Dalam tutorial ini saya belajar tentang web security pada spring boot. Melakukan setting security pada spring boot ternyata lebih mudah karena sudah tersedia library otomatis sehingga tidak terlalu repot mengaturnya. Yang saya lakukan untuk menyelesaikan latihan hanyalah mengikuti tutorial yang ada yaitu melakukan pengaturan pada file WebSecurityConfig dimana pada file tersebut dilakukan query pada table user untuk melakukan autentikasi. Berikut file WebSecurityConfig.java yang dibuat pada tutorial kali ini.

|  |
| --- |
| package com.example;  import javax.activation.DataSource;  import org.springframework.beans.factory.annotation.Autowired;  import org.springframework.context.annotation.Configuration;  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;  @Configuration  @EnableWebSecurity  public class WebSecurityConfig extends WebSecurityConfigurerAdapter{  @Override  protected void configure(HttpSecurity http) throws Exception{  http.authorizeRequests().  antMatchers("/").permitAll().  antMatchers("/course/\*\*").hasRole("ADMIN").antMatchers("/student/\*\*").hasRole("USER").  anyRequest().authenticated().and().formLogin().  loginPage("/login").permitAll().and().logout().permitAll();  }  // @Autowired  // public void configureGlobal(AuthenticationManagerBuilder auth) throws Exception{  // auth.inMemoryAuthentication().withUser("user").password("password").roles("USER");  // auth.inMemoryAuthentication().withUser("admin").password("admin").roles("ADMIN");  // }  @Autowired  javax.sql.DataSource dataSource;  @Autowired  public void configAuthentication(AuthenticationManagerBuilder auth) throws Exception{  auth.jdbcAuthentication().dataSource(dataSource).usersByUsernameQuery("select username,password,enabled from users where username =? "). authoritiesByUsernameQuery( "select username, role from user\_roles where username =? ");  }  } |