**Exercise 1: Implementing Centralized Authentication with OAuth 2.1/OIDC**

**Pom.xml**

<?xml version="1.0" encoding="UTF-8"?>  
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
 <parent>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-parent</artifactId>  
 <version>3.5.3</version>  
 <relativePath/> <!-- lookup parent from repository -->  
 </parent>  
 <groupId>com.example</groupId>  
 <artifactId>oauth2demo</artifactId>  
 <version>0.0.1-SNAPSHOT</version>  
 <name>Microservices</name>  
 <description>Microservices</description>  
 <url/>  
 <licenses>  
 <license/>  
 </licenses>  
 <developers>  
 <developer/>  
 </developers>  
 <scm>  
 <connection/>  
 <developerConnection/>  
 <tag/>  
 <url/>  
 </scm>  
 <properties>  
 <java.version>17</java.version>  
 </properties>  
 <dependencies>  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-oauth2-client</artifactId>  
 </dependency>  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-security</artifactId>  
 </dependency>  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-web</artifactId>  
 </dependency>  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-test</artifactId>  
 <scope>test</scope>  
 </dependency>  
 <dependency>  
 <groupId>org.springframework.security</groupId>  
 <artifactId>spring-security-test</artifactId>  
 <scope>test</scope>  
 </dependency>  
 </dependencies>  
 <build>  
 <plugins>  
 <plugin>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-maven-plugin</artifactId>  
 </plugin>  
 </plugins>  
 </build>  
</project>

**Application.yml**

spring:  
 security:  
 oauth2:  
 client:  
 registration:  
 google:  
 client-id: 302979113374-h5hbkuc2qg78kvotknphemvkjg93u47a.apps.googleusercontent.com  
 client-secret: GOCSPX-hG-\_ygfdZ2i5IGguBUphG24YuUda  
 scope:  
 - openid  
 - profile  
 - email  
 redirect-uri: "{baseUrl}/login/oauth2/code/{registrationId}"  
 authorization-grant-type: authorization\_code  
 provider:  
 google:  
 authorization-uri: https://accounts.google.com/o/oauth2/v2/auth  
 token-uri: https://oauth2.googleapis.com/token  
 user-info-uri: https://openidconnect.googleapis.com/v1/userinfo  
 user-name-attribute: sub

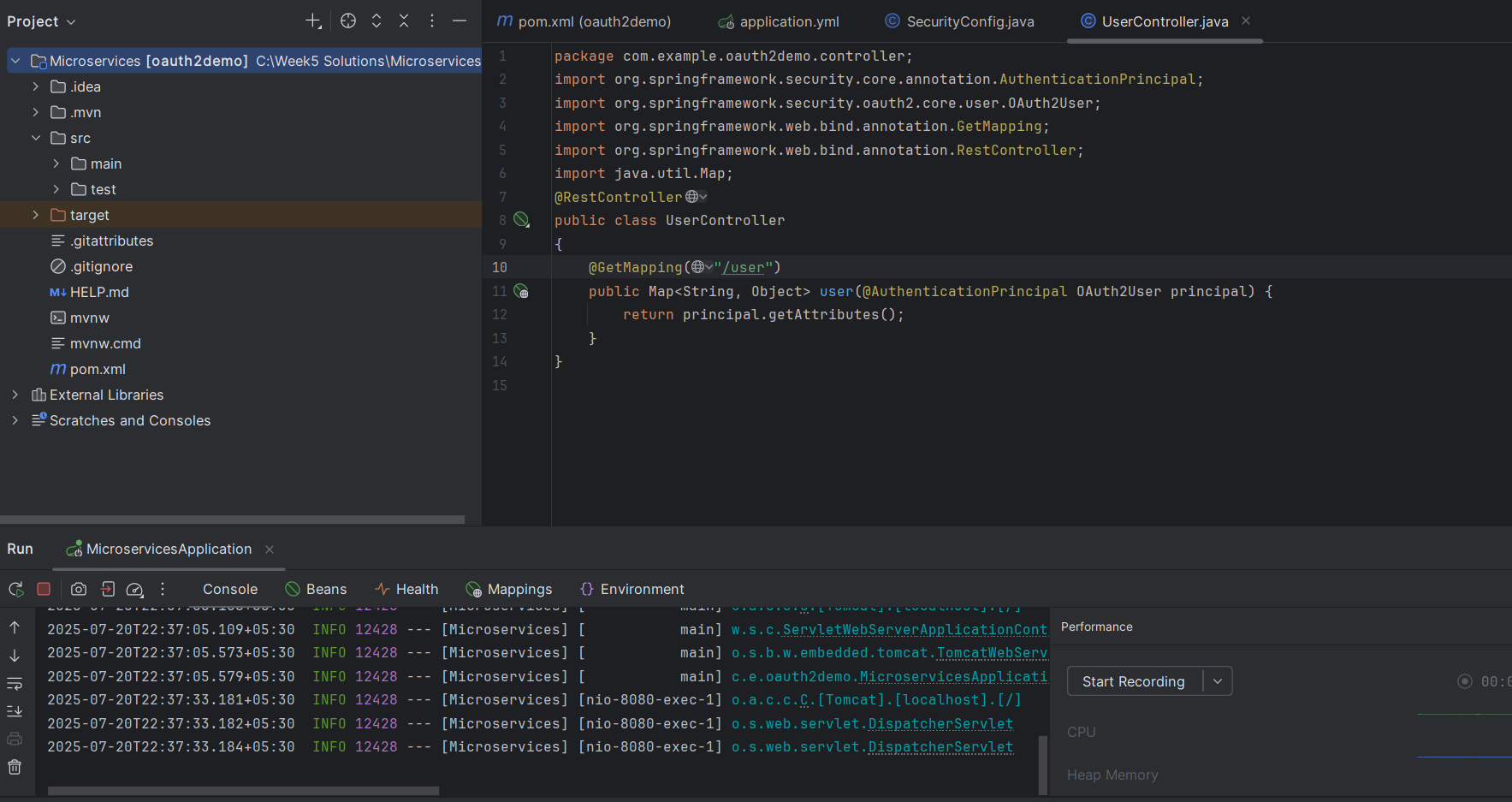
**SecurityConfig.java**

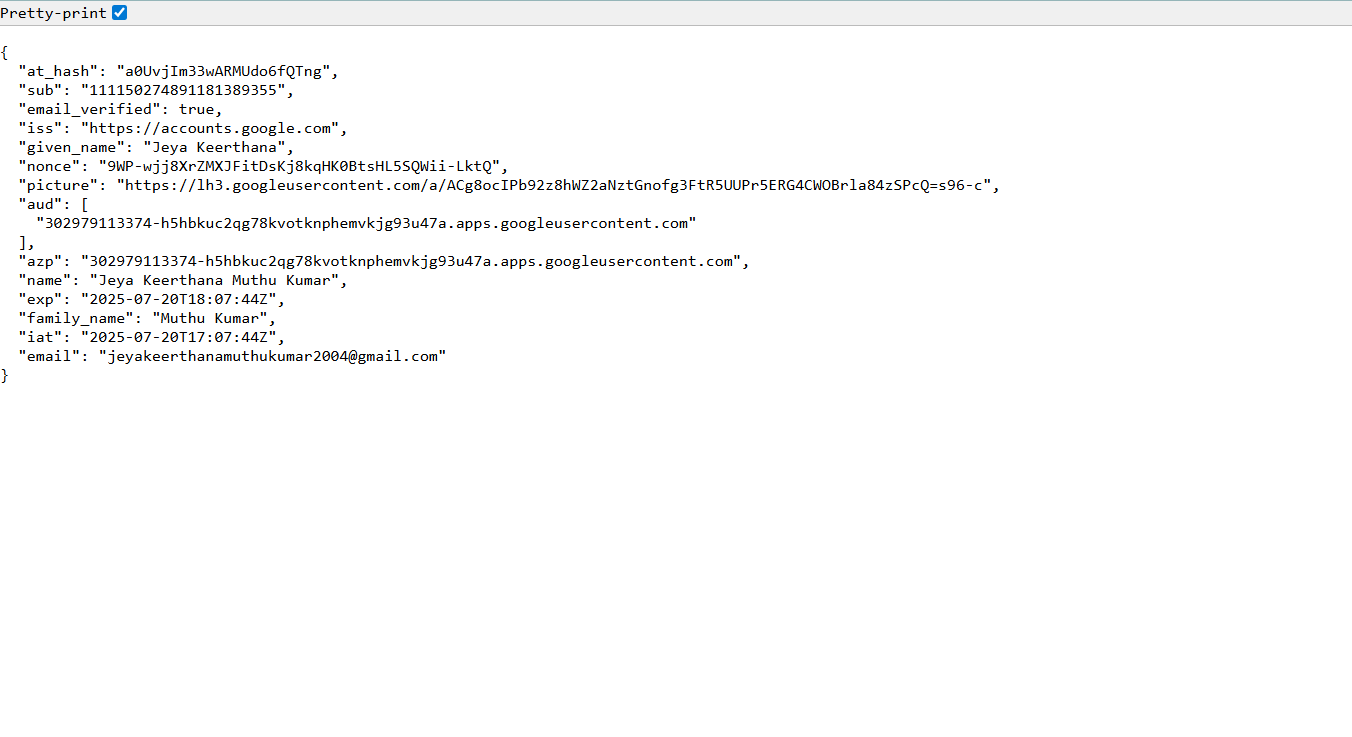
package com.example.oauth2demo.config;  
import org.springframework.context.annotation.Bean;  
import org.springframework.context.annotation.Configuration;  
import org.springframework.security.config.annotation.web.builders.HttpSecurity;  
import org.springframework.security.web.SecurityFilterChain;  
@Configuration  
public class SecurityConfig  
{  
 @Bean  
 public SecurityFilterChain securityFilterChain(HttpSecurity http) throws Exception  
 {  
 http  
 .authorizeHttpRequests(auth -> auth  
 .anyRequest().authenticated()  
 )  
 .oauth2Login();  
 return http.build();  
 }  
}

**UserController.java**

package com.example.oauth2demo.controller;  
import org.springframework.security.core.annotation.AuthenticationPrincipal;  
import org.springframework.security.oauth2.core.user.OAuth2User;  
import org.springframework.web.bind.annotation.GetMapping;  
import org.springframework.web.bind.annotation.RestController;  
import java.util.Map;  
@RestController  
public class UserController  
{  
 @GetMapping("/user")  
 public Map<String, Object> user(@AuthenticationPrincipal OAuth2User principal) {  
 return principal.getAttributes();  
 }  
}

**Output**





**Creating Microservices for account and loan**

**AccountController.java**

package com.cognizant.account.controller;  
import org.springframework.web.bind.annotation.\*;  
import java.util.Map;  
@RestController  
@RequestMapping("/accounts")  
public class AccountController  
{  
 @GetMapping("/{number}")  
 public Map<String, Object> getAccount(@PathVariable String number)  
 {  
 return Map.*of*(  
 "number", number,  
 "type", "savings",  
 "balance", 234343  
 );  
 }  
}

**LoanController.java**

package com.cognizant.loan.controller;  
import org.springframework.web.bind.annotation.\*;  
import java.util.Map;  
@RestController  
@RequestMapping("/loans")  
public class LoanController {  
  
 @GetMapping("/{number}")  
 public Map<String, Object> getLoan(@PathVariable String number) {  
 return Map.*of*(  
 "number", number,  
 "type", "car",  
 "loan", 400000,  
 "emi", 3258,  
 "tenure", 18  
 );  
 }  
}

**Output**

