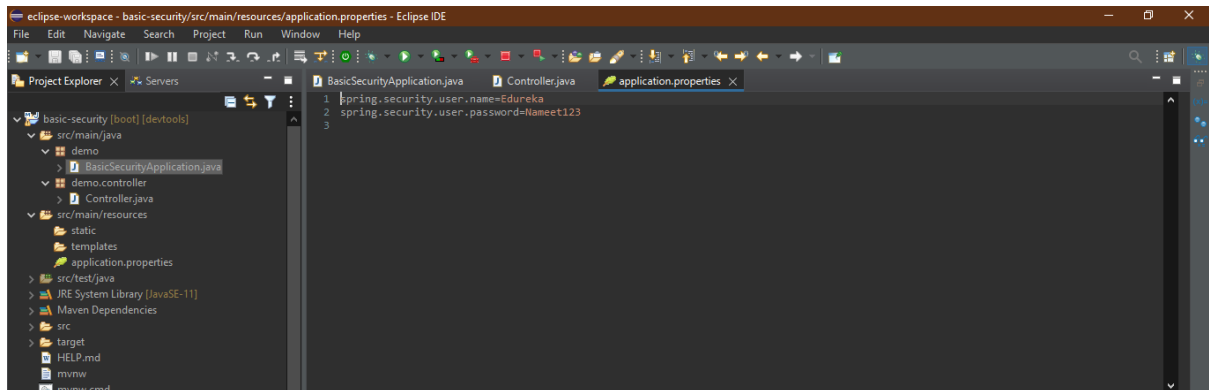


1st Method in Demo 1 is Springboot Default Security

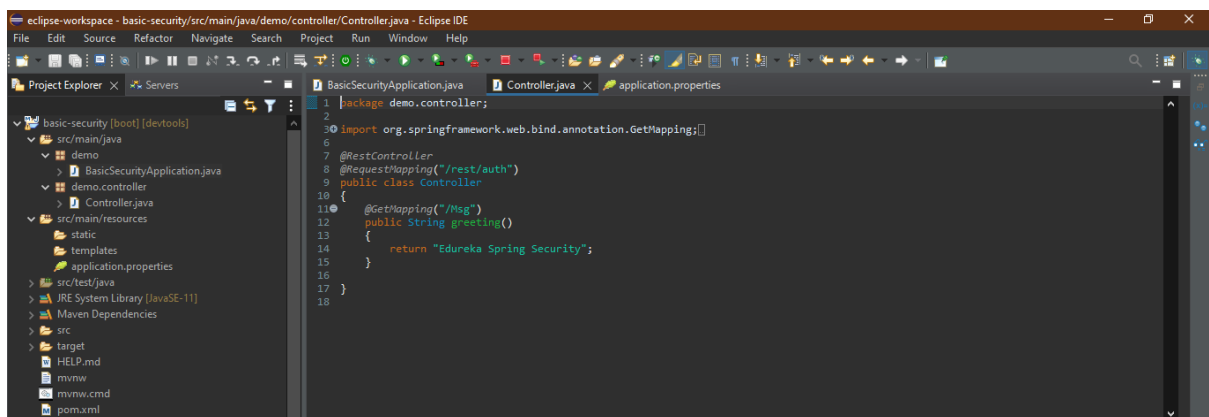
application.properties file settings code



The screenshot shows the Eclipse IDE interface. The Project Explorer on the left displays the project structure for 'basic-security'. The main editor window shows the 'application.properties' file with the following content:

```
1 spring.security.user.name=Edureka
2 spring.security.user.password=Nameet123
3
```

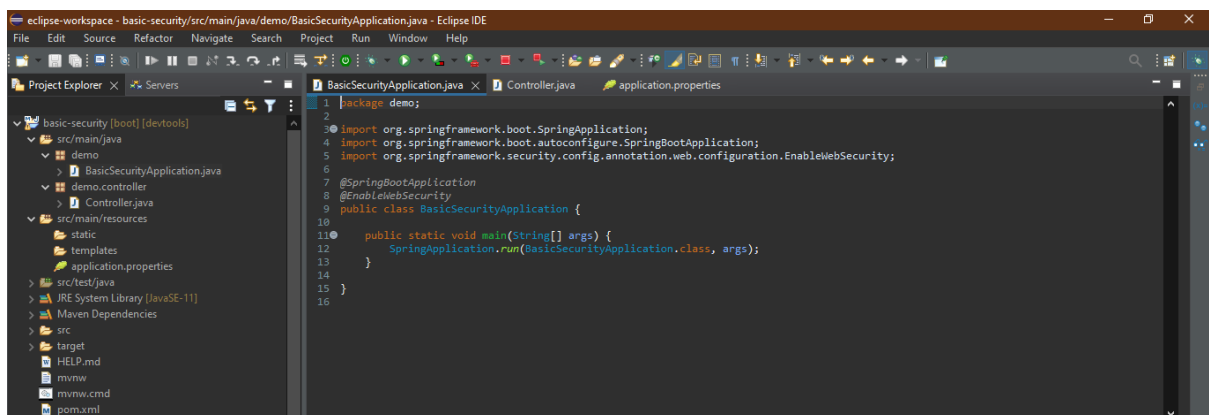
Controller file



The screenshot shows the Eclipse IDE interface. The Project Explorer on the left displays the project structure for 'basic-security'. The main editor window shows the 'Controller.java' file with the following content:

```
1 package demo.controller;
2
3 import org.springframework.web.bind.annotation.GetMapping;
4
5 @RestController
6 @RequestMapping("/rest/auth")
7 public class Controller
8 {
9     @GetMapping("/Msg")
10    public String greeting()
11    {
12        return "Edureka Spring Security";
13    }
14 }
```

Main application file

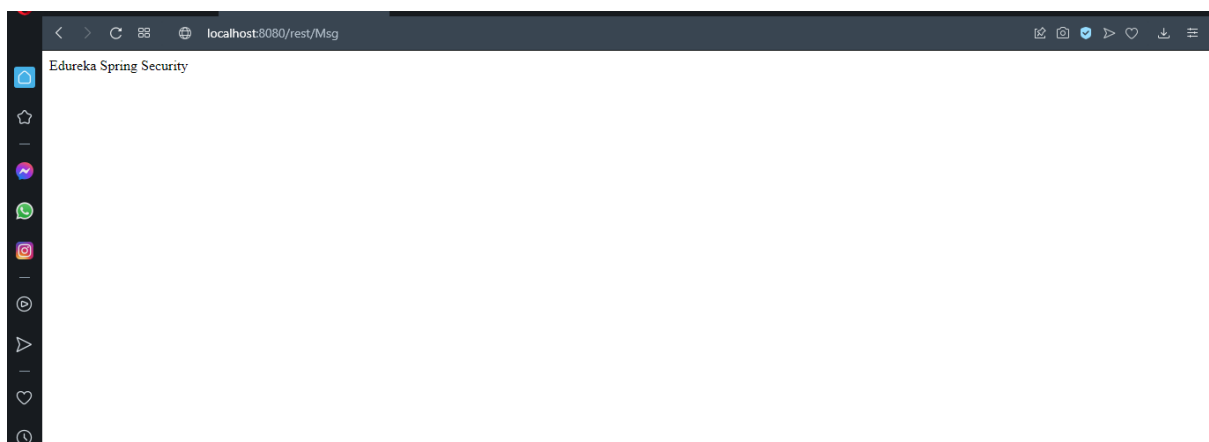
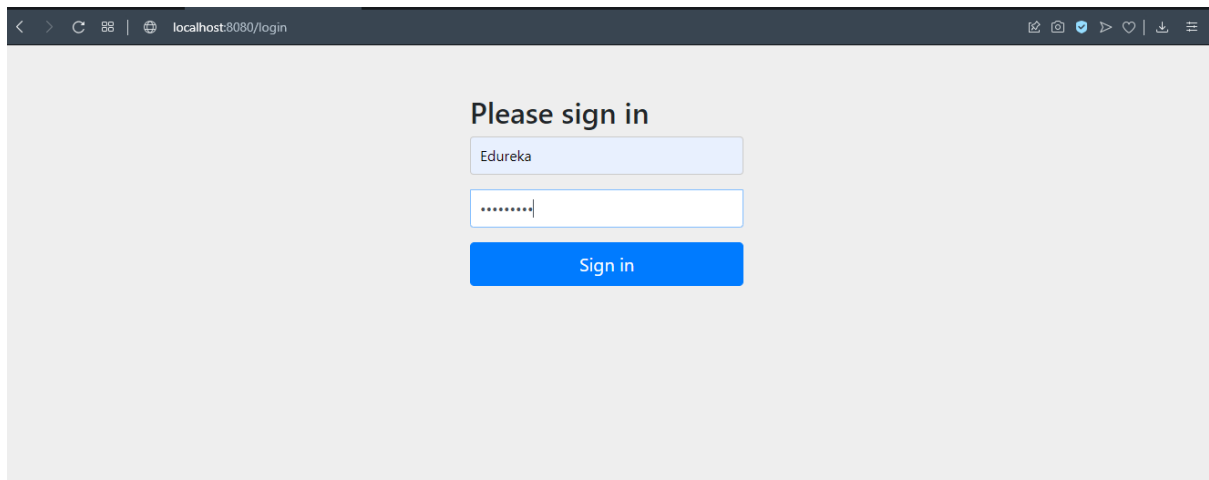


The screenshot shows the Eclipse IDE interface. The Project Explorer on the left displays the project structure for 'basic-security'. The main editor window shows the 'BasicSecurityApplication.java' file with the following content:

```
1 package demo;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.SpringBootApplication;
5 import org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;
6
7 @SpringBootApplication
8 @EnableWebSecurity
9 public class BasicSecurityApplication {
10
11    public static void main(String[] args) {
12        SpringApplication.run(BasicSecurityApplication.class, args);
13    }
14 }
```

Run: localhost:8080/rest/Msg

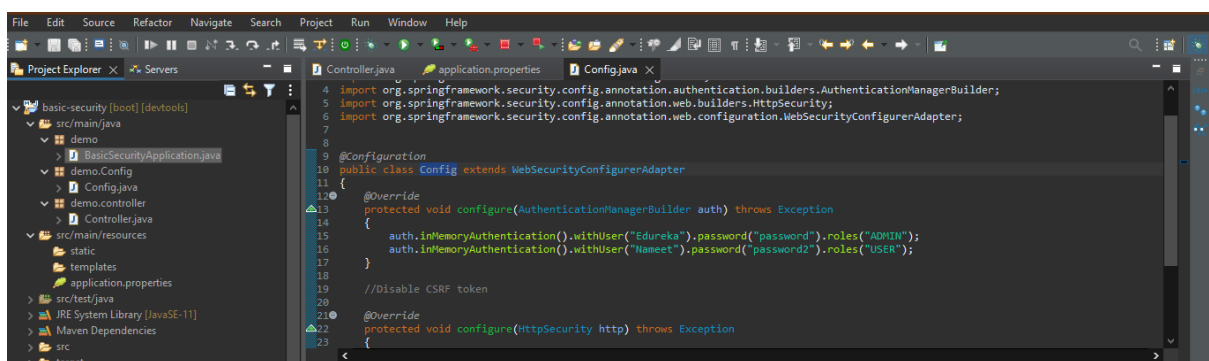
On default browser



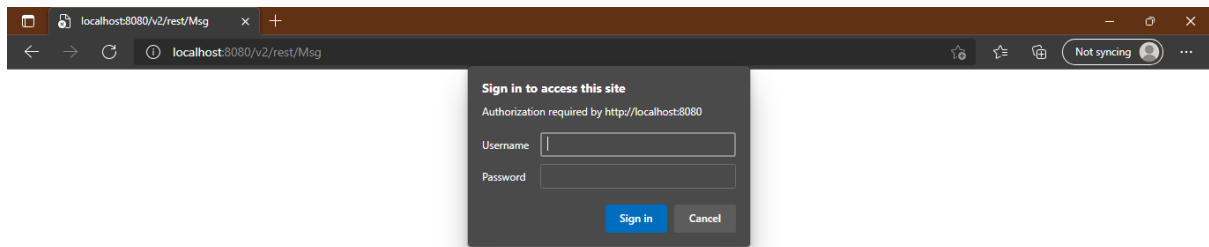
PART 2

Full Authentication:

Program the Config class



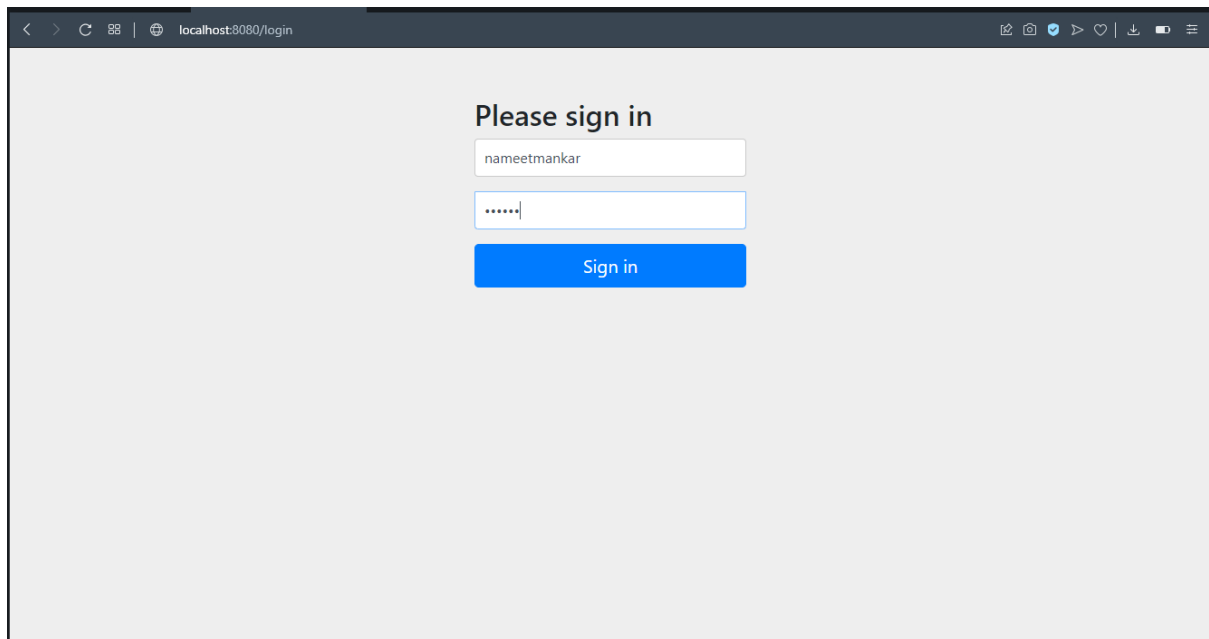
Run : localhost:8080/rest/Msg



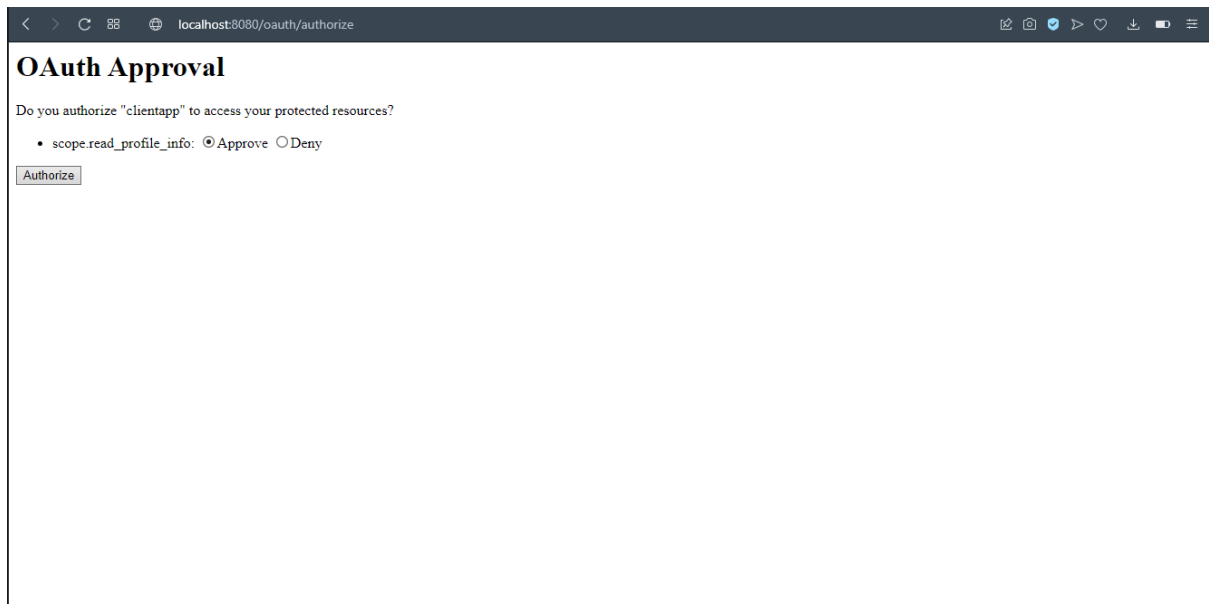
OAUTH 2.0

Getting Authorization Grant Code from User

the first step is to get an authorization grant from the resource owner from URL :
http://localhost:8080/oauth/authorize?client_id=clientapp&response_type=code&scope=read_profile_info



After login, you will be redirected to the grant access page where you choose to give access to third-party applications.



Approve and click Authorize

You will get

<http://localhost:8081/login?code=RMXq39>

RMXq39 is access code for third party application

For operation of postman and access token see class tutorial it gets a bit complicated