

Hello World actualizado: AWS Lambda + API Gateway con Java

Haremos un microservicio REST que recibe un número **value** y devuelve su cuadrado. Usaremos:

- **Java 17**
 - **Maven**
 - **AWS Lambda**
 - **AWS API Gateway (REST API)**
-

1. Crear el proyecto y la clase del servicio

Crea un proyecto Maven simple.

Tu clase de servicio:

```
package co.edu.escuelaing.services;

public class MathServices {

    public static Integer square(Integer i){
        return i * i;
    }
}
```

Empaqueta:

```
mvn clean package
```

El jar resultante → lo subirás a Lambda.

2. Crear la función Lambda

En la consola de AWS:

1. AWS → **Lambda**
2. **Create function**
3. Tipo: **Author from scratch**
4. Nombre: **square**

5. Runtime: **Java 17**

6. Role:

- Si usas AWS Academy → **LabRole**
- Si no → usa un role existente o crea uno básico de ejecución Lambda.

Subir el código

En **Code source**:

- Selecciona "Upload from" → ".zip or .jar file"
- Sube tu *jar* del target de Maven.

Configurar Handler

En "Runtime settings":

```
co.edu.escuelaing.services.MathServices::square
```

Guarda.

3. Crear un Test Event

Vamos a probar pasando SOLO el número, **sin JSON**, porque la función recibe un **Integer** plano.

1. "Test"
2. "Configure test event"
3. Nombre: **testSquare**
4. Contenido del evento → **solo esto**:

```
5
```

Ejecuta. Debería devolverte **25**.

4. Crear el API REST en API Gateway

Crear el API

1. AWS → **API Gateway**
2. **Create API**
3. Tipo: **REST API**
4. "New API"

5. Nombre: `mathServices`

6. Endpoint type: **Regional**

Crear método GET

1. En el panel izquierdo → selecciona `/` (root)

2. "Actions" → **Create Method**

3. Selecciona **GET**

4. Integration type: **Lambda Function**

5. Lambda a usar: `square`

6. Guarda

5. Configurar Method Request (parámetro)

1. Selecciona **Method Request**

2. En *URL Query String Parameters* → **Add query string**

3. Nombre del parámetro: `value`

6. Configurar Integration Request

Aquí es donde hacemos el mapeo correcto del parámetro.

6.1 Mapear query → integración

En **Integration Request**:

- En "URL Query String Parameters":
 - Nombre: `value`
 - Mapea a: `method.request.querystring.value`

6.2 Crear Mapping Template (para Lambda)

En "Mapping Templates":

1. Content-Type: `application/json` (sí, aunque enviemos solo un número)

2. Template:

```
$input.params("value")
```

Esto hace que **Lambda reciba solo el número**, no un JSON.

Guarda.

7. Probar el método desde API Gateway

En "Method Execution" → **Test**

- Query string: `value=7`

Debes ver respuesta:

```
49
```

8. Deploy del API

1. Actions → **Deploy API**
2. [New Stage]
3. Nombre: `Beta`

El sistema generará una URL como:

```
https://xxxxx.execute-api.us-east-1.amazonaws.com/Beta
```

Para probar:

```
https://xxxxx.execute-api.us-east-1.amazonaws.com/Beta?value=14
```

Debería devolver:

```
196
```

9. Limpieza (para evitar cobros)

- Borrar la función Lambda
- Borrar el API en API Gateway