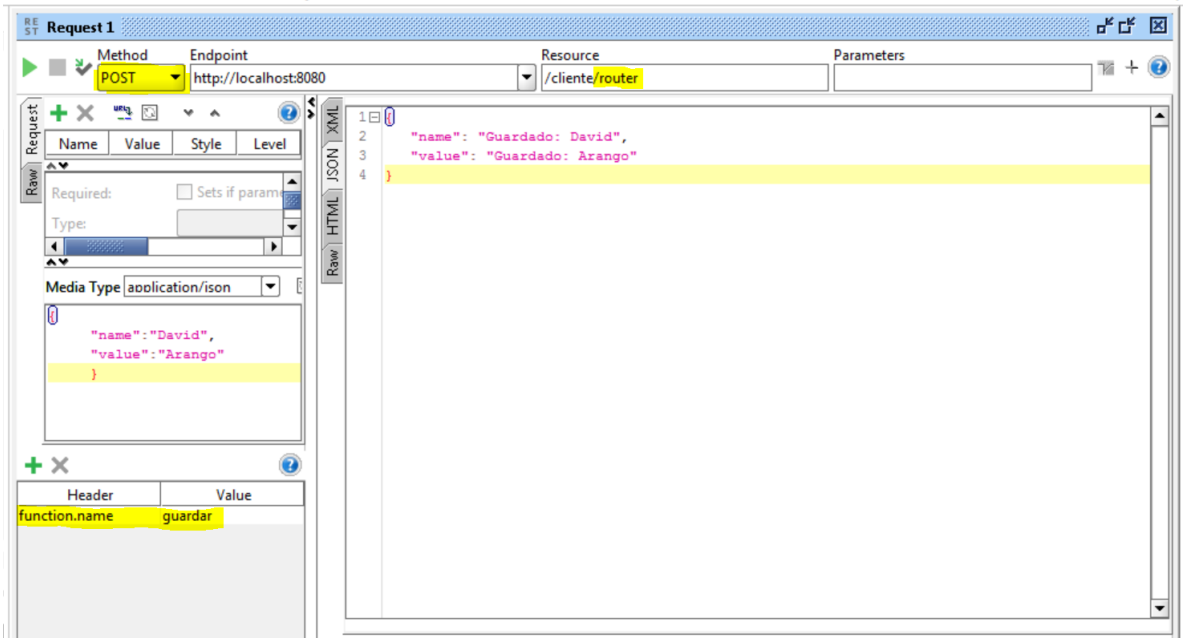
Como ejecutar el microservicio construido con Spring Cloud Function local:

La aplicación, es una aplicación Spring Boot con Maven, por lo cual simplemente se importa en el editor preferido y dar Run sobre el archivo Application.java.

Como consumir las funciones:

1. Siempre se debe utilizar el método POST
2. Se debe enviar un header http (function.name:guardar) o (function.name:saludar) con el objetivo de identificar la función que se requiere llamar.
3. El recurso que se debe invocar es router, internamente spring invocara una función de enrutamiento, la cual utilizara (function.name) para invocar la función solicitada.



Como ejecutar el microservicio en una lambda AWS:

Crear una lambda nueva, sin configuraciones adicionales, para este tutorial se supone que se tiene conocimiento sobre los roles de ejecución.

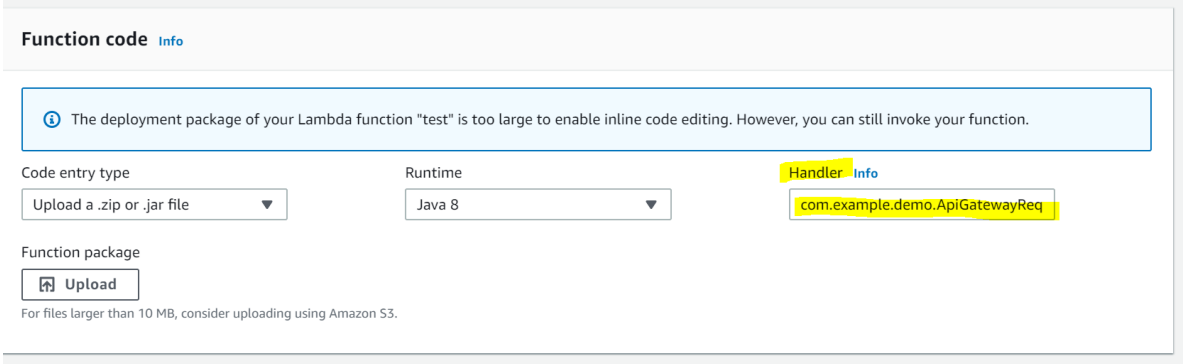
Configurar el handler en la lambda (Paquete.Clase :: Método):
com.example.demo.ApiGatewayRequestHandler::handleRequest

Nota:

Fue necesario personalizar este handler para el manejo de evento del ApiGateway, debido a que la librería de Spring Cloud Function en la versión 2.1.1-release y en el milestone 3.0M. Presentan errores al momento de parsear los objetos Json, haciendo uso del enrutamiento de las funciones. Cuando se utiliza org.springframework.cloud.function.adapter.aws.
SpringBootApiGatewayRequestHandler.

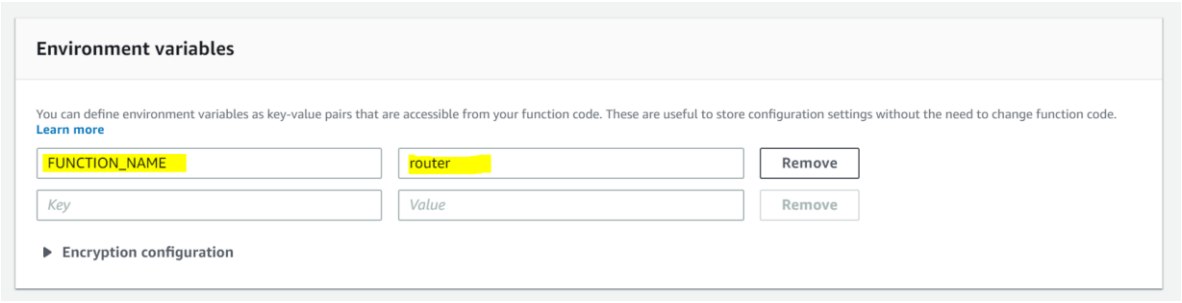
Esta fue la solicitud a Spring Project para que se haga la corrección del error:

https://github.com/spring-projects/spring-integration/pull/2635

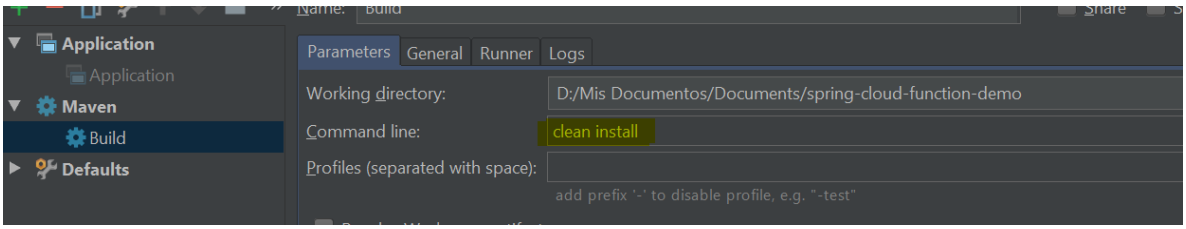


Adicionalmente siempre es necesario configurar la siguiente variable de entorno:

FUNCTION_NAME:router



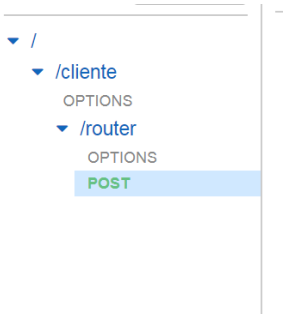
Para generar el paquete utilice el siguiente comando:



El siguiente paso es la configuración del APIGateway:

Para este tutorial se da por hecho el previo conocimiento del ApiGateway de AWS.

Se debe mantener la misma estructura de los recursos en ambiente local, por si es necesarios hacer una migración de la aplicación a contenedores.



Al momento de configurar el recurso tener en cuenta lo siguiente:

Es importante activar el lambda proxy, para que al momento de llegar la petición a la función podamos acceder a los headers y demás elementos del evento.