**AppSecurityConfig**

package com.example.secureapp;

import java.util.ArrayList;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.beans.factory.annotation.Configurable;

import org.springframework.context.annotation.Bean;

import org.springframework.security.authentication.AuthenticationProvider;

import org.springframework.security.authentication.dao.DaoAuthenticationProvider;

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

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

import org.springframework.security.core.userdetails.User;

import org.springframework.security.core.userdetails.UserDetails;

import org.springframework.security.core.userdetails.UserDetailsService;

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

import org.springframework.security.provisioning.InMemoryUserDetailsManager;

@Configurable

@EnableWebSecurity

public class AppSecurityConfig extends WebSecurityConfigurerAdapter

{

@Autowired

private UserDetailsService userDetailsService;

@Bean

public AuthenticationProvider authProvider()

{

DaoAuthenticationProvider provider=new DaoAuthenticationProvider();

provider.setUserDetailsService(userDetailsService);

provider.setPasswordEncoder(NoOpPasswordEncoder.getInstance());

return provider;

}

// @Bean

// @Override

// protected UserDetailsService userDetailsService() {

//

//

// List<UserDetails> users=new ArrayList<>();

//

// users.add(User.withDefaultPasswordEncoder().username("Ashish").password("1234").roles("USER").build());

//

// return new InMemoryUserDetailsManager(users);

// }

}

**HomeController**

package com.example.secureapp;

import org.springframework.stereotype.Controller;

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

@Controller

public class HomeController {

@RequestMapping("/")

public String home()

{

return "home.jsp";

}

}

**class MyUserDetailsService**

package com.example.secureapp;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.security.core.userdetails.UserDetails;

import org.springframework.security.core.userdetails.UserDetailsService;

import org.springframework.security.core.userdetails.UsernameNotFoundException;

import org.springframework.stereotype.Service;

@Service

public class MyUserDetailsService implements UserDetailsService {

@Autowired

private UserRepository repo;

@Override

public UserDetails loadUserByUsername(String username) throws UsernameNotFoundException {

User user =repo.findByUsername(username);

if(user==null)

throw new UsernameNotFoundException("Error 404");

return new UserPrincipal(user);

}

}

**SecurityInSpringApplication**

package com.example.secureapp;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SecurityInSpringApplication {

public static void main(String[] args) {

SpringApplication.run(SecurityInSpringApplication.class, args);

}

}

**package** com.example.secureapp;

**import** javax.persistence.Entity;

**import** javax.persistence.Id;

@Entity

**public** **class** User {

@Id

**private** **long** id;

**private** String username;

**private** String password;

**public** **long** getId() {

**return** id;

}

**public** **void** setId(**long** id) {

**this**.id = id;

}

**public** String getUsername() {

**return** username;

}

**public** **void** setUsername(String username) {

**this**.username = username;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

}

**package** com.example.secureapp;

**import** java.util.Collection;

**import** java.util.Collections;

**import** org.springframework.security.core.GrantedAuthority;

**import** org.springframework.security.core.authority.SimpleGrantedAuthority;

**import** org.springframework.security.core.userdetails.UserDetails;

**public** **class** UserPrincipal **implements** UserDetails {

**private** User user;

**public** UserPrincipal(User user) {

**super**();

**this**.user = user;

}

@Override

**public** Collection<? **extends** GrantedAuthority> getAuthorities() {

// **TODO** Auto-generated method stub

**return** Collections.*singleton*(**new** SimpleGrantedAuthority("USER"));

}

@Override

**public** String getPassword() {

// **TODO** Auto-generated method stub

**return** user.getPassword();

}

@Override

**public** String getUsername() {

// **TODO** Auto-generated method stub

**return** user.getUsername();

}

@Override

**public** **boolean** isAccountNonExpired() {

// **TODO** Auto-generated method stub

**return** **true**;

}

@Override

**public** **boolean** isAccountNonLocked() {

// **TODO** Auto-generated method stub

**return** **true**;

}

@Override

**public** **boolean** isCredentialsNonExpired() {

// **TODO** Auto-generated method stub

**return** **true**;

}

@Override

**public** **boolean** isEnabled() {

// **TODO** Auto-generated method stub

**return** **true**;

}

}

**package** com.example.secureapp;

**import** org.springframework.data.jpa.repository.JpaRepository;

**public** **interface** UserRepository **extends** JpaRepository<User, Long> {

User findByUsername(String username);

}

**JSp Page**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

Your Home Page Is Unlocked Successfully.....

</body>

</html>