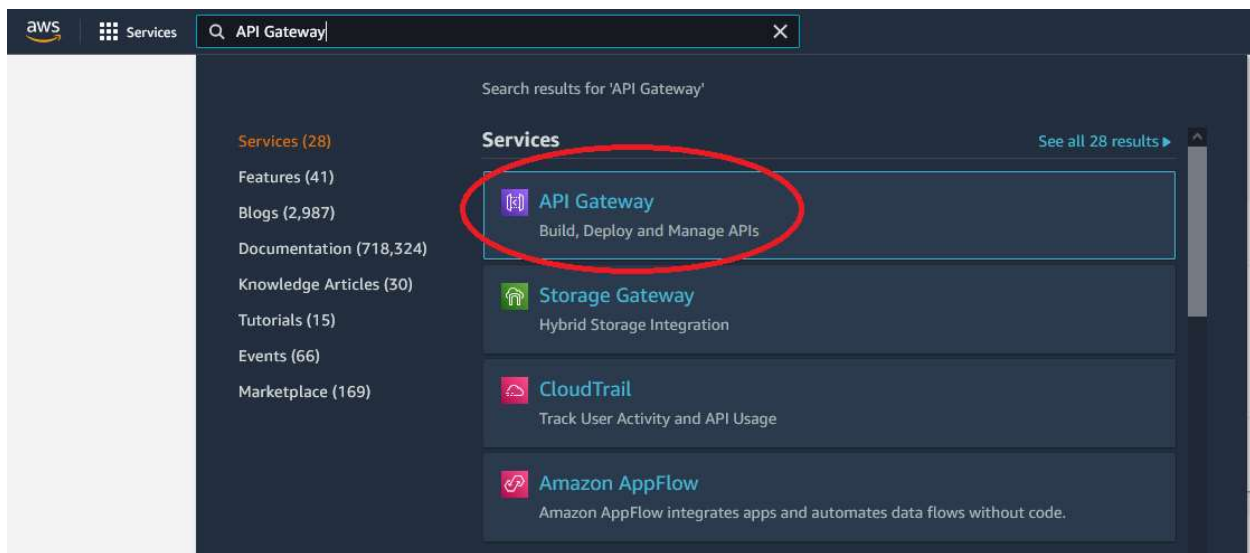
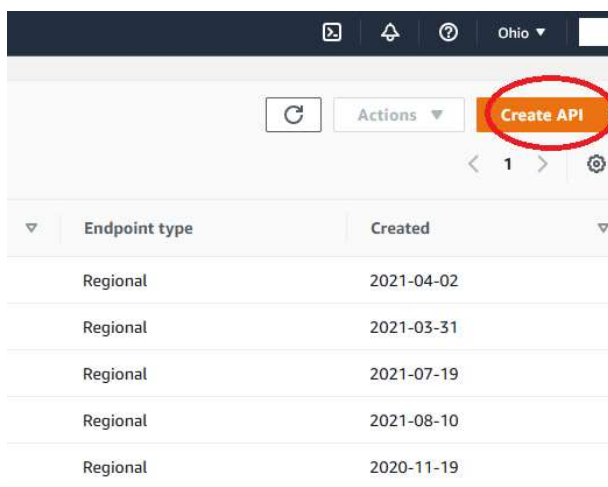


# Steps to create HTTP endpoint for a cloud function using API Gateway in AWS

1. Navigate to AWS Console at: <https://aws.amazon.com/console/>
2. Sign in using your credentials
3. Once in the AWS Console, search for “API Gateway” and select.



4. In the API Gateway window, select “Create API”



## 5. Select “HTTP API” – Build in the next window

### Choose an API type

#### HTTP API

Build low-latency and cost-effective REST APIs with built-in features such as OIDC and OAuth2, and native CORS support.

Works with the following:  
Lambda, HTTP backends

Import

Build

#### WebSocket API

Build a WebSocket API using persistent connections for real-time use cases such as chat applications or dashboards.

Works with the following:  
Lambda, HTTP, AWS Services

Build

#### REST API

Develop a REST API where you gain complete control over the request and response along with API management capabilities.

Works with the following:  
Lambda, HTTP, AWS Services

Import

Build

#### REST API Private

Create a REST API that is only accessible from within a VPC.

Works with the following:  
Lambda, HTTP, AWS Services

Import

Build

6. In the next window, select and enter the details as follow:

Create an API

---

**Create and configure integrations**

Specify the backend services that your API will communicate with. These are called integrations. For a Lambda integration, API Gateway invokes the Lambda function and responds with the response from the function. For HTTP integration, API Gateway sends the request to the URL that you specify and returns the response from the URL.

**Integrations** [Info](#)

(2) Lambda Remove

AWS Region (3) us-east-2 ▼ Lambda function (4)  × Version [Learn more.](#) 2.0 ▼

(1) Add integration

**API name**  
An HTTP API must have a name. This name is cosmetic and does not have to be unique; you will use the API's ID (generated later) to programmatically refer to this API.

(5)  (6)

Cancel Review and Create Next

- 1) Click "Add integration"
- 2) Select "Lambda"
- 3) Check the region that your Lambda function was created
- 4) Choose the Lambda function name from the dropdown list
- 5) Give the API a meaningful name
- 6) Select "Next"

7. In the “Configure routes” window, select the right method (POST) and click “Next”

## Configure routes

**Configure routes** [Info](#)

API Gateway uses routes to expose integrations to consumers of your API. Routes for HTTP APIs consist of two parts: an HTTP method and a resource path (e.g., GET /pets). You can define specific HTTP methods for your integration (GET, POST, PUT, PATCH, HEAD, OPTIONS, and DELETE) or use the ANY method to match all methods that you haven't defined on a given resource.

Method	Resource path	Integration target	
POST	/FactorialFunction	FactorialFunction	<a href="#">Remove</a>

[Add route](#)

[Cancel](#) [Previous](#) [Next](#)

8. In the “Define Stages”, leave everything intact and click “Next”

## Define stages

**Configure stages** [Info](#)

Stages are independently configurable environments that your API can be deployed to. You must deploy to a stage for API configuration changes to take effect, unless that stage is configured to autodeploy. By default, all HTTP APIs created through the console have a default stage named \$default. All changes that you make to your API are autodeployed to that stage. You can add stages that represent environments such as development or production.

Stage name	Auto-deploy	
\$default	<input checked="" type="checkbox"/>	<a href="#">Remove</a>

[Add stage](#)

[Cancel](#) [Previous](#) [Next](#)

9. In the next window “Review and create”, check the details again and click “Create”

## Review and create

**API name and integrations** Edit

API name

FactorialFunction

Integrations

FactorialFunction (Lambda)

**Routes** Edit

Routes

POST /FactorialFunction → FactorialFunction (Lambda)

**Stages** Edit

Stages

\$default (Auto-deploy: enabled)

Cancel Previous Create

10.The newly created API Gateway will have a field called “Invoke URL”, it has the following format:

[https://<APP\\_ID>.execute-api.us-east-2.amazonaws.com](https://<APP_ID>.execute-api.us-east-2.amazonaws.com)

11.Copy this URL and append it with the path you have entered in Step (6), the URL to invoke the function is:

[https://<APP\\_ID>.execute-api.us-east-2.amazonaws.com/FactorialFunction](https://<APP_ID>.execute-api.us-east-2.amazonaws.com/FactorialFunction)

12.Copy this whole URL and paste it in the file:

/workload.smoothen/src/main/resources/application.properties –  
“aws.factorial.url” key