**LAB -OAuth2 Password Grant Type – Authorization Server**

1. Create a Spring Boot Starter Project
2. Add Spring Web , Spring Security, Cloud Bootstrap as Dependency
3. Add the following Dependency to pom.xml and update the project

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-oauth2</artifactId>

<version>2.2.5.RELEASE</version>

</dependency>

1. Register two Beans BCryptPasswordEncoder
2. Create a Class by Extending AuthorizationServerConfigurerAdapter
3. Add the annotations @Configuration,@EnableAuthorizationServer
4. Autowire authenticationmanager and bcryptpasswordencoder
5. Override the method Configure with AuthorizationServerSecurityConfigurer

security.checkTokenAccess("permitAll()");

1. Override the method Configure with ClientDetailsServiceConfigurer

clients.inMemory().withClient("myclient")

.authorizedGrantTypes("password")

.scopes("read","write").secret(encoder.encode("pass123"))

.accessTokenValiditySeconds(3000)

1. Override the method configure with AuthorizationServerEndpointsConfigurer

endpoints.authenticationManager(manager);

1. Create a Java class extending WebSecurityConfigurerAdapter
2. Autowire BcryptPassword Encoder
3. Add the @Configuration Annotation
4. Override the Configure method with HttpSecurity

http.anonymous().disable().csrf().disable()

.authorizeRequests()

.antMatchers("/oauth/token").permitAll();

1. Override the Configure method with AuthenticationManagerBuilder
2. Override authenticationManagerBean add the @Bean annotation
3. Register a Bean AuthenticationManager with @Bean Annotation

=======================================================

**LAB -OAuth2 Password Grant Type – Resource Server**

1. Create a Spring Boot Starter Project for Resource Server
2. Add Spring Web , Spring Security, Cloud Bootstrap,Lombok as Dependency
3. Add the following Dependency to pom.xml and update the project

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-oauth2</artifactId>

<version>2.2.5.RELEASE</version>

</dependency>

1. Add the Annotation @EnableResourceServer on the main class
2. Add the Annotations @EnableGlobalMethodSecurity(jsr250Enabled = **true**,prePostEnabled = **true**)
3. Create a Controller with @RestController
4. Create methods with @RolesAllowed(value = "ROLE\_ADMIN") and @PreAuthorize("#oauth2.hasScope('read')")
5. Update the application.yml file

server:

port: 9090

security:

oauth2:

resource:

token-info-uri: http://localhost:4040/oauth/check\_token

client:

client-id: myclient

client-secret: pass123