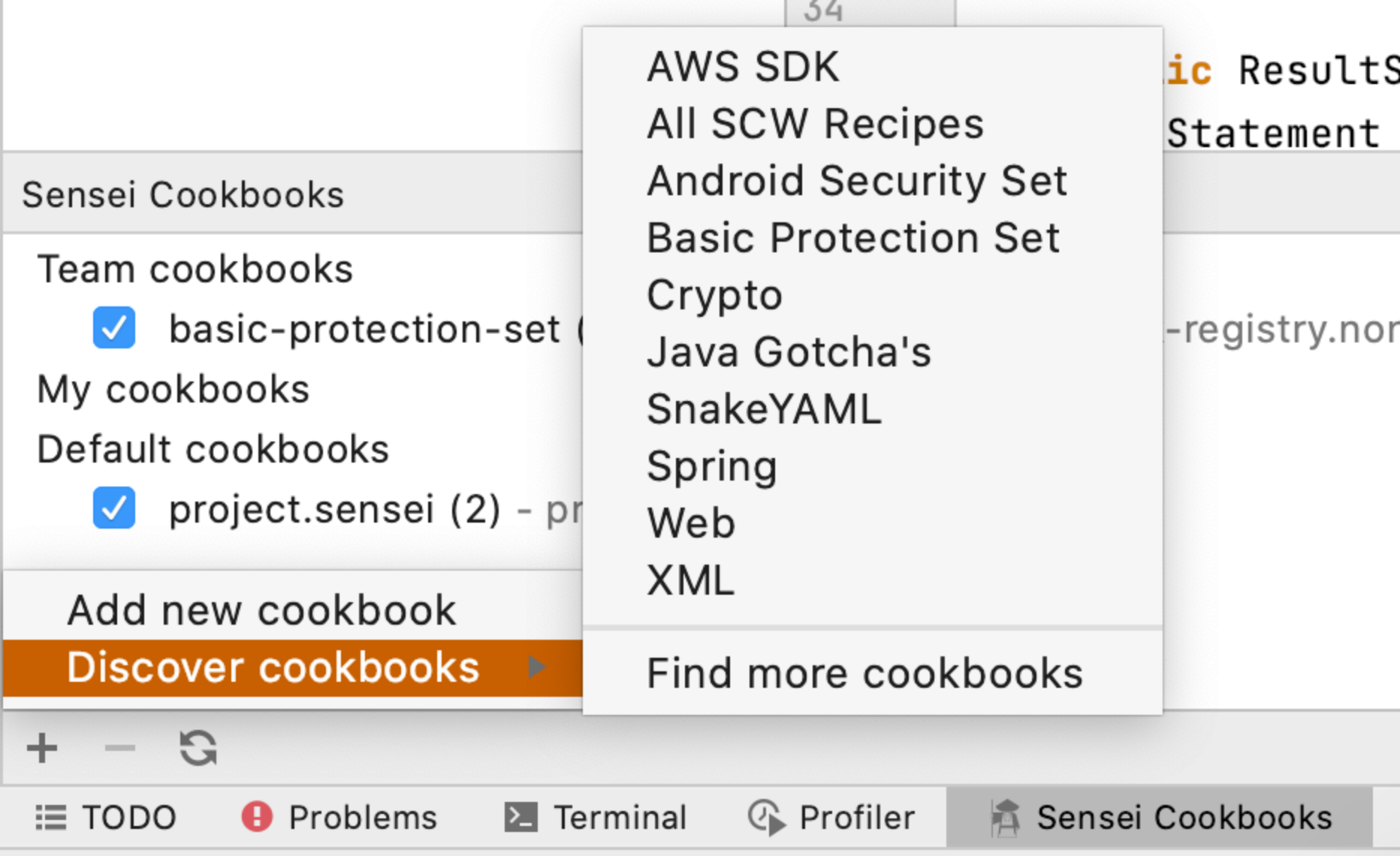
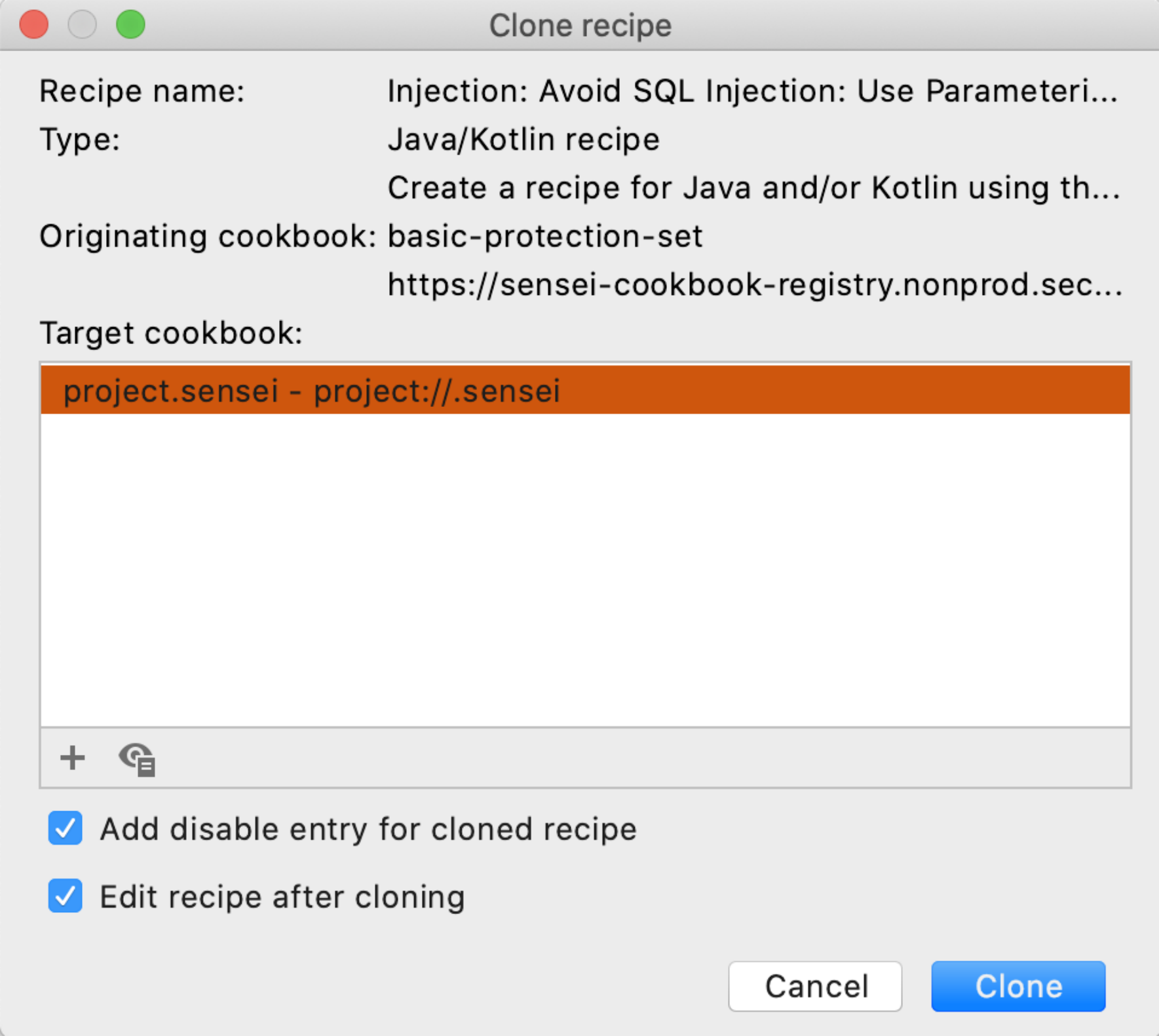
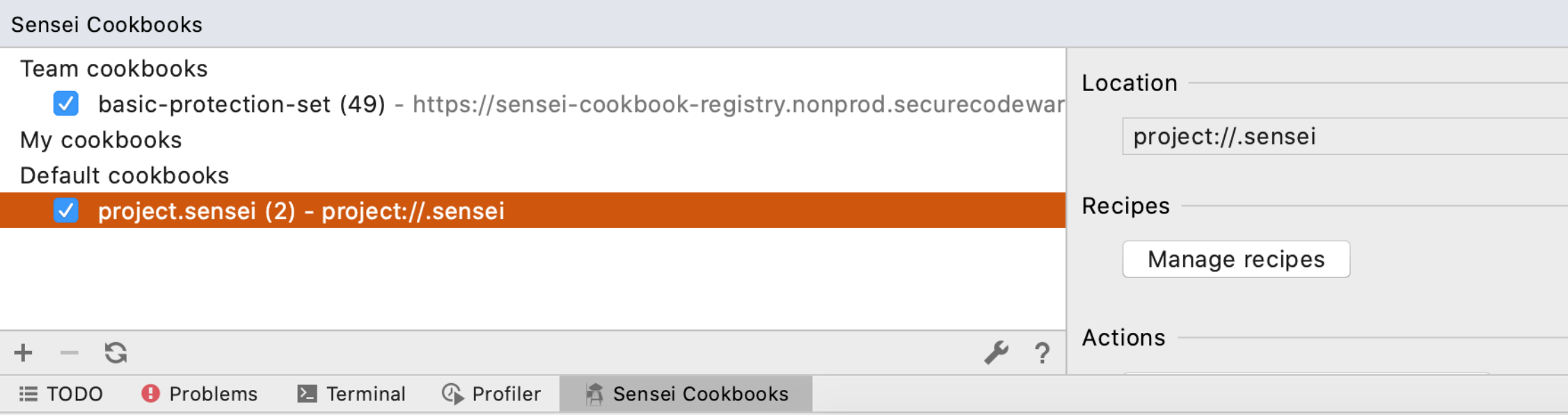
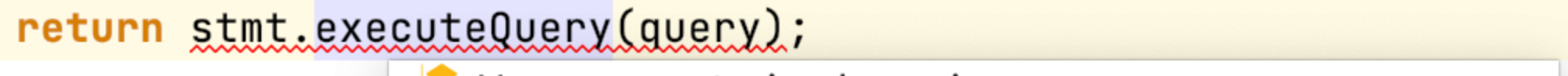

34
ic ResultS
Statement
Sensei Cookbooks
Team cookbooks
basic-protection-set (
-registry.nor
My cookbooks
Default cookbooks
project.sensei (2) - pr
Add new cookbook
Discover cookbooks
AWS SDK
All SCW Recipes
Android Security Set
Basic Protection Set
Crypto
Java Gotcha's
SnakeYAML
Spring
Web
XML
Find more cookbooks
TODO
Problems
Terminal
Profiler
Sensei Cookbooks


Clone recipe
Recipe name:
Injection: Avoid SQL Injection: Use Parameteri...
Type:
Java/Kotlin recipe
Create a recipe for Java and/or Kotlin using th...
Originating cookbook:
basic-protection-set
https://sensei-cookbook-registry.nonprod.sec...
Target cookbook:
project.sensei - project://.sensei
Add disable entry for cloned recipe
Edit recipe after cloning
Cancel
Clone


Sensei Cookbooks
Team cookbooks
basic-protection-set (49) - https://sensei-cookbook-registry.nonprod.securecodewar
My cookbooks
Default cookbooks
project.sensei (2) - project://.sensei
Location
project://.sensei
Recipes
Manage recipes
Actions
TODO
Problems
Terminal
Profiler
Sensei Cookbooks


return stmt.executeQuery(query);
Use parameterized queries
Copy recipe 'Injection: Avoid SQL Injection: Use...'

Sensei by Secure Code Warrior® - Edit Recipe

General Settings | Recipe Settings | QuickFix Settings

Code view | UI view | both — Supported in: java

```
availableFixes:
- name: "Replace with safe alternative"
  actions:
  - rewrite:
      to: "DefaultHTTPUtilities.getInstance().addCookie({{{ arguments.0 }}})"
```

availableFixes:

name: Replace with safe alternative

actions:

rewrite:

to: `DefaultHTTPUtilities.getInstance().addCookie({{{ arguments.0 }}}`

...

Hide variables

Variable	preview
argumentList	(myCookie)
expressionElement	response.addCoo...
qualifier	response
arguments	myCookie
0	myCookie
methodName	addCookie
type	void
containingClass	public class Test ...

QuickFix to preview: Replace with safe alternative

Side-by-side viewer | Do not ignore | Highlight words | 1 difference

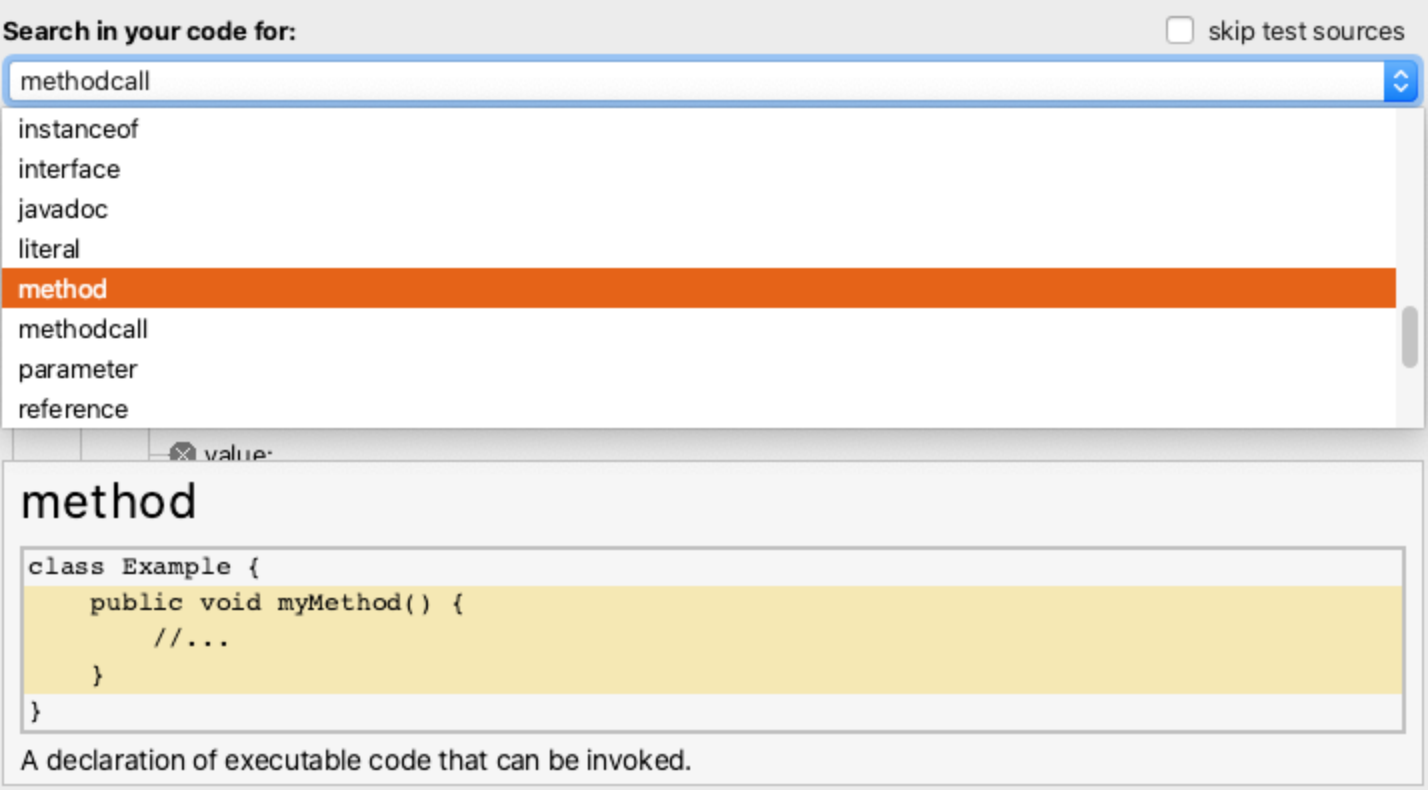
Before

```
public void safeAddCookie(Cookie myCookie, HttpServletResponse response){
    myCookie.setSecure(true);
    myCookie.setHttpOnly(true);
    myCookie.setDomain("sub.domain.scw.com");
    myCookie.setPath("more/narrow/path");
    response.addCookie(myCookie);
}
```

After

```
public void safeAddCookie(Cookie myCookie, HttpServletResponse response){
    myCookie.setSecure(true);
    myCookie.setHttpOnly(true);
    myCookie.setDomain("sub.domain.scw.com");
    myCookie.setPath("more/narrow/path");
    DefaultHTTPUtilities.getInstance().addCookie(myCookie);
}
```

Cancel | Save


Search in your code for:
skip test sources
methodcall
instanceof
interface
javadoc
literal
method
methodcall
parameter
reference
value:
method
class Example {
public void myMethod() {
//...
}
}
A declaration of executable code that can be invoked.

```
return stmt.executeQuery(query);
```

Could lead to SQL Injection read more

Inspection info:Secure coding practices prescribe that queries need to be parameterized. Concatenation of parameters is not recommended.

To create a parameterized query, replace all variables with ? placeholders. Then create a Prepared Statement and use methods from this class to bind any parameter variables.

Before

```
Statement stmt = conn.createStatement();
String query = "SELECT * FROM user WHERE username = "+ username;
return stmt.executeQuery(query)
```

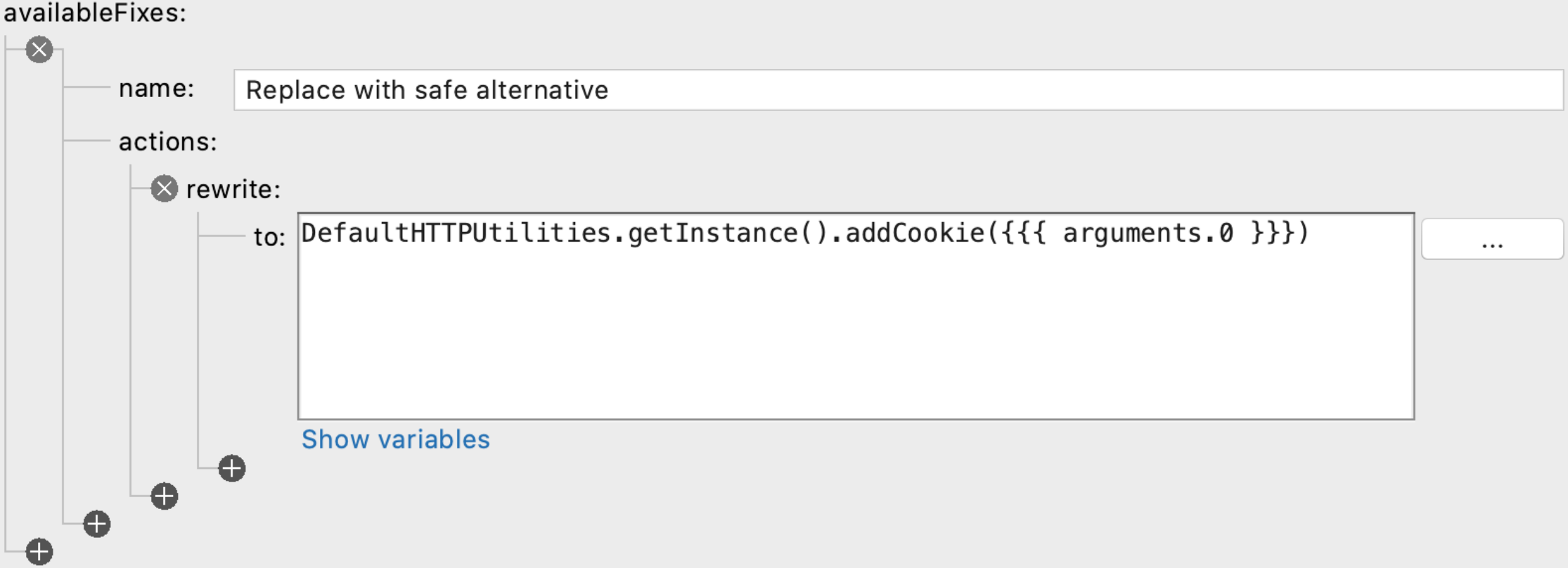
After

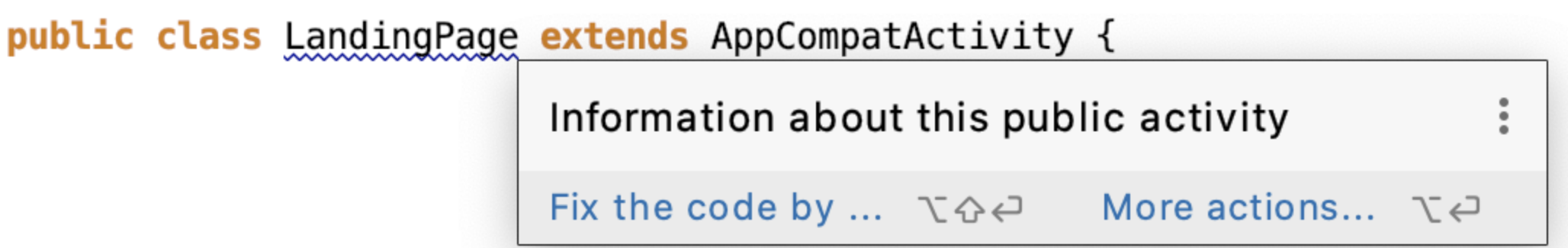
```
PreparedStatement stmt = conn.prepareStatement(query);
String query = "SELECT * FROM user WHERE username = ?";
stmt.setString(1, username);
return stmt.executeQuery()
```

Violating this guideline can cause

- Injection Flaws - SQL Injection. Learn more

Use parameterized queries ⌥⇧⏎ More actions... ⌥⏎


availableFixes:
name:
Replace with safe alternative
actions:
rewrite:
to:
DefaultHTTPUtilities.getInstance().addCookie({{{ arguments.0 }}})
...
Show variables


public class LandingPage extends AppCompatActivity {
Information about this public activity
Fix the code by ... ⌥⇧⏎
More actions... ⌥⏎

```
public class LandingPage extends AppCompatActivity {
```