Preferences

Plugins

Marketplace | Installed | ⚙

🔍 Secur | × ⋮

Manage Plugin Repositories...
HTTP Proxy Settings...

Install Plugin from Disk...

Disable All Downloaded Plugins
Enable All Downloaded Plugins

Search Results (2)

Sensei by Secure Code Warrior® ☑
2021.2.2 SecureCodeWarrior

Sense

SecureCod

Enabled

Plugins

- Appearance & Behavior
- Keymap
- Editor
- **Plugins**
- Version Control
- Build, Execution, Deployment
- Languages & Frameworks
- Tools
- Other Settings

Q Secur

Search Results (57)   Sort By: Relevance ▾

Python Security
⤓ 57,2K   Install

RIPS Security Analysis
⤓ 9,8K   Install

Sensei by Secure Code Warrior®
⤓ 14,2K   Installed

Node Security
⤓ 15,5K   Install

Swagger/OpenAPI Security Assessment
⤓ 2,4K   Install

Reshift Security
⤓ 2K   Install

Android SecuRefactor
⤓ 414   Install

CheckMate Security Analysis
⤓ 440   Install

Snyk Vulnerability Scanner
⤓ 58,3K   Install

Sensei by Secure Code Warrior®   Installed

⤓ 14,2K   Secure Code Warrior®

Code tools   2021.2.2   Jul 23, 2021

Plugin homepage ↗

Sensei IntelliJ plugin helps you find and fix bad coding patterns dynamically by automatically highlighting violations as you write code and suggesting fixes that can be applied with a mouse click.

Wipe out recurring anti-patterns in your code. Create Sensei recipes which detect the anti-patterns and can automatically rewrite the code using your preferred coding pattern. Making it easy to proactively improve your code quality.

The recipes you create can be stored in your project and added to version control, making it easy to share recipes with your team. You can gradually build team-specific coding guidelines.

Repeated use in the IDE helps build muscle memory so you stop making the same mistakes and move from detecting and fixing issues, to actively preventing them from entering your code in the first place.

Additional Benefits include:

- Dynamically spot low-quality code patterns as you write code
- Automate code changes to implement your coding patterns
- Codify your coding standards
- No waiting for a static analysis tool report, analyze your code automatically on the fly
- Share your coding knowledge among teams
- Create recipes to recommend best practices
- Increase the consistency of your coding approach
- Link to any training material accessible online from the code in the IDE itself
- Link back to Secure Code Warrior learning platform material using categories
- Works with Java
- Create recipes for instant feedback against common potential bugs
- Mitigate security vulnerabilities by creating rules that detect vulnerable code and provide an automated fix to a secure implementation

Cancel   Apply   OK

```xml
<activity
    android:name="PublicActivity"
    android:theme="@style/AppTheme.NoActionBar"
    android:exported="true" >
</activity>
```

```java
public ResultSet findUser(Connection conn, String username) throws SQLException {
    Statement stmt = conn.createStatement();
    String query = "SELECT * FROM user WHERE username = "+ username;
    return stmt.executeQuery(query);
}
```

🪓 Use parameterized queries	▶
🪓 Copy recipe 'Injection: Avoid SQL Injection: Use...'	▶
🪓 Take the training provided by Secure Code Warrior	▶

```java
public ResultSet findUser(Connection conn, String username) throws SQLException {
    Statement stmt = conn.createStatement();
    String query = "SELECT * FROM user WHERE username = "+ username;
    return stmt.executeQuery(query);
}
```

🔨 **Use parameterized queries** ▶   Injection: Avoid SQL Injection: Use Parameterized Queries (Statement)

R↗ Introduce local variable ▶   🔧 Edit inspection profile setting

Create a new recipe

Add method contract to 'executeQuery' ▶   Fix all 'Injection: Avoid SQL Injection: Use Parameterized Queries (Statement)' problems in file

Press ⌥Space to open preview   Run inspection on ...

✕ Disable inspection

Suppress all inspections for class

Suppress for class

Suppress for method

Suppress for statement

Sensei by Secure Code Warrior®

Sensei uses recipes to analyze your code and help you in real time.

Results from Sensei are shown directly in the editor.

When a recipe is triggered, you can use ⌥↵ to get access to relevant actions.

Recipes can have quick fixes that understand your code and can intelligently transform it.

💡 Access Sensei actions by using Tools | Sensei in your IDE's menu bar.

Close   Next (1/2)

Replace methodcall

Package name: regex: `javax\.servlet\.http`

Class name: regex: `HttpServletResponse`

Method name: regex: `addCookie`

Check arguments: ☐

Arguments: [+]

QuickFixes

QuickFix Settings

Solution description: `Replace with safe alternative`

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

General Settings | Recipe Settings | QuickFix Settings   ⑦

Name   OWASP ESAPI Banned method: addCookie

Short description   Avoid using OWASP ESAPI banned methods

Level   Error   ⬍

Only mark on new code   ☐

Enable   ☑

─ additional settings

Category   Cross-Site Scripting (XSS)   ⬍

                         Other   ⬍

CWE Category

ID   04df5d6b-4ffc-4cba-a1a9-c8def7689e8d

                         Changing the ID of a recipe may invalidate existing override configurations

Overriding recipe

                         ID of the recipe that this recipe overrides.
                         The overridden recipe will not run if this recipes's scope matches.

Disabled recipes   +

                         IDs of recipes that should be disabled when this recipe is enabled

Coding policy   banned001

Description file   change   banned001.html

Comment   This field can be used to write down some additional information.
                         Cookies are the most common target of XSS attacks, making them httpOnly mitigates these attacks.
                         Such as relevant references:
                         https://javadoc.io/doc/org.owasp.esapi/esapi/2.1.0/org/owasp/esapi/reference/DefaultHTTPUtilities
                         https://owasp.org/www-project-enterprise-security-api/

Tags

                         Suggested tags: no suggestions available
                         Use semicolons to separate tags

Scope   +

                         Scope constraints for this recipe. All constraints should be satisfied for the recipe to be applied.

                                                                              Close   Save

Search in your code for:

skip test sources

```
methodcall
```

⊗ type: `java.lang.Runtime`   ...

Show variables

⊗ name: `exec`   ...

⊗ args:

⊗ 1:

⊗ type: `java.lang.String`   ...

Show variables

⊗ value:

⊕

⊕

⊕

⊕

Add scopes to this

| stringified |
| integer |
| is |
| **containsUntrustedInput** |
| trustNonTextualsByDefault |
| trustedSources |
| |
| anyOf |
| allOf |
| with |
| without |
| not |
| in |

General Settings   **Recipe Settings**   QuickFix Settings

Test recipe   [Hide preview >>]

○ Code view   ○ UI view   ● both   Supported in: java

```
search:
  methodcall:
    name: "addCookie"
    type: "javax.servlet.http.HttpServletResponse"
    args:
      1:
        type: "javax.servlet.http.Cookie"
```

Search in your code for: ☐ skip test sources

[methodcall   ▼]

  ⊗ type: [avax.servlet.http.HttpServletResponse] [...]
            Show variables
  ⊗ name: [addCookie   ] [...]
  ⊗ args:
      ⊗ 1:
          ⊗ type: [javax.servlet.http.Cookie] [...]
                  Show variables
          ⊕
      ⊕
  ⊕

```java
public void executeCommand( String command) thro
    Runtime r = Runtime.getRuntime();
    r.exec(command);
}


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

[Cancel]   [Save]

```
myCookie.setDomain("sub.domain.scw.com");
myCookie.setPath("more/narrow/path");
response.addCookie(myCookie);
```

search for similar methodcalls
start from scratch

Preferences

Account

Sensei is not authenticated. Please log in to an existing account or create a free account to authenticate.

[Authenticate in Browser]

Show recipe writing helpers

- [x] "Create a new recipe" intention
- [x] "Edit recipe" quickfix
- [x] "Clone recipe" quickfix

Other settings

- [ ] Ignore "only mark on new code" recipe setting when VCS is unavailable
- [x] Enable Sensei Recipe Diagnostic Tool to help identifying reasons why recipes don't trigger

Data collection

- [ ] Send usage data

Help Secure Code Warrior improve its product by sending anonymous data about your usage of Sensei. This includes information relating to features used, number of cookbooks and recipes, and similar.
Please note that this will not include any person data or sensitive information, such as source code, file names, etc. Please see our help pages for more information, and our privacy policy.
Data privacy statement

Sidebar:

- Editor
- Plugins ... 2
- Version Control
- Build, Execution, Deployment
- Languages & Frameworks
- Tools
  - Web Browsers
  - External Tools
  - Terminal
  - Database
  - SSH Configurations
  - SSH Terminal
  - Android Device File Explorer
  - Diagrams
  - Diff & Merge
  - Features Trainer
  - Remote SSH External Tools
  - Sensei by Secure Code Warrior®
  - Server Certificates
  - Settings Repository
  - Shared Indexes
  - Startup Tasks
  - Tasks
  - Web Services
  - XPath Viewer
- Other Settings

[Cancel] [Apply] [OK]

```java
public ResultSet findUser(Connection conn, String username) throws SQLException {
    Statement stmt = conn.createStatement();
    String query = "SELECT * FROM user WHERE username = "+ username;
    return stmt.executeQuery(query);
}
```

Could lead to SQL Injection read more

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

```java
public ResultSet findUser(Connection conn, String username) throws SQLException {
    Statement stmt = conn.createStatement();
    String query = "SELECT * FROM user WHERE username = "+ username;
    return stmt.executeQuery(query);
}
```

Violates a guideline on data retrieval read more

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

General Settings	Recipe Settings	QuickFix Settings

○ Code view ● UI view ○ both

availableFixes:

name: Fix the code by ...
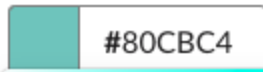
actions:

rewrite:

to:

Show variables

Colors

Custom theme

Column BG

#263238

Active Item

#80CBC4

Active Item Text

#FFFFFF

Hover Item

#13191C

ctive Presence

#50FA7B

Mention Badge

#FF5555

op Nav Text

#FFFFFF

Copy and paste these val   eme with others

#263238,#2E3A40,#80CBC4,#FFFFFF,#13191C,#FFFFFF,#

Copy