

Handling User Authentication .Codes.

Democontroller.java

```
package com.example.demo;
```

```
import org.springframework.stereotype.Controller;
```

```
import org.springframework.web.bind.annotation.RequestMapping;
```

```
import org.springframework.web.bind.annotation.ResponseBody;
```

```
@Controller
```

```
public class DemoController {
```

```
    @ResponseBody
```

```
    @RequestMapping("/")
```

```
    public String all() {
```

```
        return "all simplilearn can access";
```

```
    }
```

```
    @ResponseBody
```

```
@RequestMapping("/admin")  
  
public String admin() {  
    return "hi admins";  
}  
  
@ResponseBody  
  
@RequestMapping("/user")  
  
public String user() {  
    return "hi simplilearn users";  
}  
  
}
```

SecurityConfig.java

```
package com.example.demo;
```

import org.springframework.context.annotation.Bean;

import

org.springframework.security.config.annotation.authentication.builders.AuthenticationManagerBuilder;

import

org.springframework.security.config.annotation.web.builders.HttpSecurity;

import

org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;

import

org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter;

import

org.springframework.security.crypto.password.NoOpPasswordEncoder;

import

org.springframework.security.crypto.password.PasswordEncoder;

@EnableWebSecurity

public class SecurityConfig extends WebSecurityConfigurerAdapter {

//authentication

@Override

```
protected void configure(AuthenticationManagerBuilder auth)  
throws Exception {
```

```
// chain of config
```

```
auth.inMemoryAuthentication()
```

```
.withUser("admin")
```

```
.password("admin")
```

```
.roles("ADMIN")
```

```
.and()
```

```
.withUser("user")
```

```
.password("user")
```

```
.roles("USER");
```

```
}
```

```
@Bean
```

```
public PasswordEncoder getPassword() {
```

```
//dont change the password
```

```
return NoOpPasswordEncoder.getInstance();
```

```
}
```

//authorization

@Override

protected void configure(HttpSecurity http) throws Exception {

http.authorizeRequests()

.antMatchers("/admin").hasRole("ADMIN")

.antMatchers("/user").hasRole("USER")

.antMatchers("/user").hasRole("ADMIN")

.antMatchers("/").permitAll().and().formLogin();

}

}

SpringSecurityApplication.java

package com.example.demo;

```
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SpringSecurityApplication {

    public static void main(String[] args) {
        SpringApplication.run(SpringSecurityApplication.class, args);
    }

}
```

Name :- Vishnu Avtar