**Spring Security 5.2**

* Metoda configure, dar si altele, au deja inca un prototip, fiind supraincarcate, de ex pentru:

protected void configure(HttpSecurity http) throws Exception {

* Aceasta metoda e facuta sa suporte LambdaExpression drept parametru local si deci logic si actual
* Metoda era configurata asa mai inainte:

@Override  
 protected void configure(HttpSecurity http) throws Exception {  
 http.authorizeHttpRequests().requestMatchers("/resources/\*\*").permitAll()  
 .and()  
 .authorizeRequests().requestMatchers("/managers").hasRole("MANAGER")  
 .and()  
 .authorizeRequests().requestMatchers("/admins","/managers").hasRole("ADMIN")  
 .and()  
 .authorizeRequests().anyRequest().authenticated()  
 .and()  
 .logout().permitAll()  
 .and()  
 .formLogin().loginPage("/login").loginProcessingUrl("/checkLogin").permitAll()  
 .and()  
 .exceptionHandling().accessDeniedPage("/noPermission");  
 }  
}

Acum, putem face asa mai usor:

@Override  
protected void configure(HttpSecurity http) throws Exception {  
 http.authorizeHttpRequests(authorize -> authorize.requestMatchers("/resources/\*\*").permitAll())  
 .authorizeRequests(authorize -> authorize.requestMatchers("/managers").hasRole("MANAGER"))  
 .authorizeRequests(authorize -> authorize.requestMatchers("/admins","/managers").hasRole("ADMIN"))  
 .authorizeRequests(authorize -> authorize.anyRequest().authenticated())  
 .logout().permitAll()  
 .and()  
 .formLogin(form -> form.loginPage("/login").loginProcessingUrl("/checkLogin").permitAll())  
 .exceptionHandling().accessDeniedPage("/noPermission");  
}

* Deci, vedem ca metoda authorizeRequests din http object are deja un nou prototip care suporta ca parametru formal lambda expression. Aceasta lambda expression, in clasa unde se afla metoda authorizeRequests, va fi executata si drept param cu numele authorize i se va trimite this, adica propriu zis obiectul.
* Beneficiul e ca scurteaza codul si nu mai trebuie mereu sa folosim .and() pentru a returna obiectul http. authorizeRequests() si asa il returneaza, dar deoarece toate operatiile au loc acum in authorizeRequests(), doar ca cu o lamba expression, metoda returneaza obiectul si atat. Toate metodele principale, ca si fromLogin() etc. au noi metoda supraincarcate ce suporta un param de tip lamba expression.