "Create a simple web app" tutorial.

Screenshot of the AWS API Gateway console showing a successful deployment of the UserSubmissionAPI.

APIs (1/1)

Name	Description	ID	Protocol	API endpoint type	Created
UserSubmissionAPI		n9649krgbj	REST	Regional	2025-11-01

CloudShell Feedback

Screenshot of the AWS API Gateway console showing the resources for the UserSubmissionAPI.

Resources

Create resource

- /
- /submissions
 - GET
 - OPTIONS
- /submit
 - OPTIONS
 - POST

Resource details

Path: /

Resource ID: 8ewbf9htrb

Methods (0)

No methods defined.

API actions Deploy API

CloudShell Feedback

Screenshot of the AWS API Gateway console showing a successful deployment of the UserSubmissionAPI.

The deployment summary indicates:

- Successfully created deployment for UserSubmissionAPI. This deployment is active for prod.
- 0 0 0 6 0 0 0

The Resources section shows the following endpoint structure and methods:

- Path: /
 - /submissions
 - GET
 - OPTIONS
 - /submit
 - OPTIONS
 - POST

The Resource details for the /submit endpoint show:

- Path: /submit
- Resource ID: 5cj01x

The Methods section for the /submit endpoint shows two entries:

Method type	Integration type	Authorization	API key
OPTIONS	Mock	None	Not required
POST	Lambda	None	Not required

API actions and Deploy API buttons are available at the top right.

Screenshot of the AWS API Gateway console showing a successful deployment of the UserSubmissionAPI.

The deployment summary indicates:

- Successfully created deployment for UserSubmissionAPI. This deployment is active for prod.
- 0 0 0 6 0 0 0

The Resources section shows the following endpoint structure and methods:

- Path: /
 - /submissions
 - GET
 - OPTIONS
 - /submit
 - OPTIONS
 - POST

The Resource details for the /submissions endpoint show:

- Path: /submissions
- Resource ID: cf0uaw

The Methods section for the /submissions endpoint shows two entries:

Method type	Integration type	Authorization	API key
GET	Lambda	None	Not required
OPTIONS	Mock	None	Not required

API actions and Deploy API buttons are available at the top right.

The screenshot shows the AWS API Gateway Stages page. A green success message at the top states: "Successfully created deployment for UserSubmissionAPI. This deployment is active for prod." Below this, the "Stages" section shows a single stage named "prod". The "Stage details" panel for "prod" includes fields for Rate Info (10000), Cache cluster Info (Inactive), Default method-level caching (Inactive), and Web ACL (None). The "Active deployment" section shows the copied URL: <https://n9649krbj.execute-api.ap-south-1.amazonaws.com/prod>. The left sidebar lists the API: UserSubmissionAPI with its resources, stages, authorizers, and other settings.

The screenshot shows the AWS EC2 Instances page. It displays a single instance named "FormServer" with Instance ID i-02681090999727f9a, which is currently running. The instance is of type t3.micro and is located in the ap-south-1 region. The left sidebar shows the EC2 navigation menu with options like Dashboard, Events, Instances, Images, and Capacity Manager.

The screenshot shows a CloudShell terminal window with the following content:

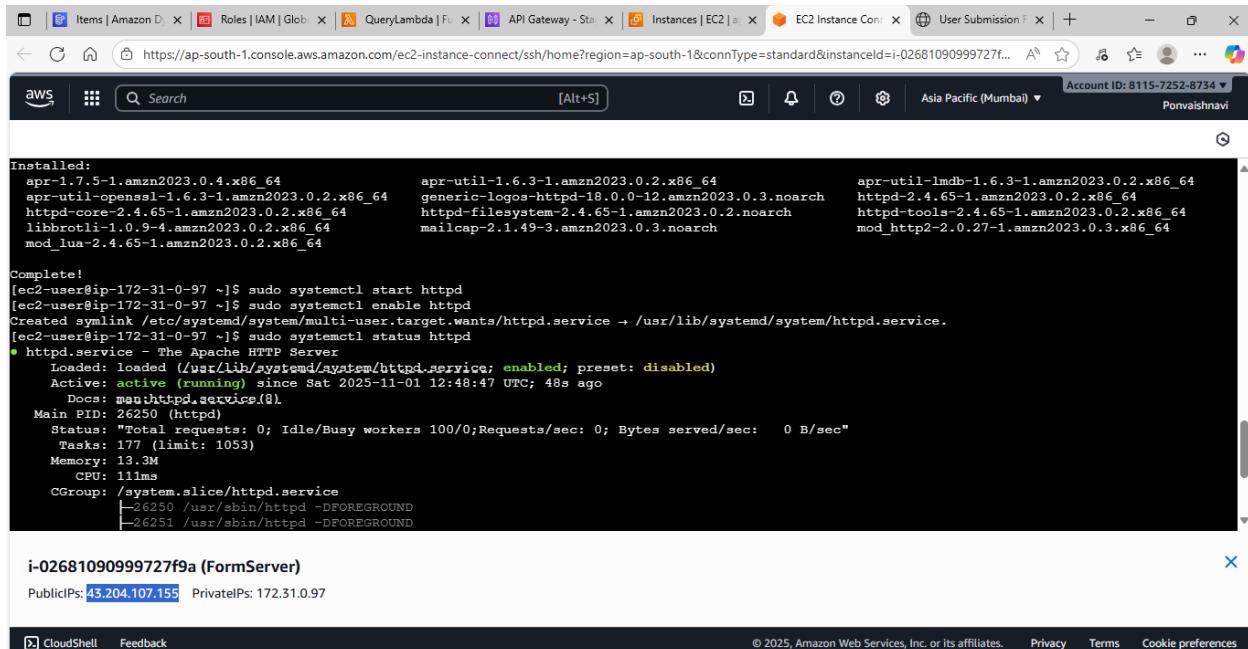
```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Sat Nov  1 12:45:07 2025 from 13.233.177.5
[ec2-user@ip-172-31-0-97 ~]$ sudo yum update -y
Amazon Linux 2023 Kernel Livepatch repository
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-0-97 ~]$ sudo yum update -y httpd
Last metadata expiration check: 0:00:12 ago on Sat Nov  1 12:47:56 2025.
Package httpd available, but not installed.
No match for argument: httpd
Error: No packages marked for upgrade.
[ec2-user@ip-172-31-0-97 ~]$ sudo yum install -y httpd
Last metadata expiration check: 0:00:25 ago on Sat Nov  1 12:47:56 2025.
Dependencies resolved.
```

i-02681090999727f9a (FormServer)

Public IPs: 43.204.107.155 Private IPs: 172.31.0.97

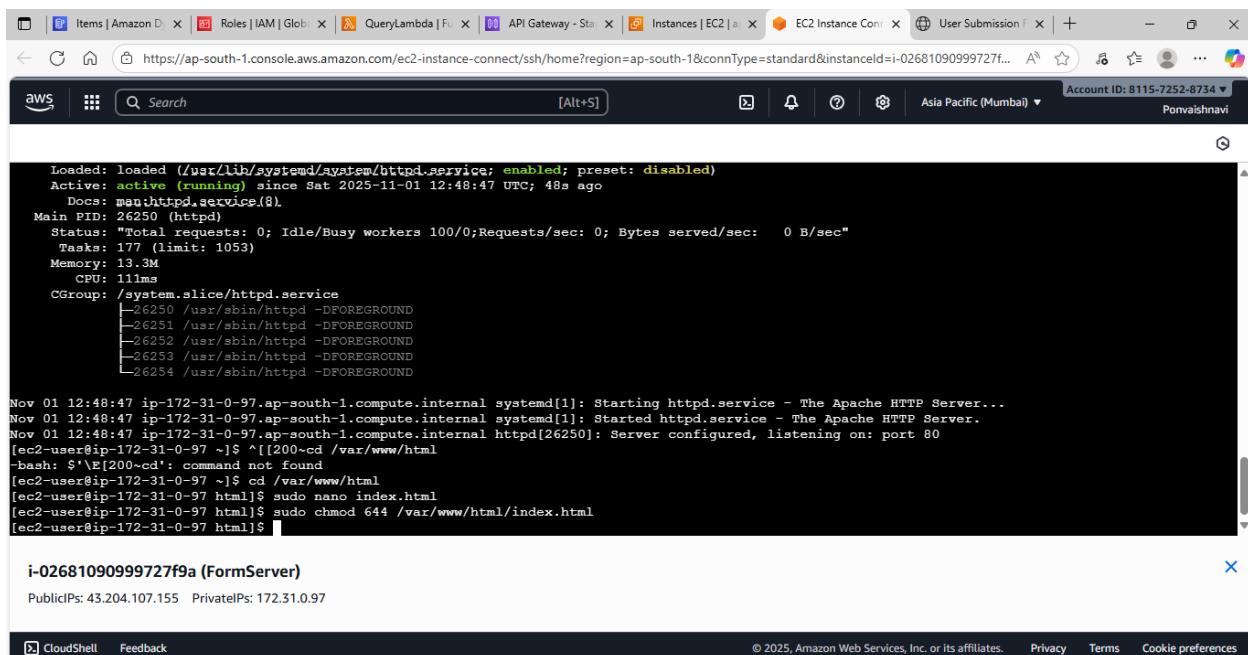
Package	Architecture	Version	Repository	Size
Installing:				
<code>httpd</code>	<code>x86_64</code>	<code>2.4.65-1.amzn2023.0.2</code>	<code>amazonlinux</code>	<code>47 k</code>
Installing dependencies:				
<code>apr</code>	<code>x86_64</code>	<code>1.7.5-1.amzn2023.0.4</code>	<code>amazonlinux</code>	<code>129 k</code>
<code>apr-util</code>	<code>x86_64</code>	<code>1.6.3-1.amzn2023.0.2</code>	<code>amazonlinux</code>	<code>97 k</code>
<code>apr-util-lmdb</code>	<code>x86_64</code>	<code>1.6.3-1.amzn2023.0.2</code>	<code>amazonlinux</code>	<code>13 k</code>
<code>generic-logos-httpd</code>	<code>noarch</code>	<code>18.0.0-12.amzn2023.0.3</code>	<code>amazonlinux</code>	<code>19 k</code>
<code>httpd-core</code>	<code>x86_64</code>	<code>2.4.65-1.amzn2023.0.2</code>	<code>amazonlinux</code>	<code>1.4 M</code>
<code>httpd-filesystem</code>	<code>noarch</code>	<code>2.4.65-1.amzn2023.0.2</code>	<code>amazonlinux</code>	<code>13 k</code>
<code>httpd-tools</code>	<code>x86_64</code>	<code>2.4.65-1.amzn2023.0.2</code>	<code>amazonlinux</code>	<code>81 k</code>
<code>libbrotli</code>	<code>x86_64</code>	<code>1.0.9-4.amzn2023.0.2</code>	<code>amazonlinux</code>	<code>315 k</code>
<code>mailcap</code>	<code>noarch</code>	<code>2.1.49-3.amzn2023.0.3</code>	<code>amazonlinux</code>	<code>33 k</code>
Installing weak dependencies:				
<code>apr-util-openssl</code>	<code>x86_64</code>	<code>1.6.3-1.amzn2023.0.2</code>	<code>amazonlinux</code>	<code>15 k</code>
<code>mod_http2</code>	<code>x86_64</code>	<code>2.0.27-1.amzn2023.0.3</code>	<code>amazonlinux</code>	<code>166 k</code>
<code>mod_lua</code>	<code>x86_64</code>	<code>2.4.65-1.amzn2023.0.2</code>	<code>amazonlinux</code>	<code>60 k</code>
Transaction Summary				
Install 13 Packages				



```
Installed:
  apr-1.7.5-1.amzn2023.0.4.x86_64      apr-util-1.6.3-1.amzn2023.0.2.x86_64
  apr-util-openssl-1.6.3-1.amzn2023.0.2.x86_64    generic-logos-httpsd-18.0.0-12.amzn2023.0.3.noarch
  httpd-core-2.4.65-1.amzn2023.0.2.x86_64     httpd-filesystem-2.4.65-1.amzn2023.0.2.noarch
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64       mailcap-2.1.49-3.amzn2023.0.3.noarch
  mod_lua-2.4.65-1.amzn2023.0.2.x86_64        mod_util-lmdb-1.6.3-1.amzn2023.0.2.x86_64
                                               httpd-2.4.65-1.amzn2023.0.2.x86_64
                                               httpd-tools-2.4.65-1.amzn2023.0.2.x86_64
                                               mod_http2-2.0.27-1.amzn2023.0.3.x86_64

Complete!
[ec2-user@ip-172-31-0-97 ~]$ sudo systemctl start httpd
[ec2-user@ip-172-31-0-97 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-172-31-0-97 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Active: active (running) since Sat 2025-11-01 12:48:47 UTC; 48s ago
     Docs: man:htpd.service(8)
 Main PID: 26250 (httpd)
 Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
   Tasks: 177 (limit: 1053)
  Memory: 13.3M
    CPU: 11ms
   CGroup: /system.slice/httpd.service
           ├─26250 /usr/sbin/httpd -DFOREGROUND
           ├─26251 /usr/sbin/httpd -DFOREGROUND
           ├─26252 /usr/sbin/httpd -DFOREGROUND
           ├─26253 /usr/sbin/httpd -DFOREGROUND
           ├─26254 /usr/sbin/httpd -DFOREGROUND

i-02681090999727f9a (FormServer)
PublicIPs: 43.204.107.155 PrivateIPs: 172.31.0.97
```



```
Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
Active: active (running) since Sat 2025-11-01 12:48:47 UTC; 48s ago
  Docs: man:htpd.service(8)
 Main PID: 26250 (httpd)
 Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
   Tasks: 177 (limit: 1053)
  Memory: 13.3M
    CPU: 11ms
   CGroup: /system.slice/httpd.service
           ├─26250 /usr/sbin/httpd -DFOREGROUND
           ├─26251 /usr/sbin/httpd -DFOREGROUND
           ├─26252 /usr/sbin/httpd -DFOREGROUND
           ├─26253 /usr/sbin/httpd -DFOREGROUND
           ├─26254 /usr/sbin/httpd -DFOREGROUND

Nov 01 12:48:47 ip-172-31-0-97.ap-south-1.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Nov 01 12:48:47 ip-172-31-0-97.ap-south-1.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Nov 01 12:48:47 ip-172-31-0-97.ap-south-1.compute.internal httpd[26250]: Server configured, listening on: port 80
[ec2-user@ip-172-31-0-97 ~]$ ^[[200~cd /var/www/html
-bash: $'\E[200~cd': command not found
[ec2-user@ip-172-31-0-97 ~]$ cd /var/www/html
[ec2-user@ip-172-31-0-97 html]$ sudo nano index.html
[ec2-user@ip-172-31-0-97 html]$ sudo chmod 644 /var/www/html/index.html
[ec2-user@ip-172-31-0-97 html]$
```

API Gateway - Stages | Instance details | EC2 | ap-south-1 | EC2 Instance Connect | ap-south-1 | User Submissions

Not secure 43.205.111.79

Submit Your Details

Name
Ponvaishnavi

Email
vaishuganesh03@gmail.com

Message
hello

Submit

✓ Submission successful

View All Submissions

Load Submissions

Name	Email	Message	Date	Status
Alice	alice@example.com	Hello, this is a test submission!	2025-11-04T10:58:04.543Z	new
Ponvaishnavi	vaishuganesh03@gmail.com	hello	2025-11-04T11:46:28.088Z	new