

Spring Professional Exam Tutorial v5.0

Question 10

Question 10 - What @PreAuthorized and @RolesAllowed annotations do? What is the difference between them?

`@PreAuthorized` and `@RolesAllowed` annotations are part of Method Security Model implemented in Spring Security. Both of those annotations are evaluated before method is executed, to check if user is allowed to execute desired method.

`@RolesAllowed` annotation, is very similar to `@Secured` annotation, and both of those allows you to specify list of roles that currently authenticated user needs to have assigned to be allowed to execute guarded method. `@RolesAllowed` annotation is part of JSR 250 standard. Annotation is enabled by:

```
@EnableGlobalMethodSecurity(jsr250Enabled = true)
```

`@PreAuthorized` annotation allows you to specify, conditions under which user is allowed to execute method, with usage of SpEL expressions. Expression is evaluated before method is executed. Annotation support is enabled by:

```
@EnableGlobalMethodSecurity(prePostEnabled = true)
```

Difference between those annotation is that `@RolesAllowed` allows you to specify list of required roles, and `@PreAuthorized` annotation allows you to specify security rule with usage of SpEL expression.

Question 10 - What @PreAuthorized and @RolesAllowed annotations do? What is the difference between them?

SpEL expressions that can be used with @PreAuthorized annotation allow for many different conditions and logic to be expressed, commonly used functions are:

- ▶ hasRole
- ▶ hasAnyRole
- ▶ hasAuthority
- ▶ hasAnyAuthority
- ▶ isAnonymous
- ▶ isAuthenticated
- ▶ isRememberMe

Example usage:

```
@PreAuthorize("hasRole('ROLE_EMPLOYEES_CREATE') && isAuthenticated() && !isAnonymous()")
```

