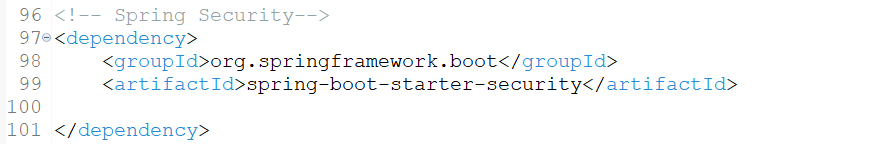
**Exercise 12: Online Bookstore - Securing RESTful Endpoints with Spring Security**

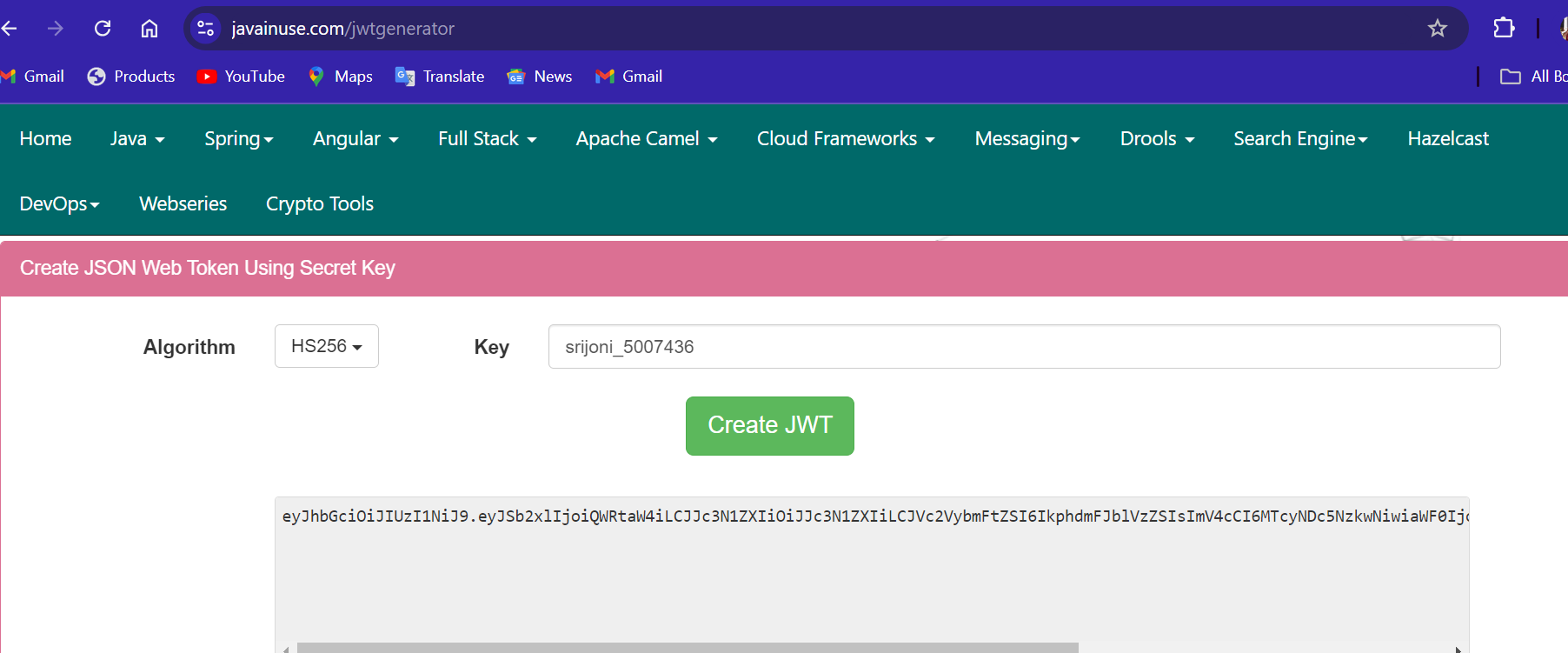
**Introduction**

This document outlines the implementation of a secure RESTful API for an online bookstore using Spring Security and JWT (JSON Web Token) authentication. The API provides endpoints for managing books and customers, ensuring that sensitive operations are protected from unauthorized access.

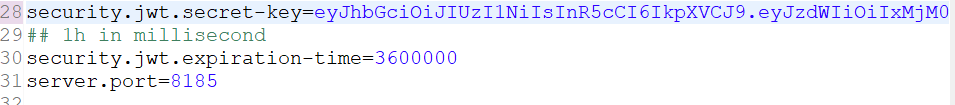
Included Spring Security dependency in the pom.xml file.



Went to <https://jwt.io> site and under Debugger under Verify signature, copied the JWT Key:



Modified application.properties:



* **Security Configuration:** A SecurityFilterChain is configured to define security rules.
  + The @EnableWebSecurity annotation enables Spring Security.
  + HttpSecurity is used to configure request authorization, session management, and authentication providers.
  + CORS is enabled to allow cross-origin requests.
  + The JwtAuthenticationFilter is added to the filter chain to handle JWT-based authentication.

**JWT Authentication**

**JWT-Based Authentication and Authorization implemented.**

Added Dependencies for JWT Support:

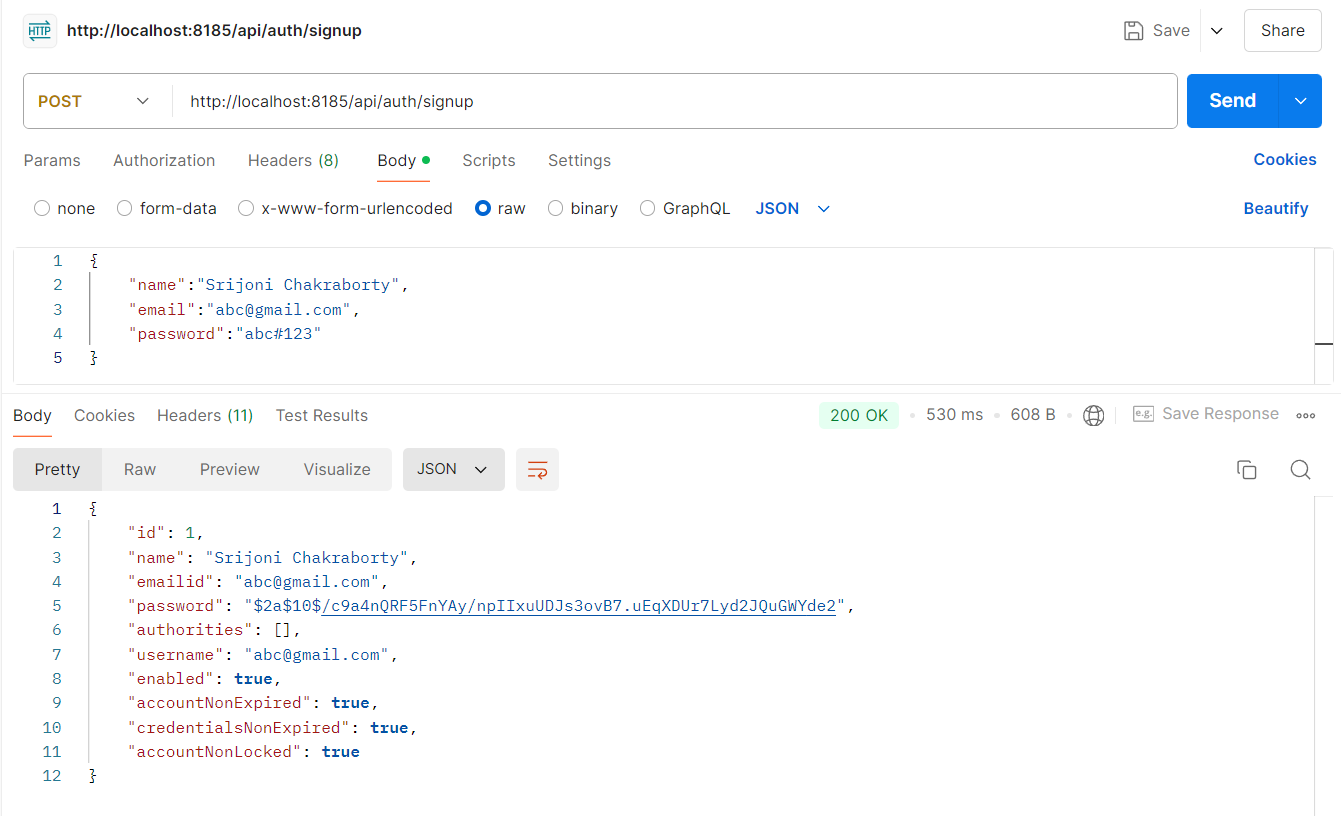
****

**CORS Handling**

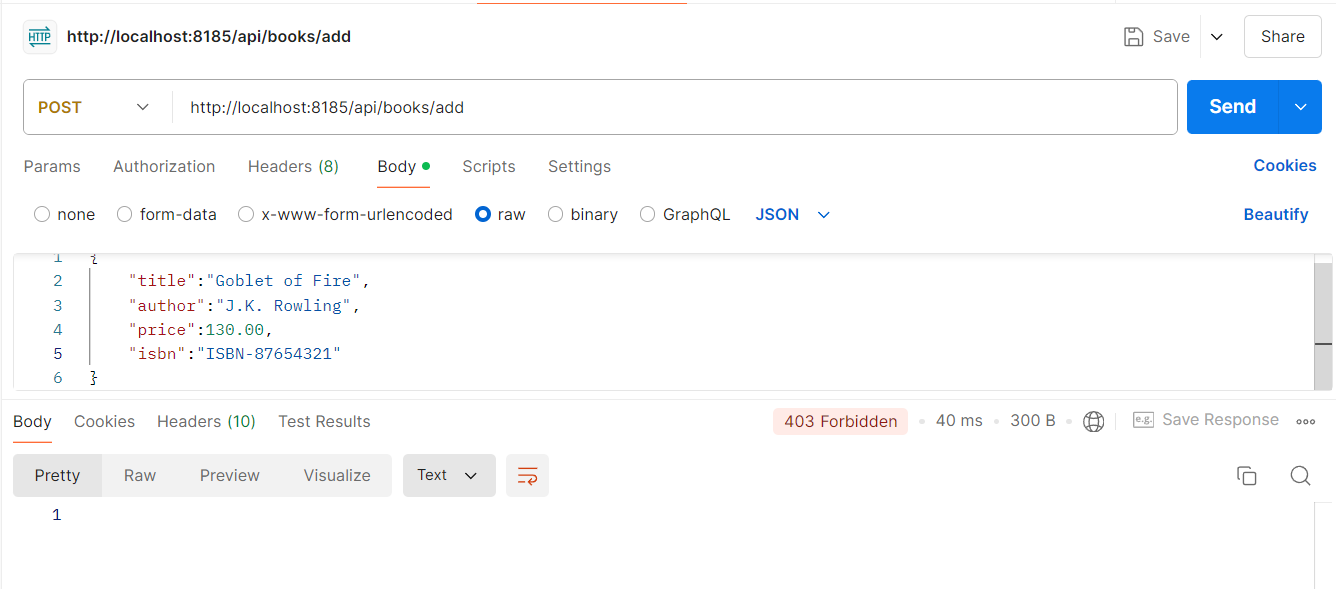
* CORS is configured to allow requests from specific origins.
* The CorsConfigurationSource bean is used to define the allowed origins, methods, and headers.

**OUTPUT:**

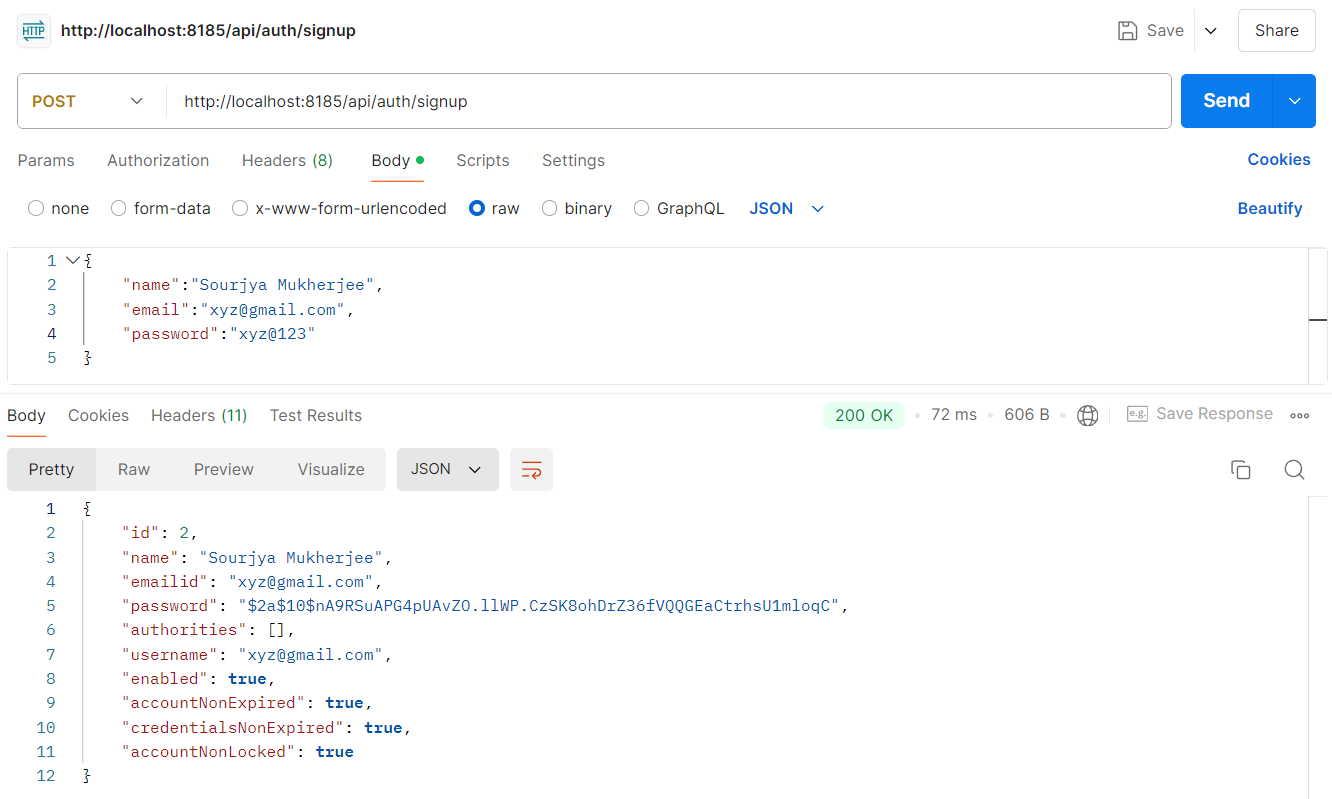
**Signup:**



**403 Forbidden:**

****

**Signup Again:**

****