Building ToDo app Spring Security

Adding security to application

When we access http://localhost:8080/list-todos we are able to get todo list this should not happen. Our application should not be accessed by unknown users.
When somebody try to access any data it should redirect to login page.
The framework which allows us to add this functionality is spring security.
Now removing login related functionality -> Delete login.jsp and make changes in LoginController

```
@Controller
@SessionAttributes("name")
public class LoginController {

	@Autowired
	private LoginService loginService;

	@RequestMapping(value = "/", method = RequestMethod.GET)
	public String loginMessage(ModelMap modelMap) {
		modelMap.put("name", "Nagaraj S Kharvi");
		return "welcome";
	}
}
```

Remove login URI link from navigation jsp page.

navigation.jspf

```
<nav role="navigation" class="navbar navbar-default">
	<div class="">
		<a href="http://www.in28minutes.com" class="navbar-
brand">Todo App</a>
	</div>
	<div class="navbar-collapse">
		<ul class="nav navbar-nav">
			<li class="active"><a href="/">Home</a></li>
			<li><a href="/list-todos">Todos</a></li>
		</ul>
	</div>
</nav>
```

Try application now.

To add spring security feature you need to add one dependency. First thing it does is it adds required jars and auto configures the application.

```
<dependency>
	<groupId>org.springframework.boot</groupId>
	<artifactId>spring-boot-starter-security</artifactId>
```

```
</dependency>
```

Now when you access the application it asks you user id and password. User id will be user and when you run the application it displays password on console.

```
Root WebApplicationContext: initialization completed in 1448 ms
2021-01-08 21:53:44.980  INFO 4482 --- [  restartedMain]
o.s.s.concurrent.ThreadPoolTaskExecutor  : Initializing
ExecutorService 'applicationTaskExecutor'
2021-01-08 21:53:44.987 DEBUG 4482 --- [  restartedMain]
s.w.s.m.m.a.RequestMappingHandlerAdapter : ControllerAdvice beans: 0
@ModelAttribute, 0 @InitBinder, 1 RequestBodyAdvice, 1
ResponseBodyAdvice
2021-01-08 21:53:45.068 DEBUG 4482 --- [  restartedMain]
s.w.s.m.m.a.RequestMappingHandlerMapping : 9 mappings in
'requestMappingHandlerMapping'
2021-01-08 21:53:45.103 DEBUG 4482 --- [  restartedMain]
o.s.w.s.handler.SimpleUrlHandlerMapping  : Patterns [/webjars/**, /
**] in 'resourceHandlerMapping'
2021-01-08 21:53:45.134 DEBUG 4482 ---
[  restartedMain] .m.m.a.ExceptionHandlerExceptionResolver :
ControllerAdvice beans: 0 @ExceptionHandler, 1 ResponseBodyAdvice
2021-01-08 21:53:45.246  INFO 4482 ---
[  restartedMain] .s.s.UserDetailsServiceAutoConfiguration :

Using generated security password: 2336394e-5a34-48e7-9378-
a6882c2bd9b5

2021-01-08 21:53:45.379  INFO 4482 --- [  restartedMain]
o.s.s.web.DefaultSecurityFilterChain     : Will secure any request
with
[org.springframework.security.web.context.request.async.WebAsyncMana
gerIntegrationFilter@73c05bc2,
org.springframework.security.web.context.SecurityContextPersistenceF
ilter@611ddb6a,
org.springframework.security.web.header.HeaderWriterFilter@b101c06,
org.springframework.security.web.csrf.CsrfFilter@2613caf5,
org.springframework.security.web.authentication.logout.LogoutFilter@
b1fd156,
org.springframework.security.web.authentication.UsernamePasswordAuth
enticationFilter@302b4967,
org.springframework.security.web.authentication.ui.DefaultLoginPageG
eneratingFilter@42be1107,
org.springframework.security.web.authentication.ui.DefaultLogoutPage
GeneratingFilter@7445330,
org.springframework.security.web.authentication.www.BasicAuthenticat
ionFilter@7e806724,
org.springframework.security.web.savedrequest.RequestCacheAwareFilte
r@4a43d639,
org.springframework.security.web.servletapi.SecurityContextHolderAwa
reRequestFilter@6a82a710,
org.springframework.security.web.authentication.AnonymousAuthenticat
```

```
ionFilter@7a00bdbb,
org.springframework.security.web.session.SessionManagementFilter@458
25e2f,
org.springframework.security.web.access.ExceptionTranslationFilter@5
c26d882,
org.springframework.security.web.access.intercept.FilterSecurityInte
rceptor@5d0a7da4]
2021-01-08 21:53:45.468  INFO 4482 --- [  restartedMain]
o.s.b.d.a.OptionalLiveReloadServer       : LiveReload server is
running on port 35729
2021-01-08 21:53:45.514  INFO 4482 --- [  restartedMain]
o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on
port(s): 8080 (http) with context path ''
2021-01-08 21:53:45.547  INFO 4482 --- [  restartedMain]
com.example.demo.Application             : Started Application in
2.659 seconds (JVM running for 8.563)
```

Login with user id user and password shown in the console. Typically we have database to store user id and password.
So setting a security configuration now. Using a hardcoded values and for password Password encoder should be mapped.

```
@Configuration
public class SecurityConfiguration extends
WebSecurityConfigurerAdapter {

     @Autowired
     public void
configureGlobalSecurity(AuthenticationManagerBuilder auth) throws
Exception {
          auth.inMemoryAuthentication().withUser("Nagaraj S
Kharvi").password("{noop}1234")
               .roles("USER", "ADMIN");
     }
}
```

Instead of pop-up we can provide usual login page and we can restrict users to view specific page

```
@Configuration
public class SecurityConfiguration extends
WebSecurityConfigurerAdapter {

     @Autowired
     public void
configureGlobalSecurity(AuthenticationManagerBuilder auth) throws
Exception {
          auth.inMemoryAuthentication().withUser("Nagaraj S
Kharvi").password("{noop}1234")
               .roles("USER", "ADMIN");
     }
```

```
     @Override
     protected void configure(HttpSecurity http) throws Exception {
          http.authorizeRequests().antMatchers("/
login").permitAll()
                .antMatchers("/", "/*todo*/
**").access("hasRole('USER')").and()
                .formLogin();
     }
}
```