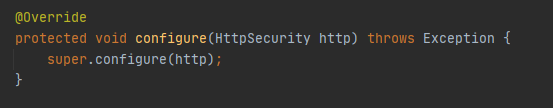
Configurando projeto em Spring Boot com SpringSecurity

* SpringSecurity

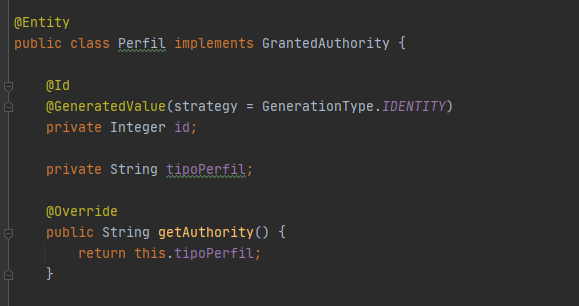
1. Criação de pacote “security” dentro do pacote de “configuracoes”
2. Criar classe “SecurityConfigurations” **extends WebSecurityConfigurerAdapter**
3. Sobrescrever métodos da classe abstrata, basta digitar configure e esperar a IDE completar ou ALT INSERT caso não funcione...

Sobrescrita do método:



****

1. Neste trecho de código aplicamos métodos que vão definir quais requisições são liberadas em nosso projeto, requisições do tipo GET referente a cadastros são permitidas para qualquer tipo de usuário, qualquer outro tipo de requisição (DELETE, POST, PUT...) são desabilitadas para quaisquer usuários, retornando erro de código 401 Unauthorized.
2. O SpringSecurity necessita de uma classe de perfil, ela vai definir os “perfis” de acesso da sua aplicação, como administrador, “somente leitura” entre outros.
3. Para isso vamos criar a entidade perfil implementando “**GrantedAuthority**”, implementando seu método para retornar seu tipo de perfil



1. Para aplicar os perfis criados será necessário criar uma entidade “**Usuario**” que vai implementar “**UserDetails**” que poderá possuir o que um usuário de login necessita, como senha, nome, e-mail etc.
2. Possuindo também um atributo List<Perfil> com a notação @ManyToMan.
3. Ao sobrescrever seus métodos, deve-se configurar os mesmos de acordo com sua aplicação.



1. Métodos:
   1. getAuthorities(): retorna lista de perfis
   2. getPassword(): retorna atributo senha
   3. getUsername(): retorna atributo de login (email)
   4. Restante dos métodos varia de acordo com a regra criada

Autenticações:

Para efetuar autenticações na aplicação, vai ser necessário