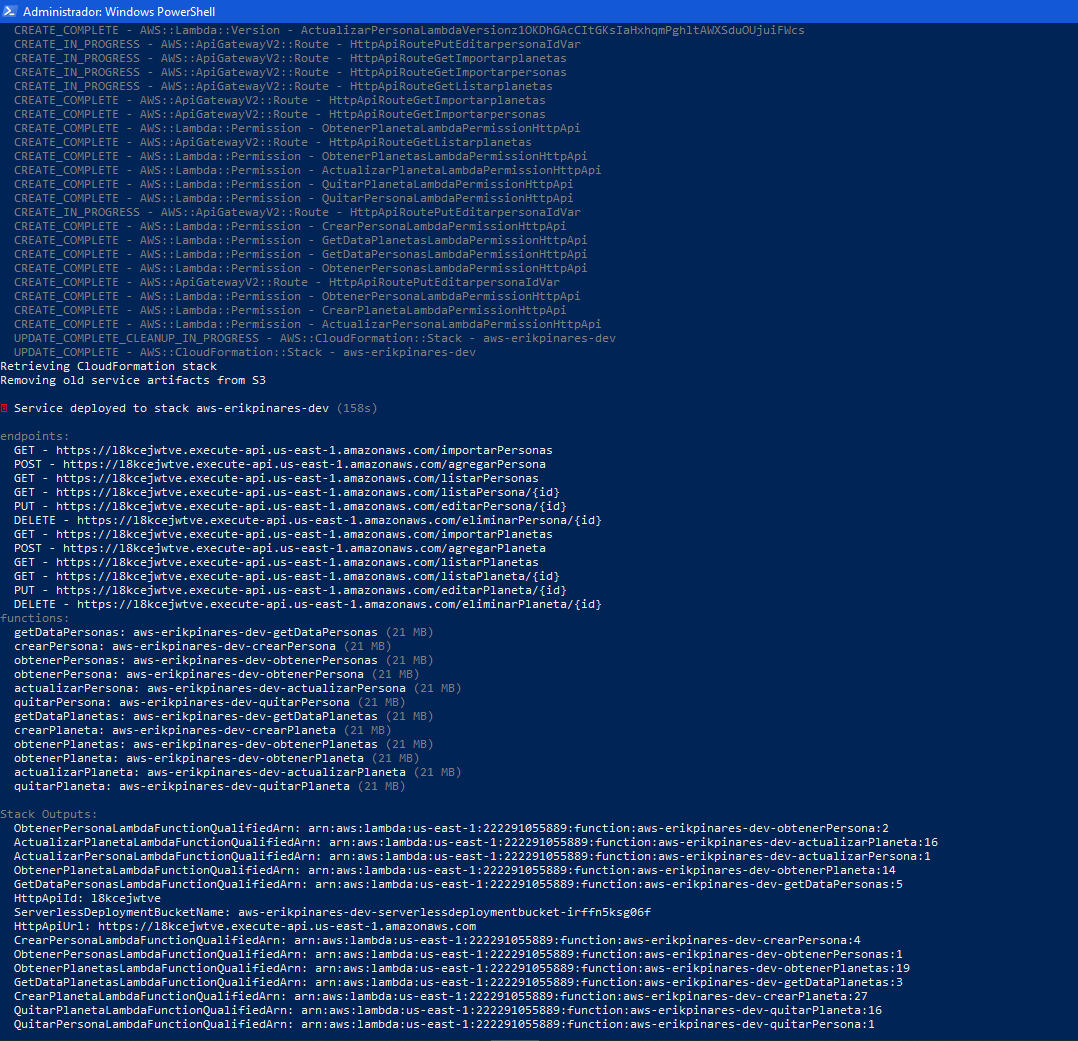
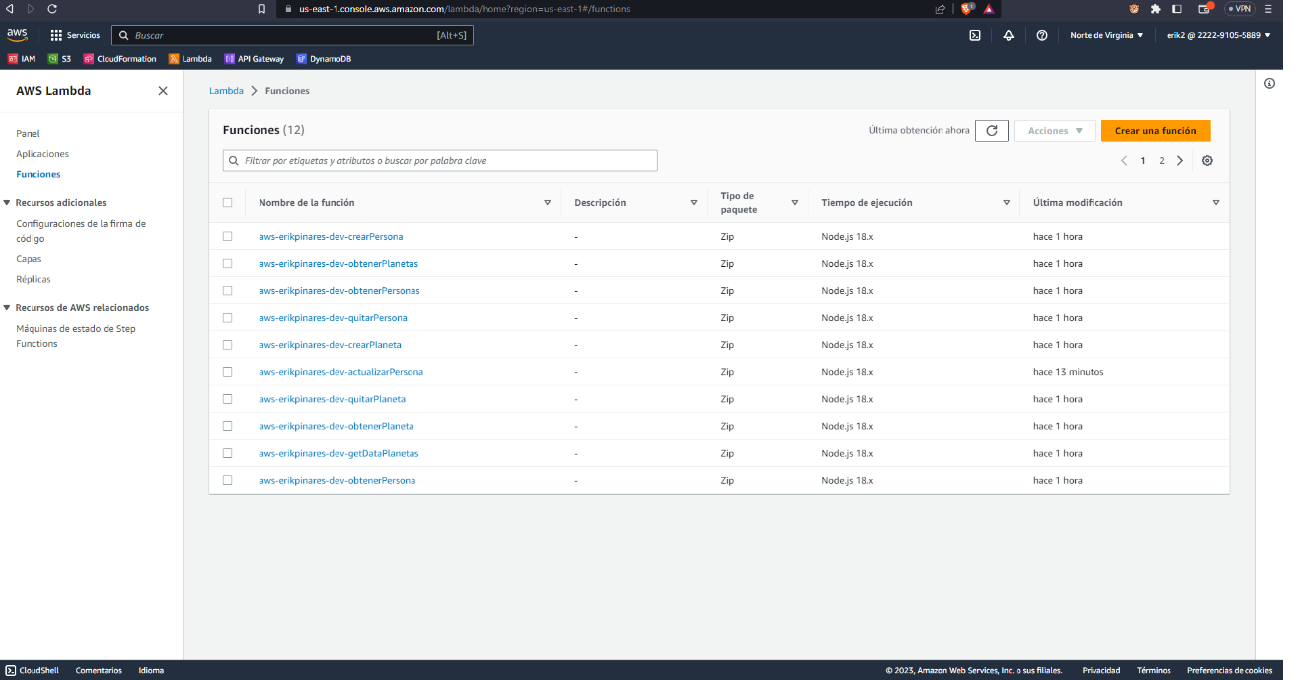
**DOCUMENTACIÓN RETO TÉCNICO**

1. **Crear un Api en NodeJS con el Framework Serveless**

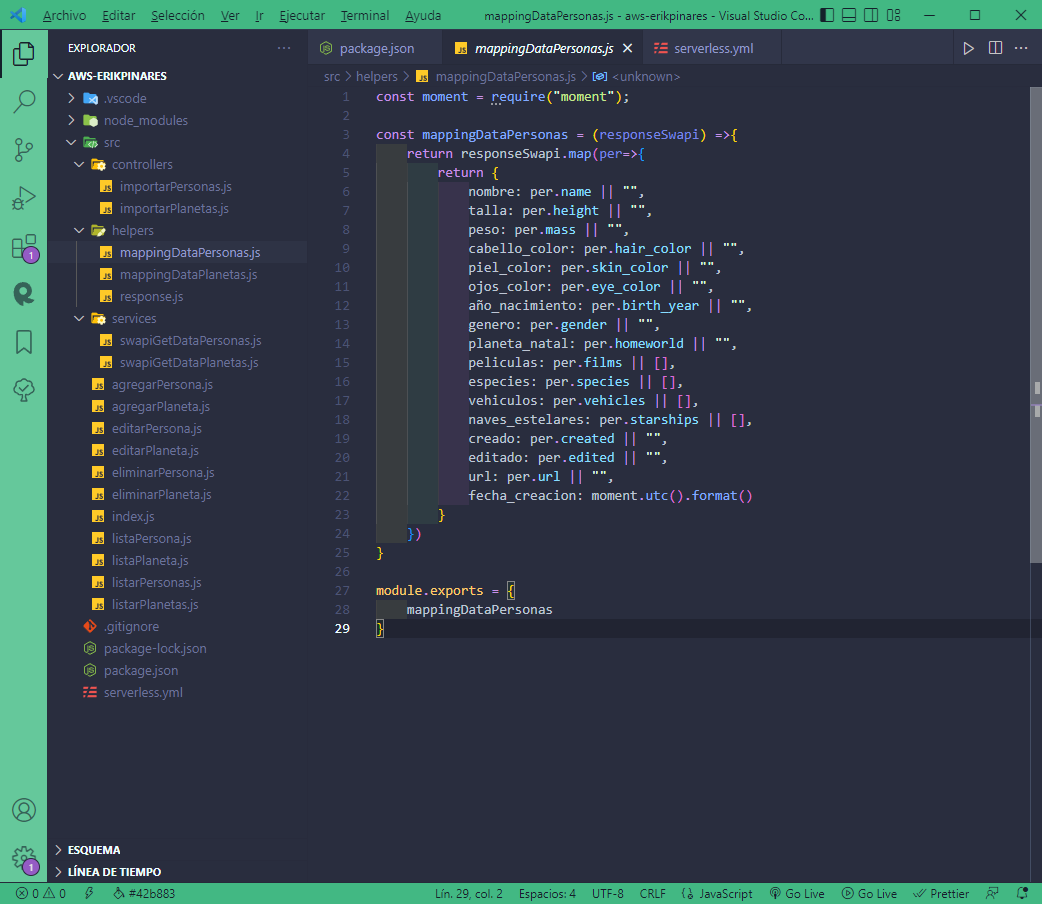
Como se puede visualizar en la imagen, el despliegue se realizó con éxito sin errores.



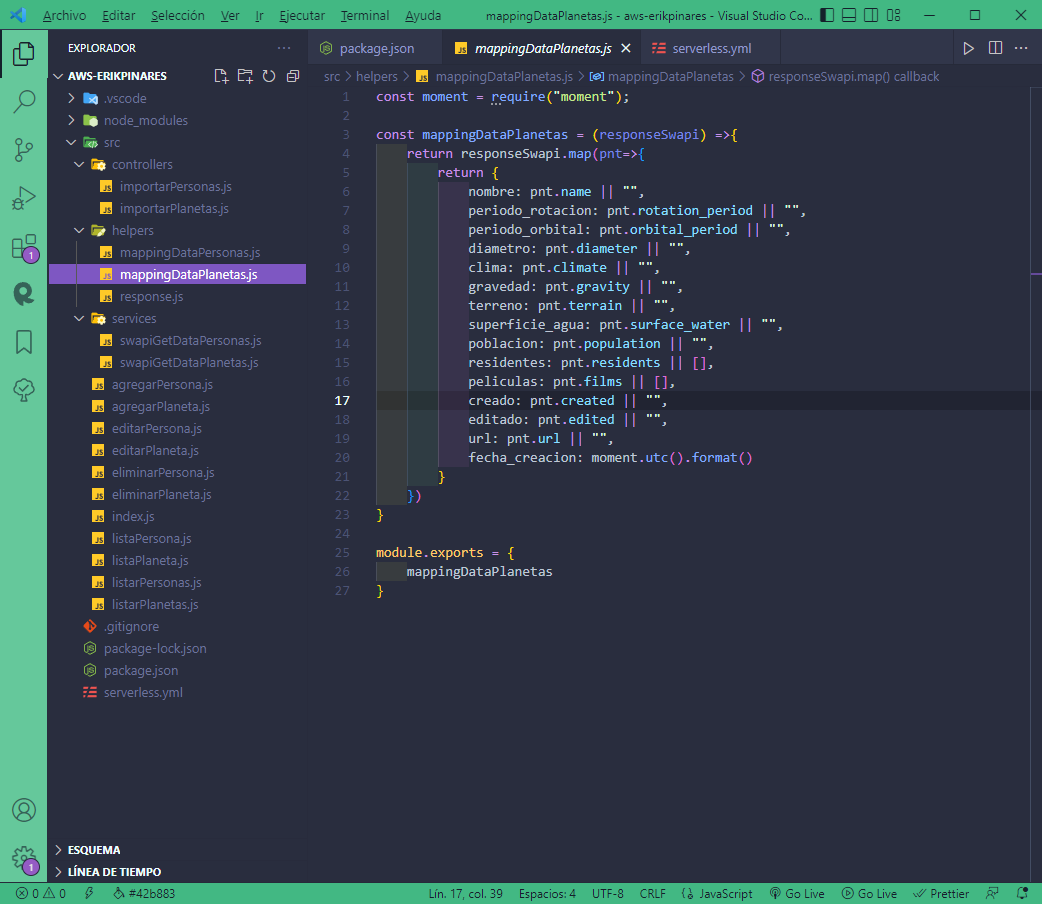
En la siguiente imagen se aprecian las funciones creadas en AWS después del deploy con serveless:



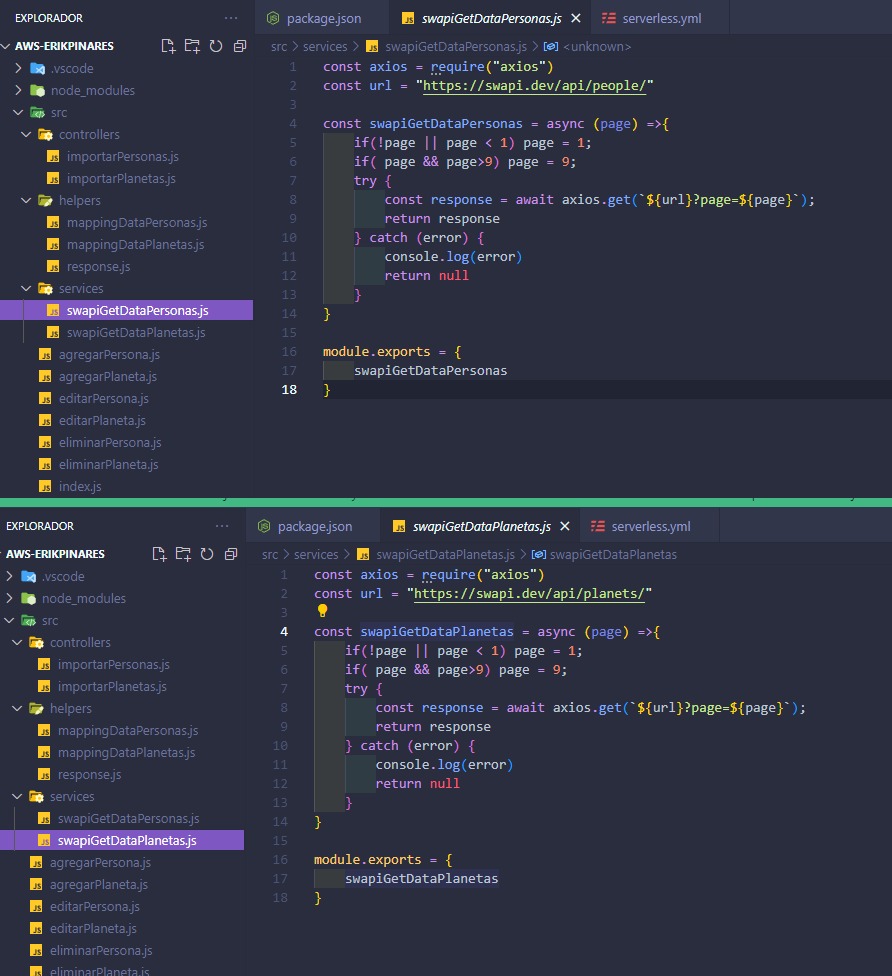
1. **Como se aprecia en la imagen, se realizó el mapeo de todos los nombres de atributos de inglés al español**
2. Modelo: Personas



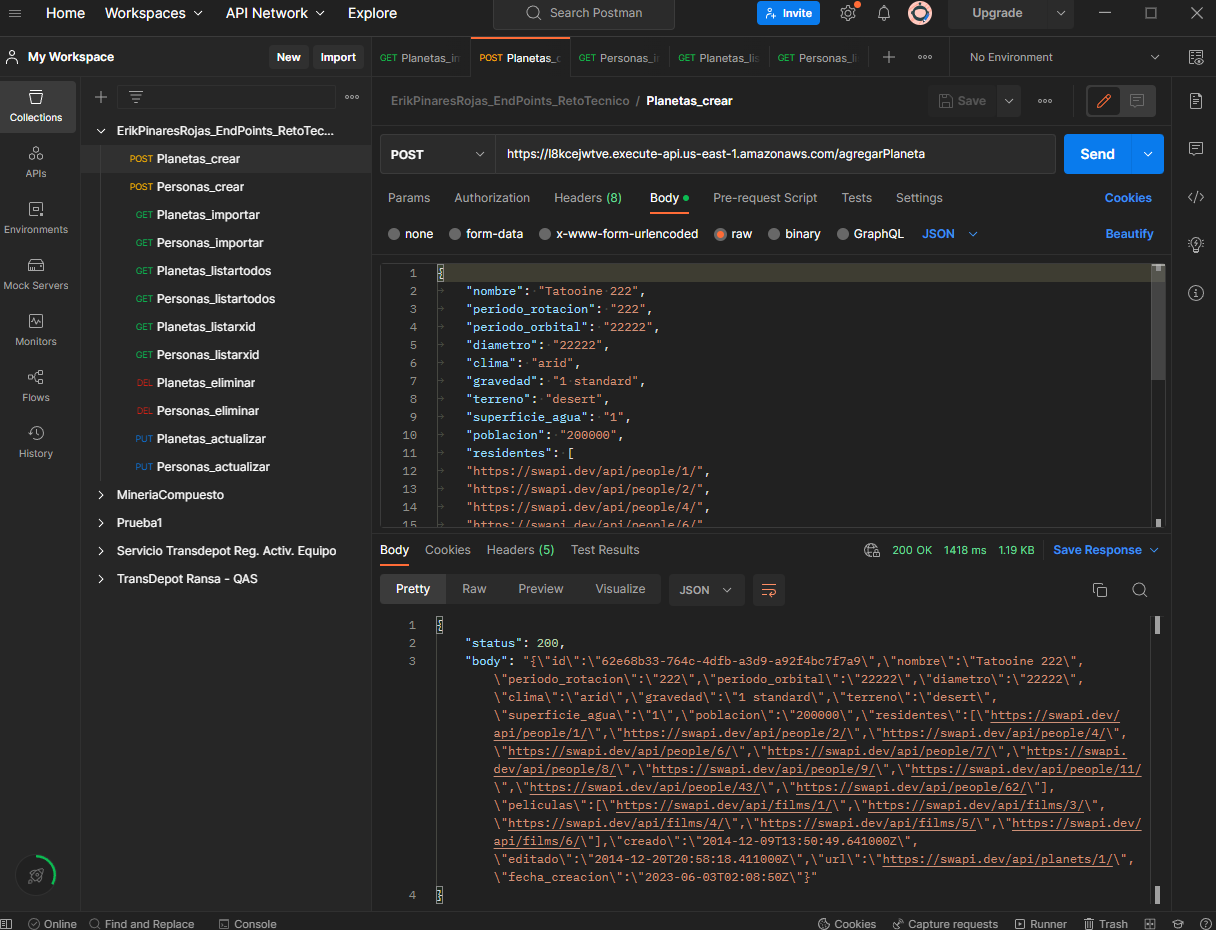
1. Modelo: Planetas

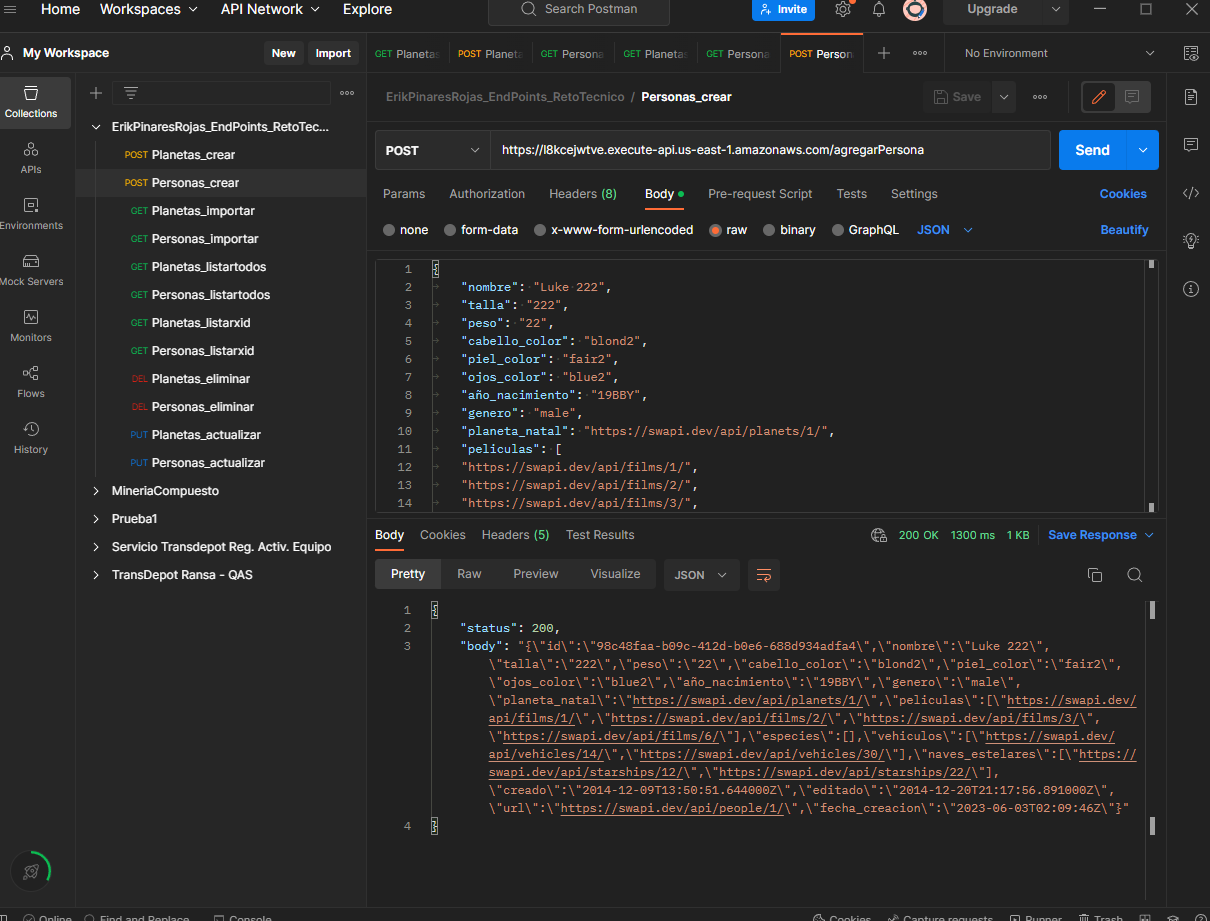


1. **Como se aprecia en la imagen, se usó el módulo de Axios para consumir el API de StarWars, utilizando dos modelos de dicha Api (people y planets)**:

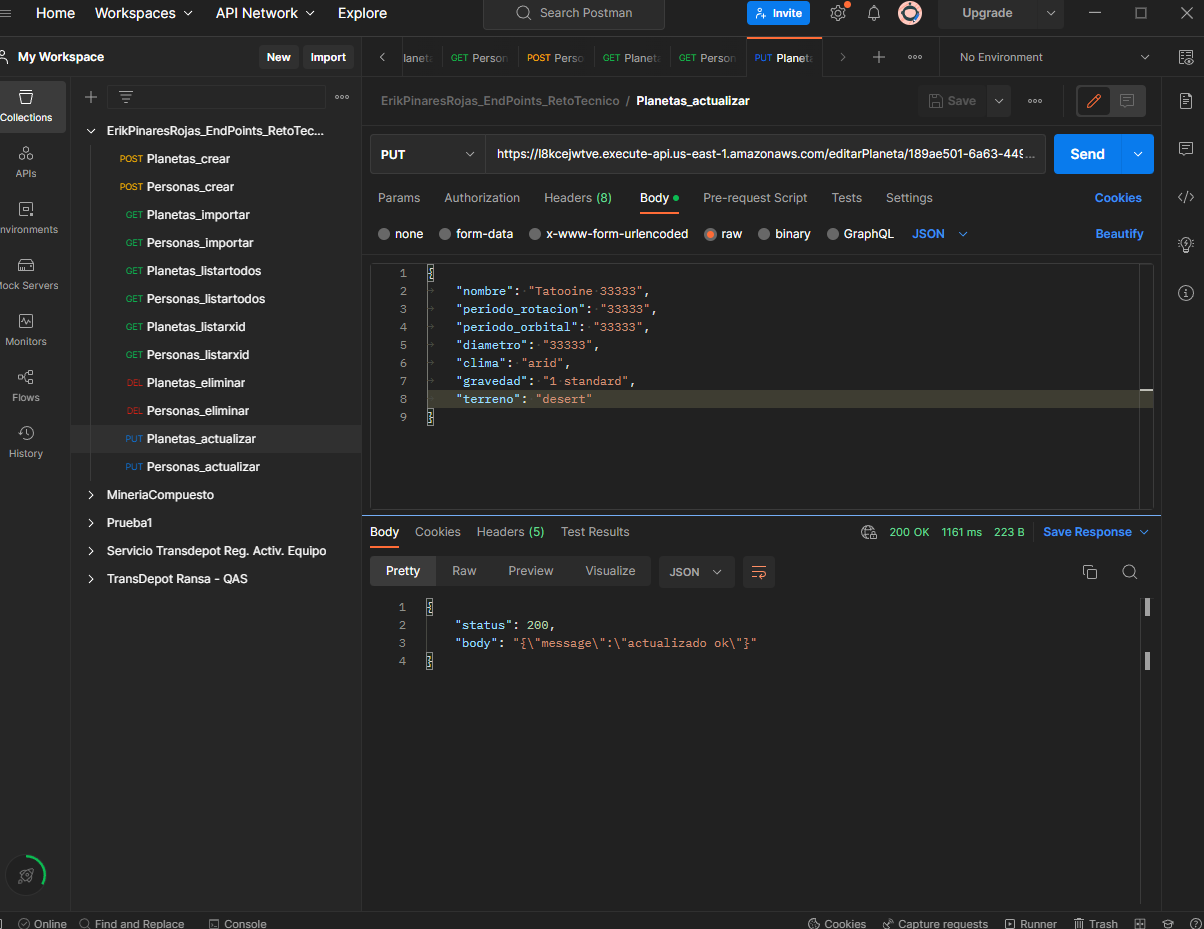


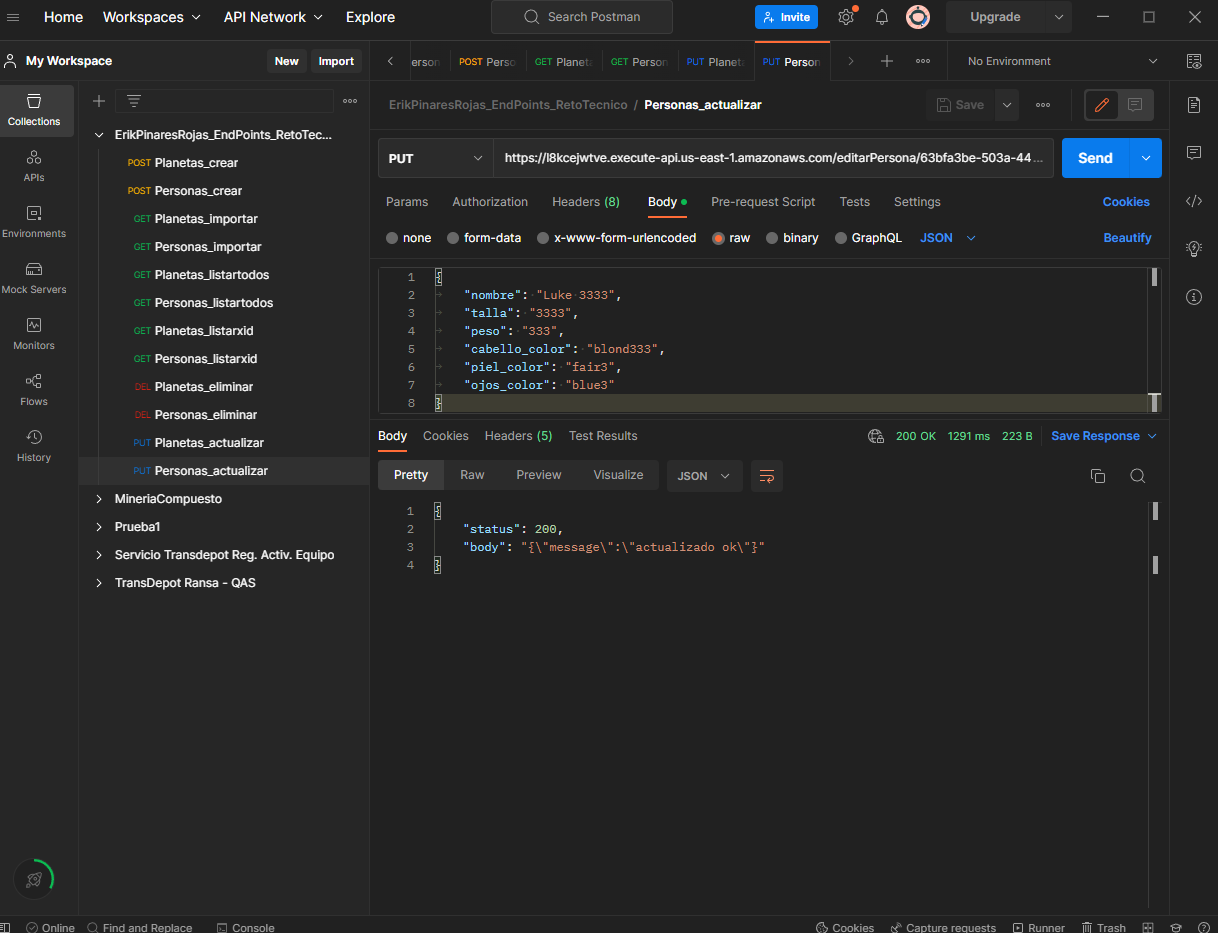
1. **Crear un modelo de su selección mediante el uso de endpoint post.**Se realizaron varios endpoint usando la bd DynamoDB.

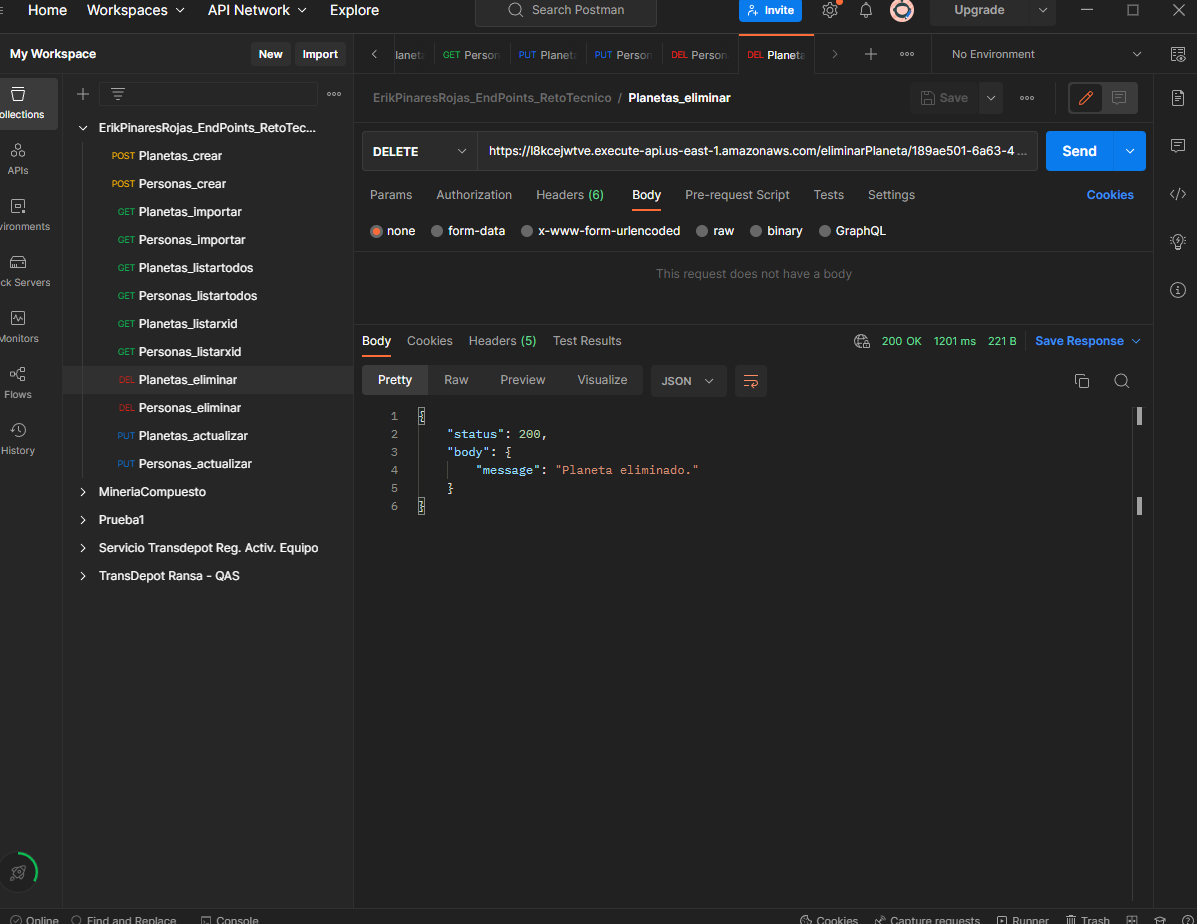


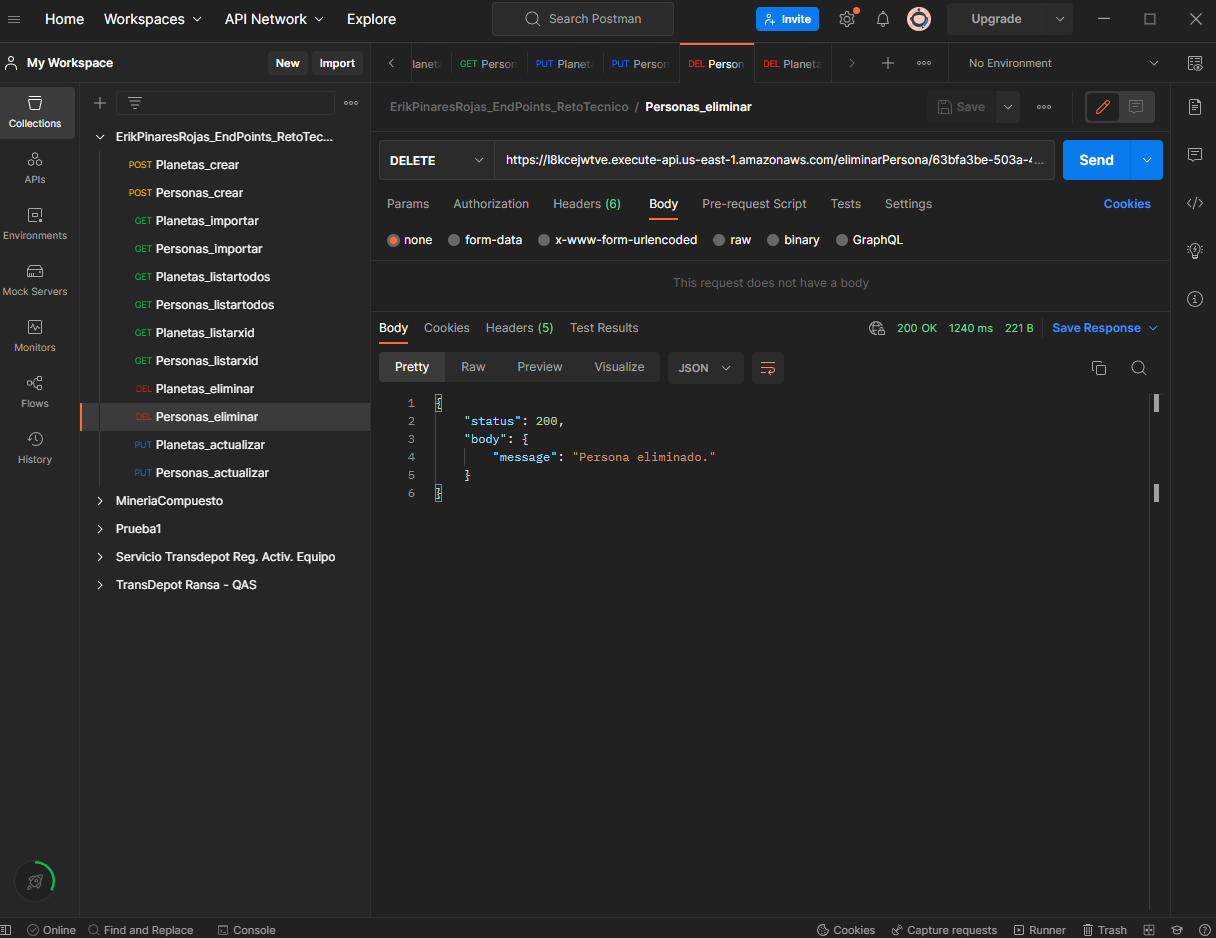


Adicionalmente también se utilizaron los métodos PUT y DELETE para actualizar y eliminar datos.



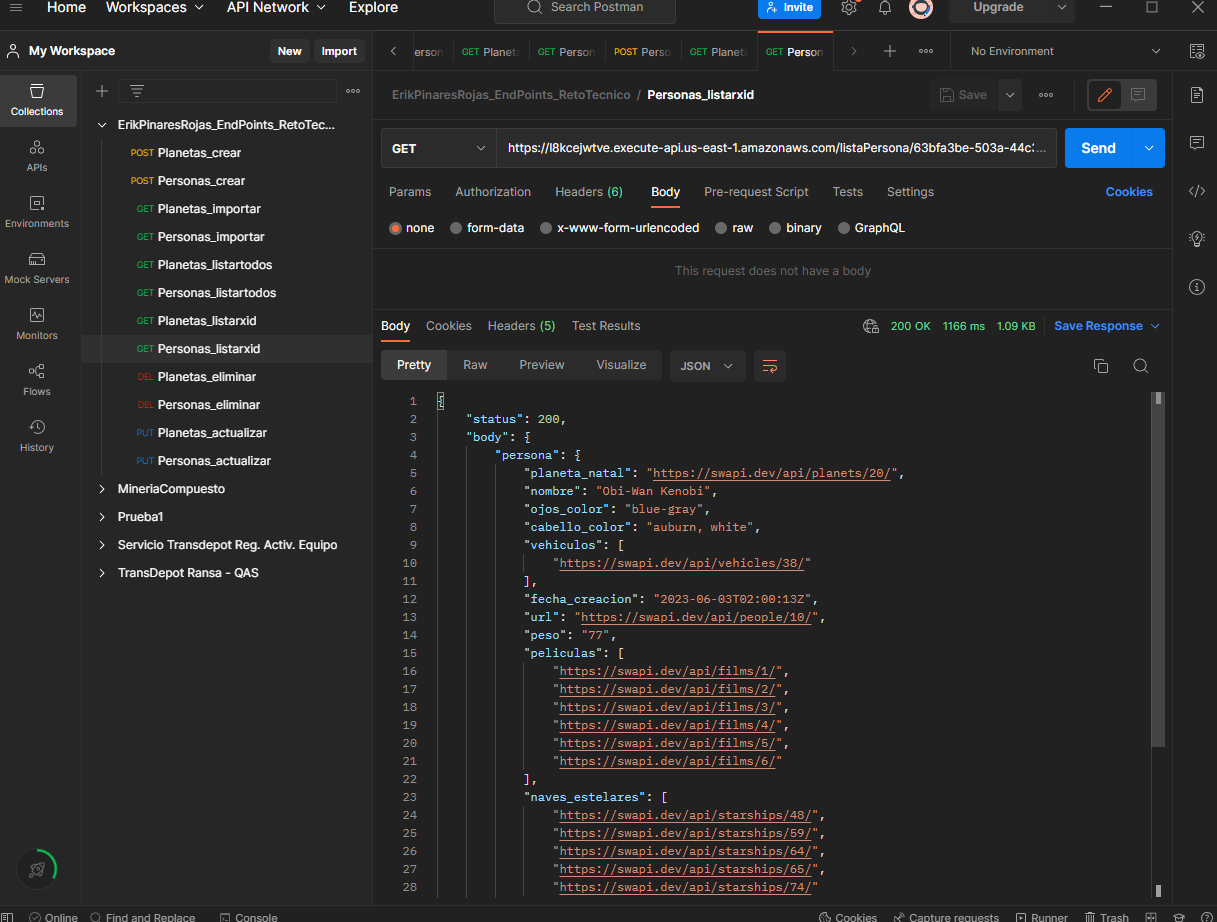
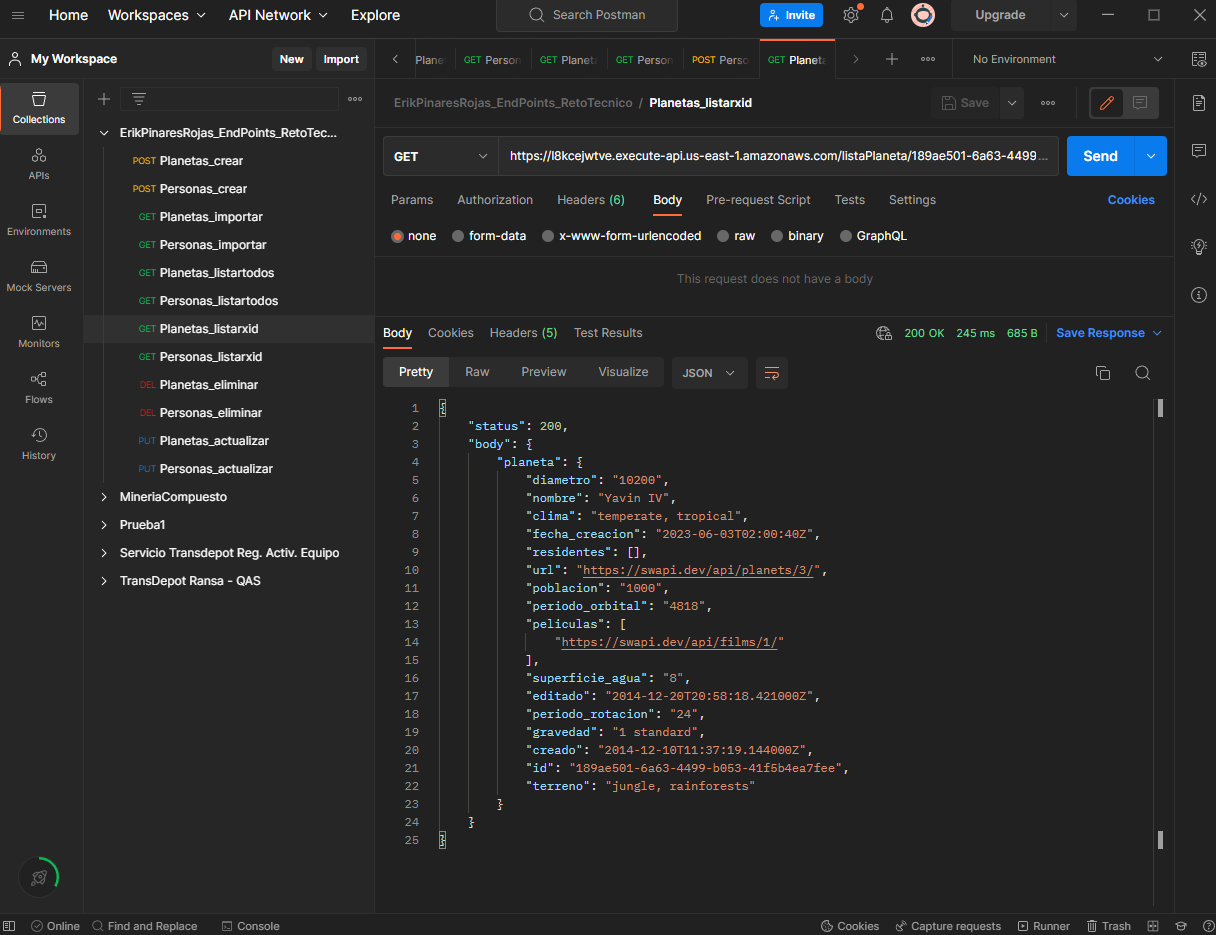
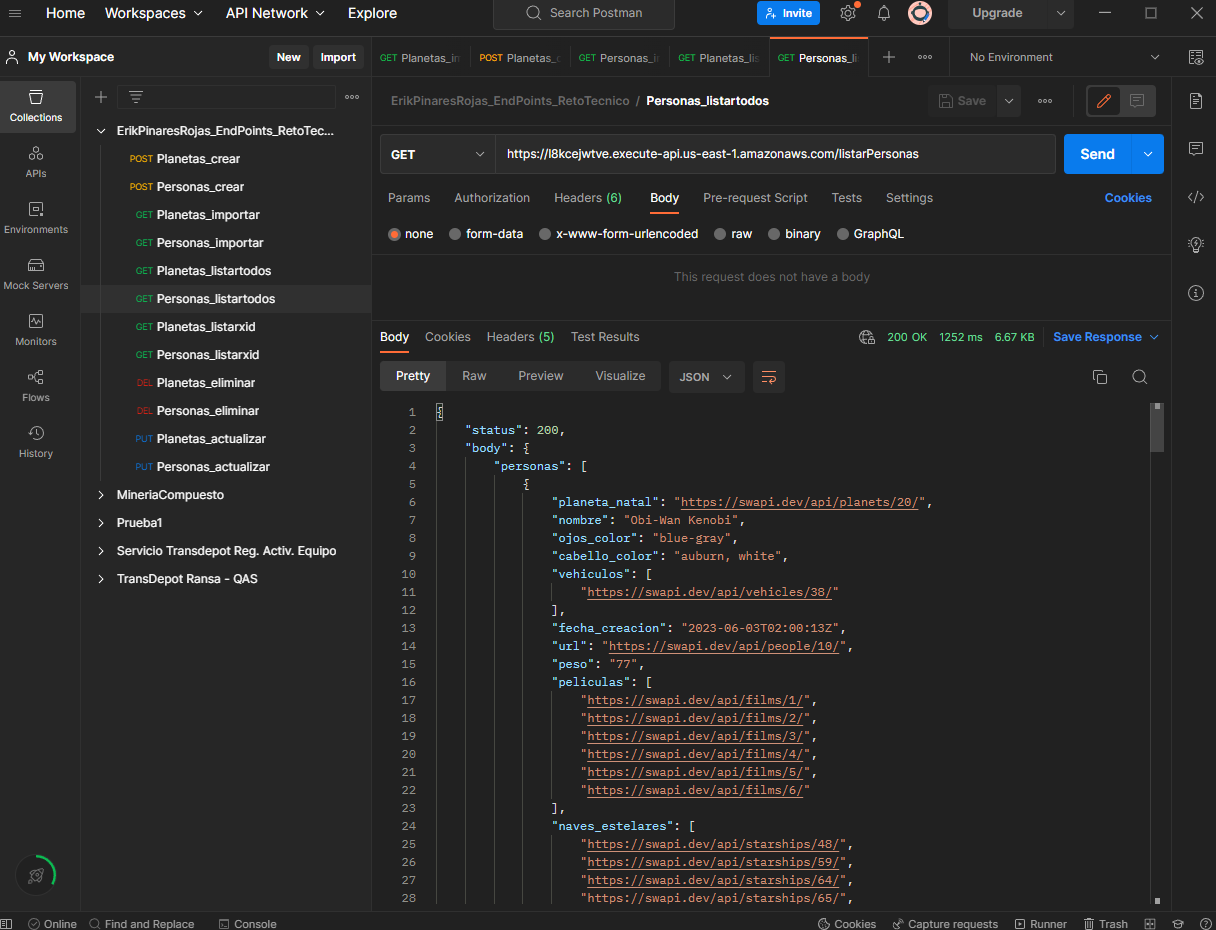
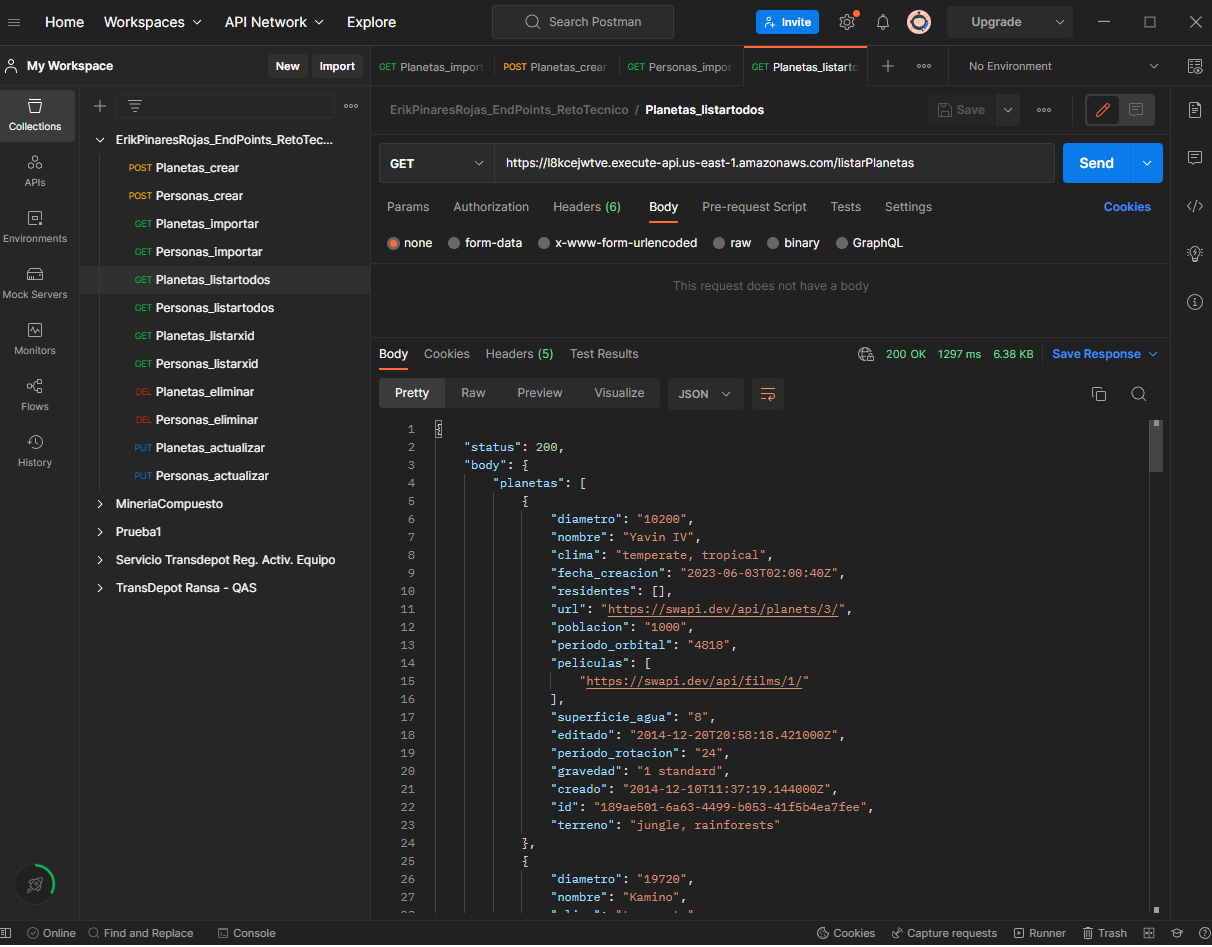
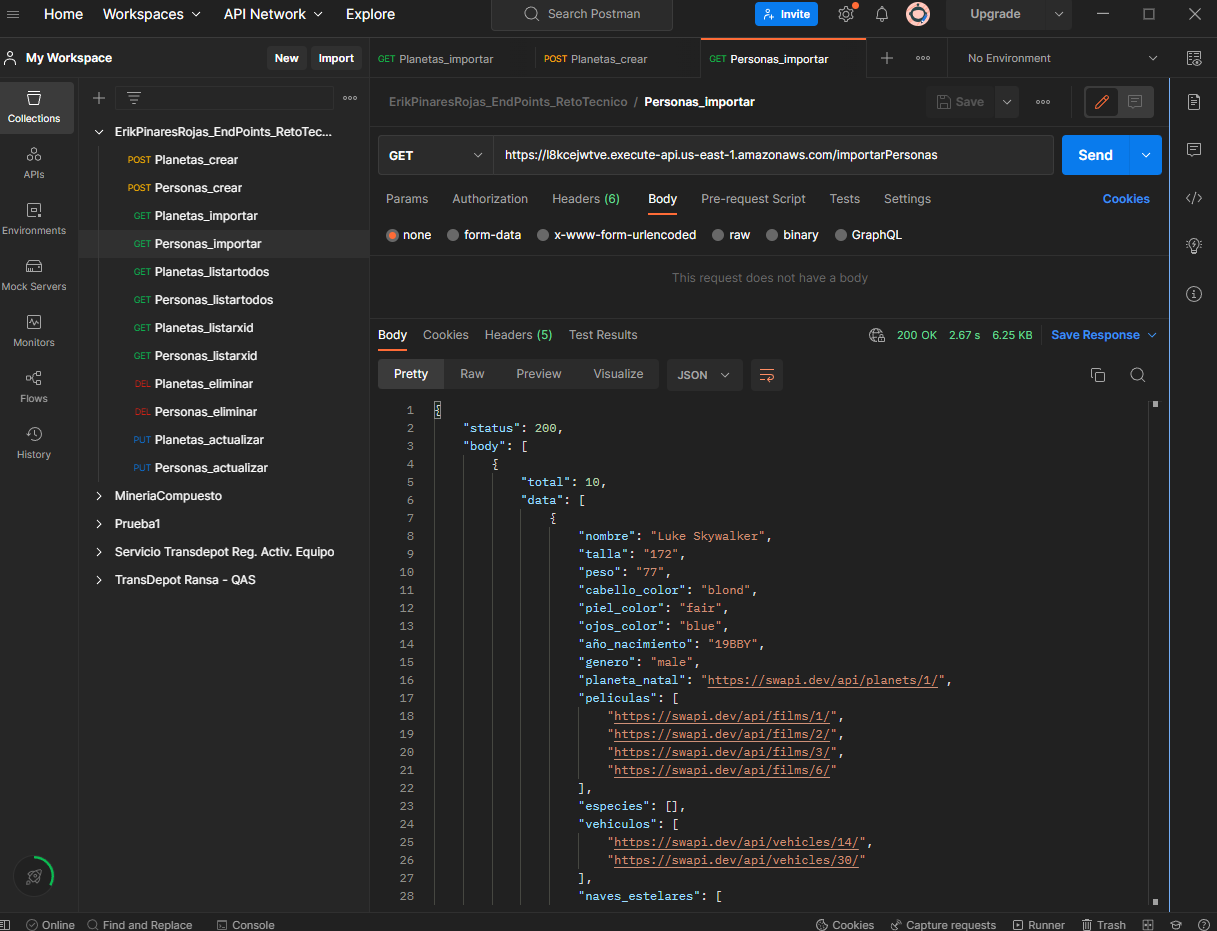
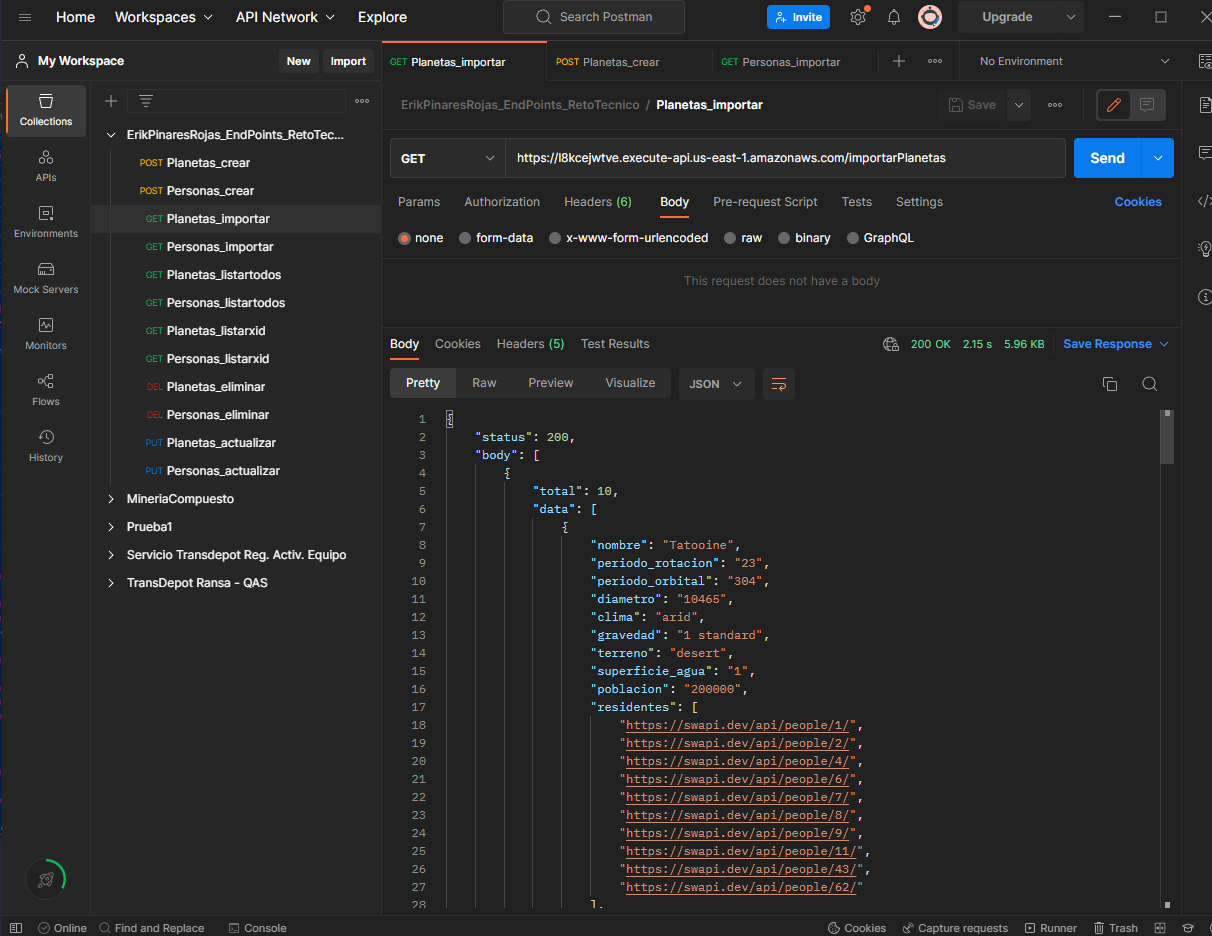


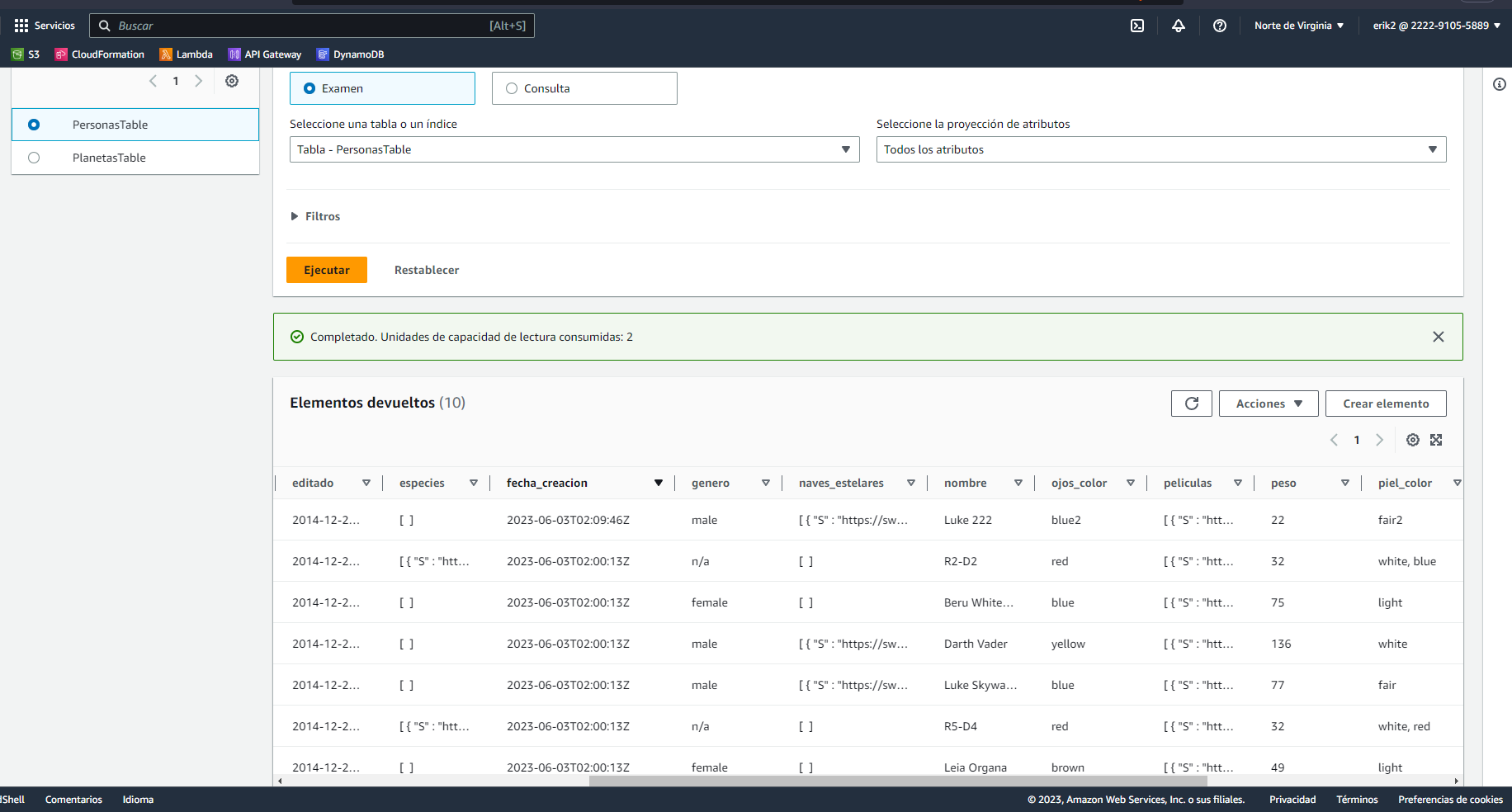
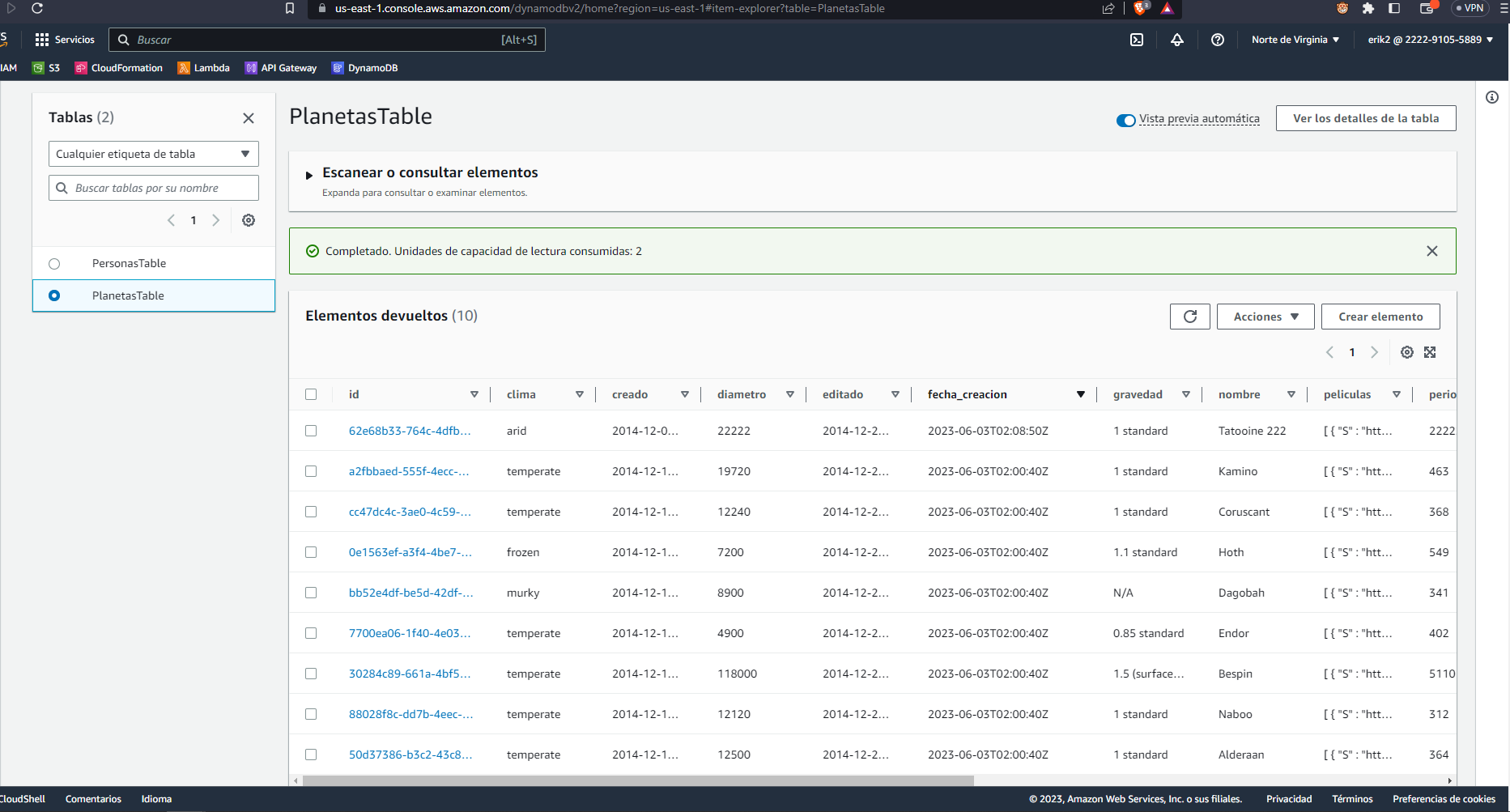




1. **Crear un ednpoint GET para mostrar la data almacenada**

Como se muestra en las imágenes tenemos varios endpoint GET para mostrar la información guardada en la base de datos DynamoDB.



A continuación se muestra imagen de la base de datos: 

1. **Con la finalidad de verificar fácilmente el funcionamiento del uso de las apis desarrolladas, se adjunta archivo POSTMAN con todos los datos de prueba y endpoints. Este archivo se encuentra en el correo enviado con el código fuente como también en la página de GITHUB.**



1. **Buenas prácticas:**

