FORTRA

beSOURCE
5.2
Jenkins Plugin User Guide

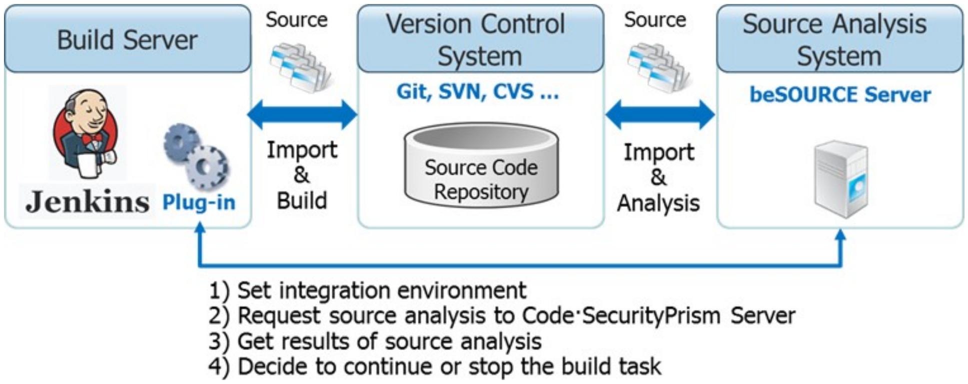
Copyright Terms and Conditions

Table of Contents

Architecture

The beSOURCE Jenkins plugin assumes that both Jenkins server and beSOURCE server are connected to the same version control system. It also assumes that the same source files group is configured to Jenkins' build target and beSOURCE server's analysis target.

A user can connect to the beSOURCE server to inspect source files for a specific build project in Jenkins. The beSOURCE server inspects the corresponding source files and returns the results.

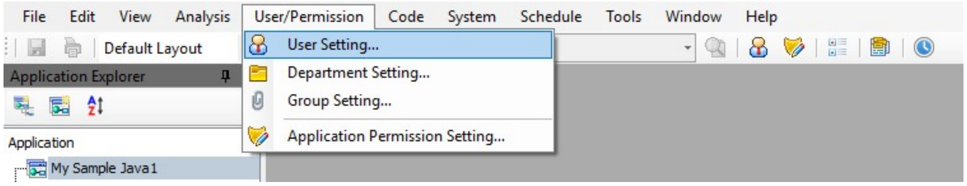


Preparation

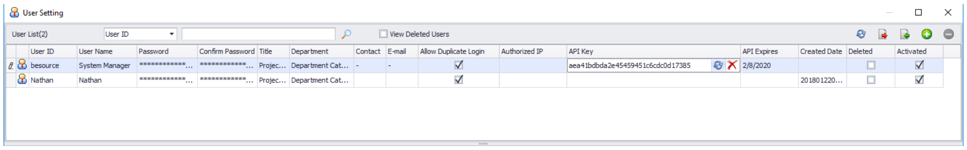
Generating the beSOURCE API key

To use the Jenkins plug-in, you must generate a RESTful API key for the beSOURCE server. To generate the API key, do the following:

1. Open the **Admin Console**
2. Log in as an beSOURCE server administrator.
3. Select **User/Permission** > **User Setting**.



4. Select a user, and then select the **Refresh** button in the **API Key** column. The new key for RESTful API communication is generated.



	User ID	User Name	Password	Confirm Password	Title	Department	Contact	E-mail	Allow Duplicate Login	Authorized IP	API Key	API Expires	Created Date	Deleted	Activated
	besource	System Manager	************...	************...	Projec...	Department Cat...	-	-	✓		aea41bdbda2e45459451c6cdc0d17385	2/8/2020		□	✓
	Nathan	Nathan	************...	************...	Projec...	Department Cat...			✓				201801220...	□	✓

beSOURCE Basic Settings

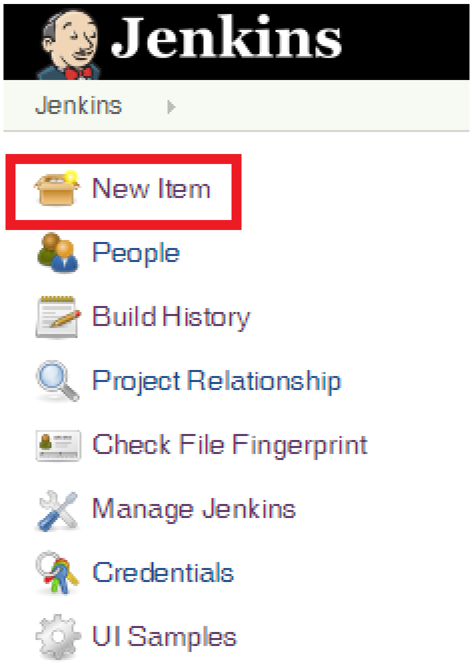
The administrator must complete the basic settings for source code analysis first. For example, Collection Unit, Analysis Unit and Application settings are required. For more information, refer to Job Management in the help files.

This document assumes that all required settings are complete.

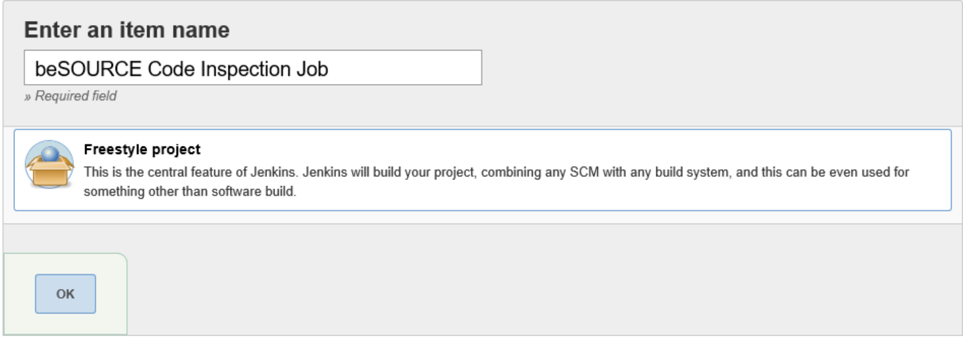
Connection Setting

To connect the Jenkins plugin to the beSOURCE server, do the following:

1. Log in to the **Jenkins server**.
2. Select **New Item**.



3. Enter the name of item in the box, and then select **Freestyle project**.



Enter an item name

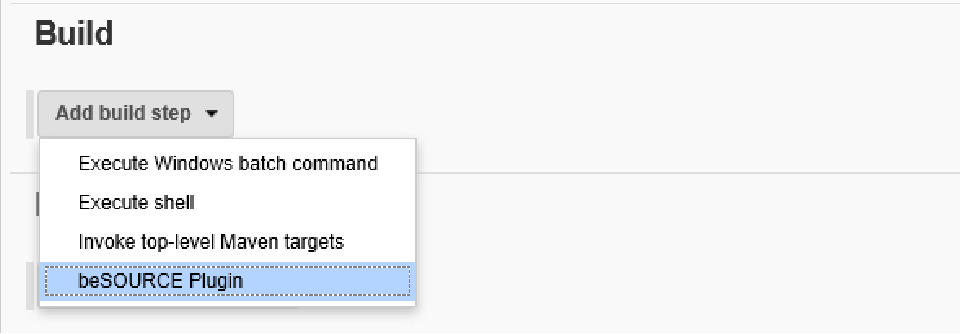
beSOURCE Code Inspection Job

» Required field

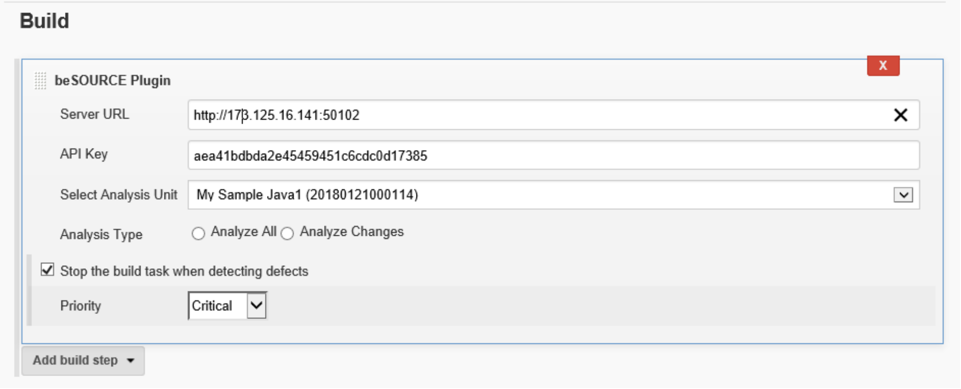
Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

4. Select **OK**.
5. Select **Build** > **Add build step** (for a Maven project, select **Pre Step** or **Post Step**), and then select **beSOURCE Plugin**.



6. Enter the settings values of the beSOURCE plugin for the following:
   - **Server URL** - The beSOURCE server URL.
   - **API Key** - The restful API key generated in the Admin Console.
   - **Select Analysis Unit** - An Analysis Unit set in the server.
   - **Analysis Type**
     - **Analyze All** - Scans all source files.
     - **Analyze Changes** - Only scans the changed source files (incremental analysis).
   - **Stop the build task when detecting defects** - If defects with the specified priority are found, they will stop task build.
   - **Priority** - **The beSOURCE** inspection rule's priority.


Build

beSOURCE Plugin

Server URL: http://173.125.16.141:50102

API Key: aea41bdbda2e45459451c6cdc0d17385

Select Analysis Unit: My Sample Java1 (20180121000114)

Analysis Type: ○ Analyze All ○ Analyze Changes

☑ Stop the build task when detecting defects

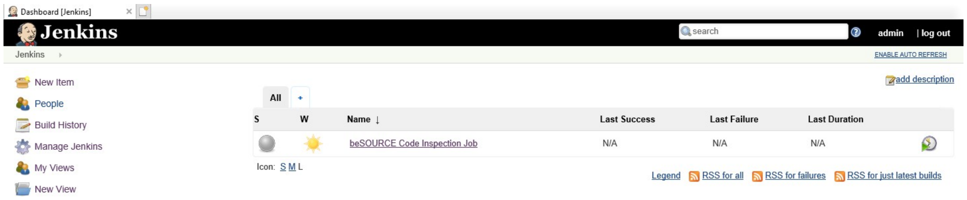
Priority: Critical

7. Select **Save**.

Run the Jenkins Plugin

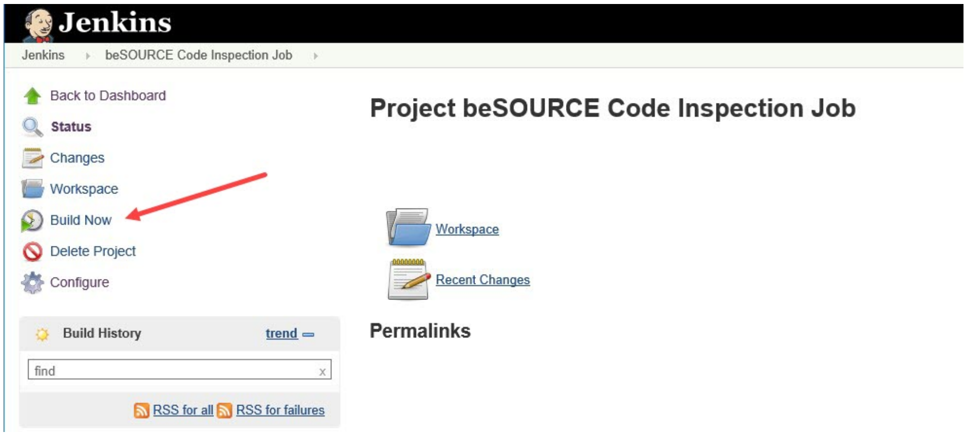
To run the Jenkins plugin, do the following:

1. Select a Jenkins job that is compatible with the beSOURCE plugin.

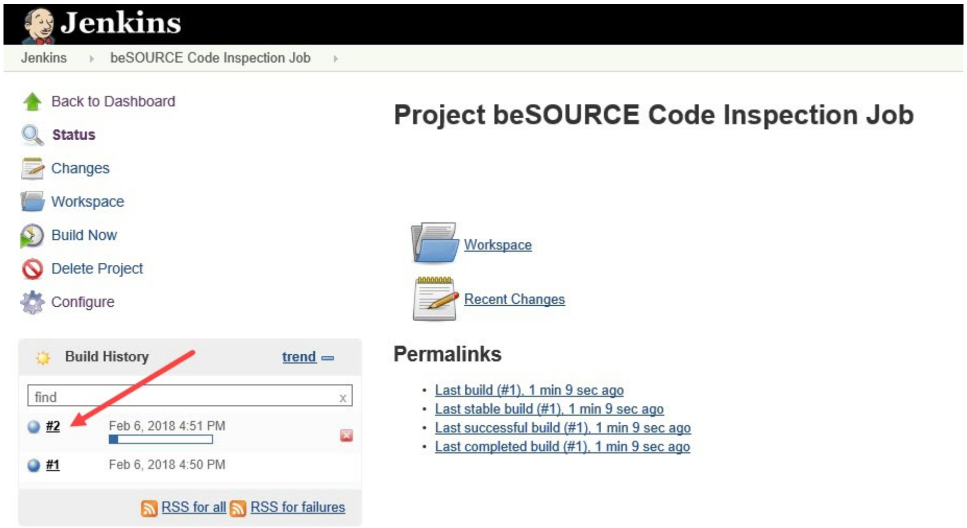


S	W	Name ↓	Last Success	Last Failure	Last Duration
		beSOURCE Code Inspection Job	N/A	N/A	N/A

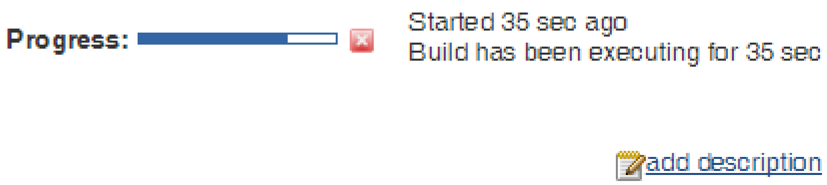
2. Select **Build Now**.



3. The build number will be shown. You can open the build job by selecting the number.

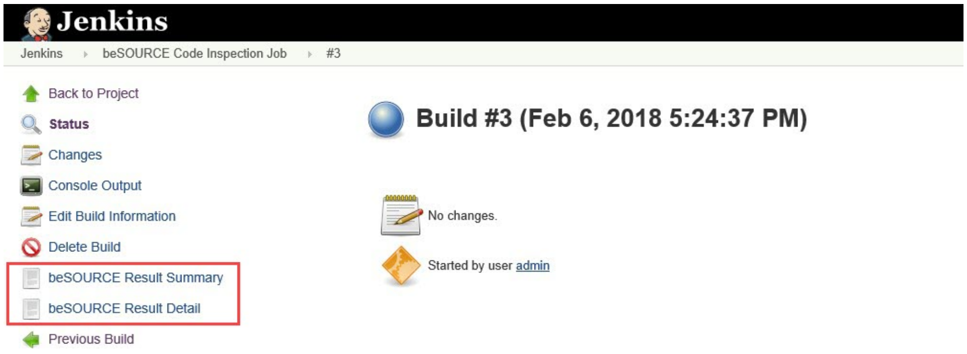

Jenkins
Jenkins
beSOURCE Code Inspection Job
Back to Dashboard
Status
Changes
Workspace
Build Now
Delete Project
Configure
Build History
trend
find
#2
Feb 6, 2018 4:51 PM
#1
Feb 6, 2018 4:50 PM
RSS for all
RSS for failures
Project beSOURCE Code Inspection Job
Workspace
Recent Changes
Permalinks
Last build (#1), 1 min 9 sec ago
Last stable build (#1), 1 min 9 sec ago
Last successful build (#1), 1 min 9 sec ago
Last completed build (#1), 1 min 9 sec ago

4. If the build job is in progress, Jenkins will display its progress.



Started 35 sec ago
Build has been executing for 35 sec

5. Once the build job is complete, select the build number. The **beSOURCE Result Detail** and **beSOURCE Result Summary** menus are shown.

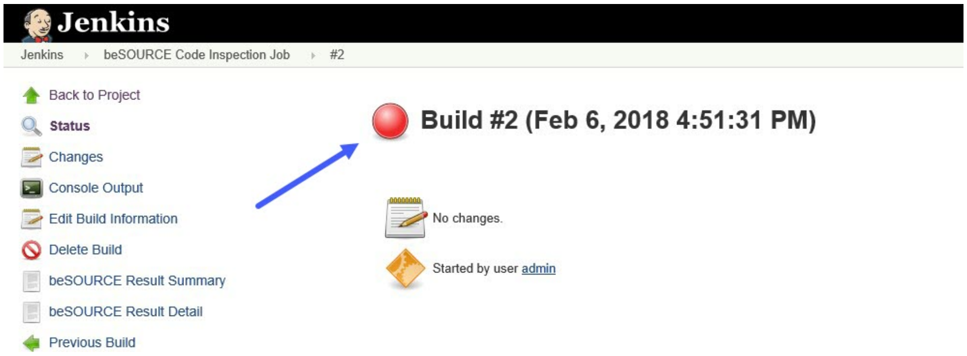


Build #3 (Feb 6, 2018 5:24:37 PM)

No changes.

Started by user admin

6. If you selected **Stop the build task when detecting defects** and defects with predefined priority are found, the build job will stop (red bulb).



Build #2 (Feb 6, 2018 4:51:31 PM)

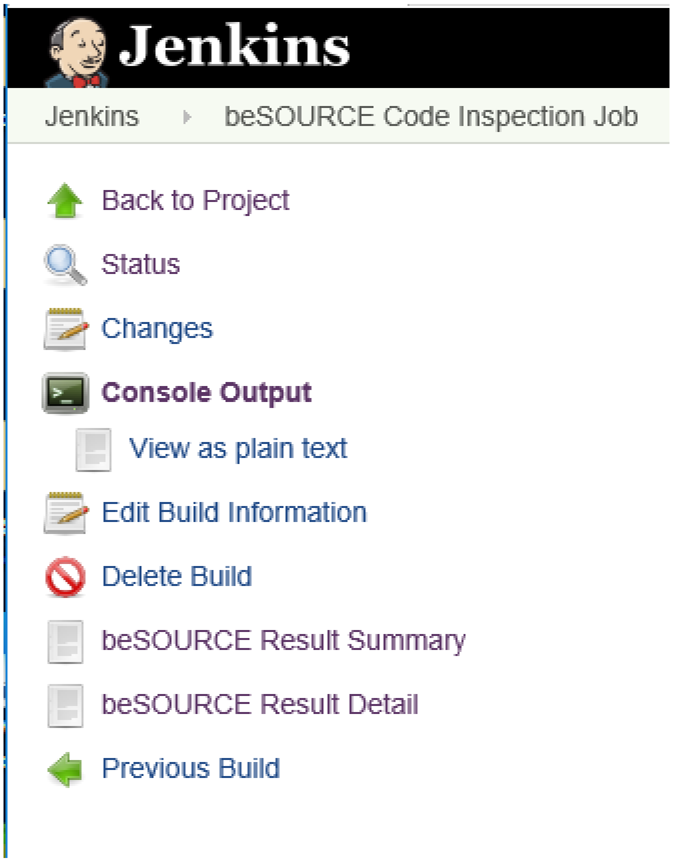
No changes.

Started by user admin

NOTE: The bulb turns red when the job is stopped due to a rule violations. It does not indicate a failure.

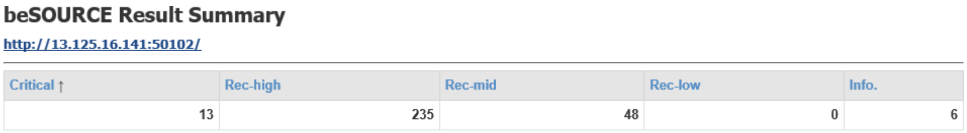
Checking Inspection Results

You can view the inspection results by selecting **beSOURCE Result Detail** or **beSOURCE Result Summary** in the completed build job.



beSOURCE Result Summary

The **beSOURCE Result Summary** shows the summary of code inspection results. The rule violation count by priority is shown in this image.



beSOURCE Result Summary

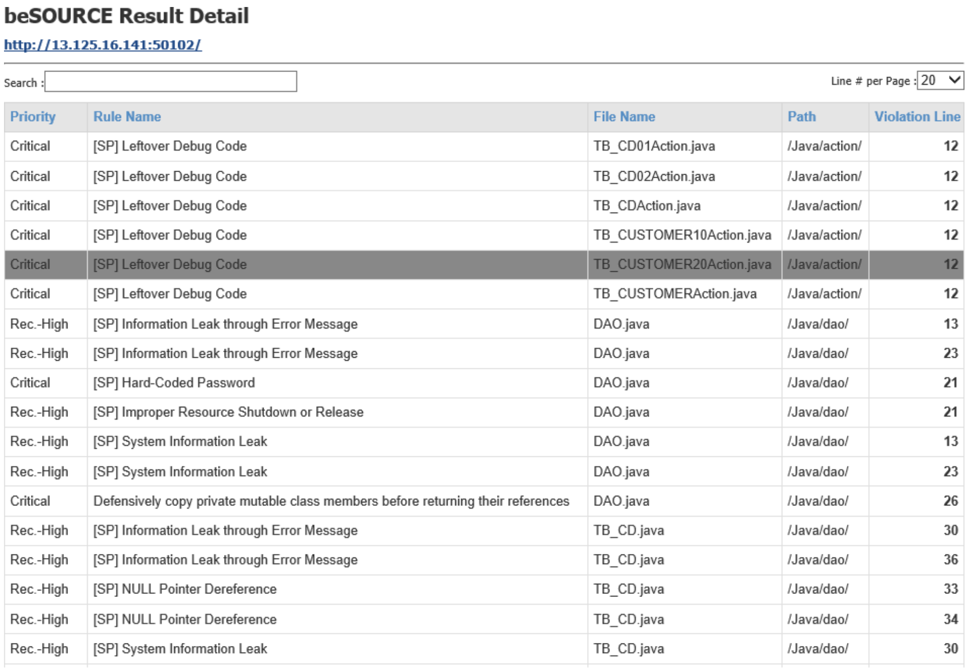
http://13.125.16.141:50102/

Critical ↑	Rec-high	Rec-mid	Rec-low	Info.
13	235	48	0	6

beSOURCE Result Detail

The **beSOURCE Result Detail** menu shows the following detailed results:

- **Search** - You can search by priority, rule name and file name.
- **Line # per Page** - Sets the violations count to show in a page.
- **Priority** - The rule’s priority.
- **Rule Name** - The name of the rule.
- **File Name** - The file name that has rule violations.
- **Path** - The path of the rule violation file.
- **Violation Line** - The line number of the rule violation.



beSOURCE Result Detail

http://13.125.16.141:50102/

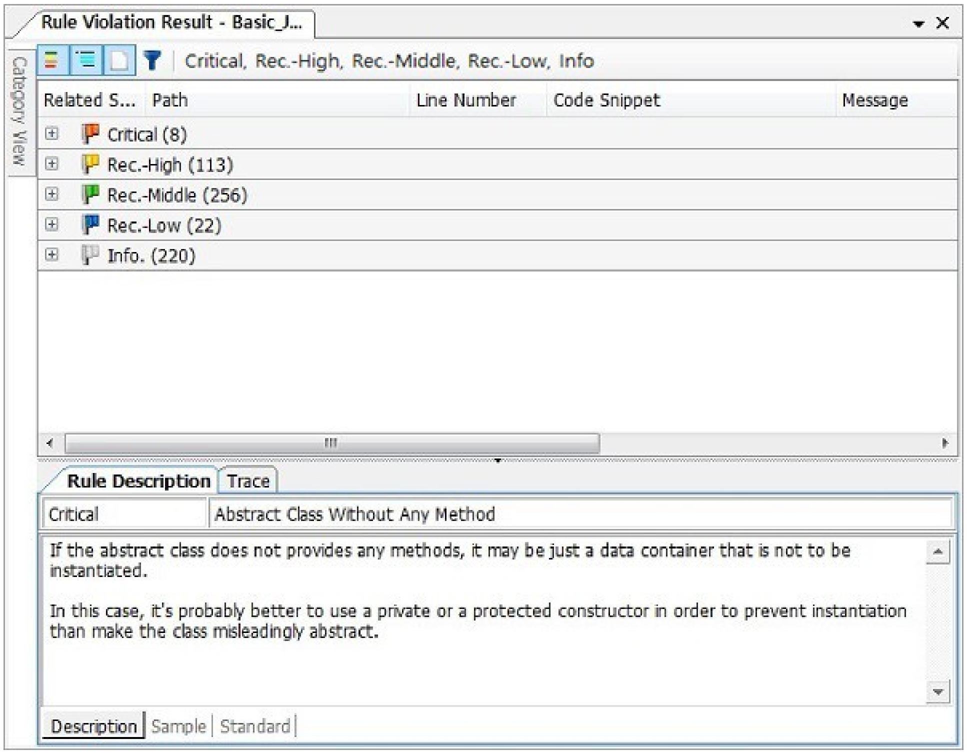
Search :

Line # per Page : 20

Priority	Rule Name	File Name	Path	Violation Line
Critical	[SP] Leftover Debug Code	TB_CD01Action.java	/Java/action/	12
Critical	[SP] Leftover Debug Code	TB_CD02Action.java	/Java/action/	12
Critical	[SP] Leftover Debug Code	TB_CDAction.java	/Java/action/	12
Critical	[SP] Leftover Debug Code	TB_CUSTOMER10Action.java	/Java/action/	12
Critical	[SP] Leftover Debug Code	TB_CUSTOMER20Action.java	/Java/action/	12
Critical	[SP] Leftover Debug Code	TB_CUSTOMERAction.java	/Java/action/	12
Rec.-High	[SP] Information Leak through Error Message	DAO.java	/Java/dao/	13
Rec.-High	[SP] Information Leak through Error Message	DAO.java	/Java/dao/	23
Critical	[SP] Hard-Coded Password	DAO.java	/Java/dao/	21
Rec.-High	[SP] Improper Resource Shutdown or Release	DAO.java	/Java/dao/	21
Rec.-High	[SP] System Information Leak	DAO.java	/Java/dao/	13
Rec.-High	[SP] System Information Leak	DAO.java	/Java/dao/	23
Critical	Defensively copy private mutable class members before returning their references	DAO.java	/Java/dao/	26
Rec.-High	[SP] Information Leak through Error Message	TB_CD.java	/Java/dao/	30
Rec.-High	[SP] Information Leak through Error Message	TB_CD.java	/Java/dao/	36
Rec.-High	[SP] NULL Pointer Dereference	TB_CD.java	/Java/dao/	33
Rec.-High	[SP] NULL Pointer Dereference	TB_CD.java	/Java/dao/	34
Rec.-High	[SP] System Information Leak	TB_CD.java	/Java/dao/	30

More Detailed Results

For more detailed inspection results, you can use the **beSOURCE Client**. For more information, refer to the Show Rule Violations Panel topic in the beSOURCE help files.



Rule Violation Result - Basic_J...
Category View
Critical, Rec.-High, Rec.-Middle, Rec.-Low, Info
Rule Description
Trace
Description
Sample
Standard

Related S...	Path	Line Number	Code Snippet	Message
⊞ Critical (8)				
⊞ Rec.-High (113)				
⊞ Rec.-Middle (256)				
⊞ Rec.-Low (22)				
⊞ Info. (220)				

Critical	Abstract Class Without Any Method

If the abstract class does not provides any methods, it may be just a data container that is not to be instantiated.

In this case, it's probably better to use a private or a protected constructor in order to prevent instantiation than make the class misleadingly abstract.