

CodeCheck

User Guide

Issue 01
Date 2022-09-30



Copyright © Huawei Technologies Co., Ltd. 2022. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Contents

1 What Is CodeCheck?	1
2 Relationships Between Members and Permissions	3
3 CodeCheck Process	5
4 Creating a Task	8
4.1 Preparations	8
4.2 Creating a CodeHub Source Code Check Task	9
4.3 Creating a Task to Check Git Code	10
5 (Optional) Configuring a Rule Set	12
5.1 Configuring a CodeCheck Rule Set	12
5.2 Customizing a CodeCheck Rule Set	14
5.3 Viewing Rule Details	15
5.4 Deleting a Custom CodeCheck Rule Set	17
6 (Optional) Configuring a Task	18
6.1 Overview	18
6.2 Modifying or Deleting a Source Code Check Task	20
6.3 Configuring Multi-Branch Check	21
6.4 Configuring Paths of Ignored Files	21
6.5 Configuring Quality Gates	22
6.6 Configuring a Task Execution Plan	23
6.7 Enabling and Disabling Notification	23
6.8 Configuring a Check Mode	24
6.9 Configuring an Automatic Check Task for Branch Merge	24
7 Running the Task	26
7.1 Running a Check Task	26
7.2 Viewing Check Details	27
7.3 Viewing Code Issues	28

1 What Is CodeCheck?

What Is CodeCheck?

CodeCheck is a cloud-based management service that checks code quality. Developers can easily perform static code and security checks in multiple languages and obtain comprehensive quality reports. CodeCheck also allows developers to view defects by group and provides Improvement suggestions, effectively securing high quality and helping achieve business success.

Main Features of CodeCheck

- Supports mainstream encoding languages.
Java, C++, PHP, JavaScript, TypeScript, HTML, CSS, Go, and Python
- Is compatible with mainstream security standards in the industry.
CWE, OWASP TOP 10, SANS TOP 25, MISRA, and CERT
- Provides domain-specific one-stop solutions.
Security, , and coding guidelines
- Is seamlessly integrated into the DevCloud process.
Offers multi-branch and MR checks.
- Provides impact description, change examples, and suggestions for code defects.
- Accurately locates code lines and allows users to view and rectify code defects online.
- Focuses on handling new defects.
Prevent generation of new technical debt.
- Automatically assigns defect owners.
The defect ownership accelerates closure of quality issues.
- Has more metrics.
Code cyclomatic complexity (built-in risk measurement system) and code repetition rate
- Other aspects

CodeCheck supports scheduled triggering, code commit triggering, and code check result notification.

2 Relationships Between Members and Permissions

Role Permission Table

Table 2-1 describes the default role permissions of CodeCheck users to deal with resources (such as tasks and rule sets) in a project.

Table 2-1 Default role permissions for CodeCheck

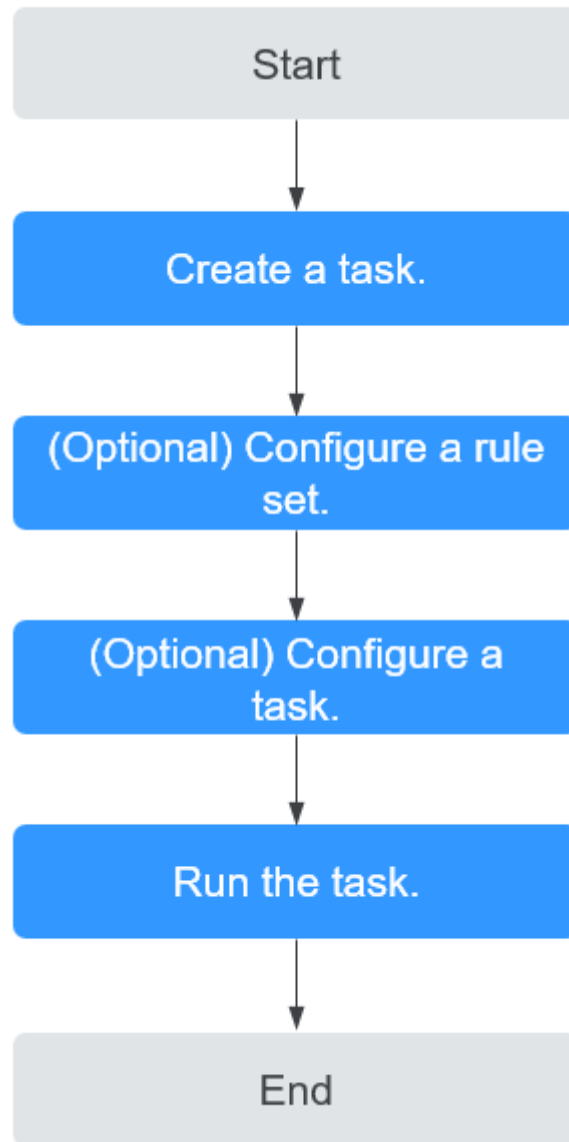
Role	CodeCheck Task	Rule Set	Rule	Issue
Project creator	<ul style="list-style-type: none">• Create, view, delete, execute, and modify tasks of all members in a project.	Create, delete, and modify personal rule sets; view and use others' rule sets.	View	<ul style="list-style-type: none">• View an issue list.• Export an issue list.• View issue details.• Create an issue ticket.• Handle issues.
Project manager				
Developer	<ul style="list-style-type: none">• Create, view, execute, modify, and delete personal tasks.• Have permissions to view, execute, and modify others' tasks in the same code repository.			
Test manager	<ul style="list-style-type: none">• Have no permissions to create or view tasks.	Create, delete, and modify personal rule sets, and view others' rule sets.		-
Tester				-

Role	CodeCheck Task	Rule Set	Rule	Issue
Participant				-
Viewer				-
Custom role				-

3 CodeCheck Process

CodeCheck Process

This topic describes the basic process of CodeCheck.

Figure 3-1 Basic operation process

Process Description

Process	Description
Create a task	You can create different types of CodeCheck tasks. For details, see 4 Creating a Task .
(Optional) Configure a rule set	CodeCheck supports default rule sets of C, Java, C++, PHP, JavaScript, TypeScript, HTML, CSS, Go, and Python languages. You can add, copy, modify, or delete a custom rule set as required. For details, see 5 (Optional) Configuring a Rule Set .

Process	Description
(Optional) Configure a task	You can modify or configure an existing check task. For details, see 6.1 Overview .
Run the task	After the task is created, go to the task details page and start the check task. After the check is complete, you can view task details and fix code defects. For details, see 7 Running the Task .

4 Creating a Task

[4.1 Preparations](#)

[4.2 Creating a CodeHub Source Code Check Task](#)

[4.3 Creating a Task to Check Git Code](#)

4.1 Preparations

Before creating a CodeCheck task, make the following preparations:

- [Creating a Project](#)
- [Creating a Code Repository](#) (for CodeHub Code Source)
- [CodeCheck Entry](#)

Creating a Project

Create a Scrum project in ProjectMan. For details, see [Creating a Project](#).

Creating a Code Repository

Finish [Creating a Code Repository](#) in CodeHub.

Creating a Service Endpoint

On the project details page, choose **Settings > General Settings > Service Endpoints**. The **Service Endpoint** page is displayed. Create a service endpoint. For details, see [Service Endpoint](#).

CodeCheck Entry

You can access CodeCheck from a project or the homepage.

- **From a project**
 - a. On the DevCloud homepage, click a target project name. The project details page is displayed.

- b. Choose **Code > CodeCheck**. The CodeCheck page is displayed.
- **From the homepage**
On the DevCloud homepage, choose **Services > CodeCheck**. The CodeCheck homepage is displayed.

Task	Quality Gate	Issue	Last Check	Operation
groovy		0 Version Issues	629 Lines of Codes	3 weeks ago
JavaMaven_3566	Passed	0 Version Issues	5 Lines of Codes	2 weeks ago
test12_28		Start Check		
test12_27		Start Check		
test12_26		Start Check		

4.2 Creating a CodeHub Source Code Check Task

This section describes how to create a CodeHub check task. For details, see [4.1 Preparations](#).

NOTE

- If you select **Allow automated creation of a code check task** when creating a cloud repository in CodeHub, you can view the check task of the repository in the CodeCheck task list after the repository is created. For operation details, see [Creating a Cloud Repository](#).

Permissions

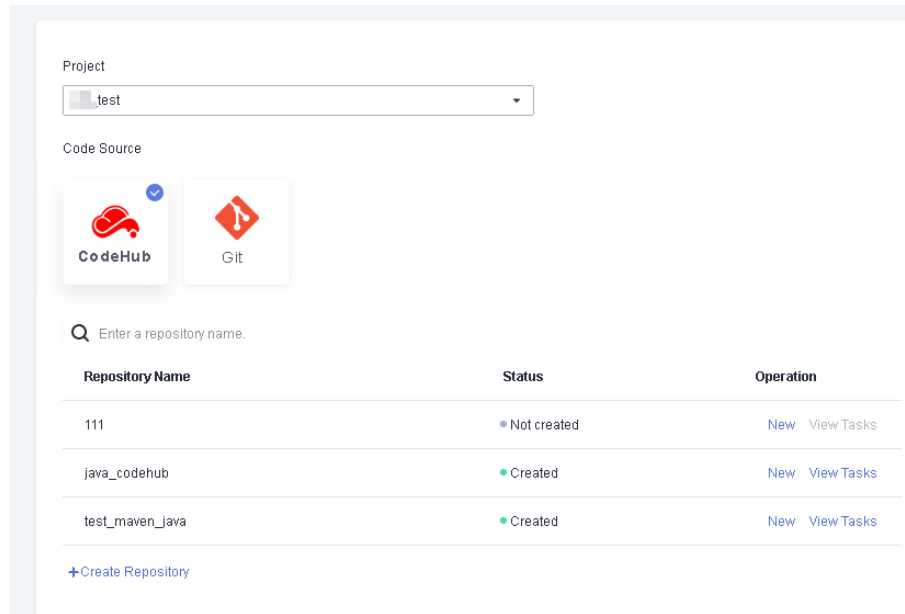
- Allow project members to access the repository
- Allow generation of a README file
- Allow automated creation of a code check task (free)

- This section how to create a task to check CodeHub code if you do not select **Allow automated creation of a code check task** when creating a cloud repository in CodeHub.

Procedure

- Step 1** On the DevCloud homepage, click a target project name. The project details page is displayed.
- Step 2** Choose **Code > CodeCheck**. The CodeCheck page is displayed.
- Step 3** Click **Create Task**.

On the page for creating a task, set the code source to **CodeHub**.



NOTE

If the CodeHub code source is not available in the project, see [4.1 Preparations](#).

Step 4 Click **Create**.

After the creation is complete, go to the CodeCheck details page and run the task as required.

----End

4.3 Creating a Task to Check Git Code

This section describes how to create a task to check Git code. For details, see [4.1 Preparations](#).

Procedure

- Step 1** On the DevCloud homepage, click a target project name. The project details page is displayed.
- Step 2** Choose **Code > CodeCheck**. The CodeCheck page is displayed.
- Step 3** Click **Create Task**.

On the page for creating a task, set the code source to **Git**.

Project
test2

Code Source
CodeHub Git

Endpoint [Manage](#) [Refresh](#)
No Endpoint

Repository

Branch

Language

OK Cancel

NOTE

If the Git code source code source is not prepared in the project, see [Preparations](#).

Step 4 Configure the endpoint instance, repository, branch, and check language based on site requirements.

Step 5 Click **Create**.

After the creation is complete, go to the CodeCheck details page and run the task as required.

----End

5 (Optional) Configuring a Rule Set

[5.1 Configuring a CodeCheck Rule Set](#)

[5.2 Customizing a CodeCheck Rule Set](#)

[5.3 Viewing Rule Details](#)

[5.4 Deleting a Custom CodeCheck Rule Set](#)

5.1 Configuring a CodeCheck Rule Set

By default, CodeCheck supports rule sets in Java, C++, PHP, JavaScript, TypeScript, HTML, CSS, Go, and Python, and each language type corresponds to multiple rule sets of different levels. [Table 5-1](#) lists system rule sets.

NOTE

- You can add, copy, modify, or delete a rule set as required.
- On the DevCloud homepage, choose **Services > CodeCheck**. On the CodeCheck homepage that is displayed, click the **Rule Sets** tab. All system rule sets are displayed.

Table 5-1 System rule sets

Language	System Rule Set
Java	<ul style="list-style-type: none">• Key check rule set• General check rule set• Comprehensive check rule set• Security check rule set
CSS	<ul style="list-style-type: none">• General check rule set
JS	<ul style="list-style-type: none">• Key check rule set• General check rule set• Comprehensive check rule set

Language	System Rule Set
TypeScript	<ul style="list-style-type: none"> ● Key check rule set ● General check rule set ● Comprehensive check rule set
HTML	<ul style="list-style-type: none"> ● General check rule set
C++	<ul style="list-style-type: none"> ● Key check rule set ● General check rule set ● Comprehensive check rule set
PHP	<ul style="list-style-type: none"> ● Key check rule set ● General check rule set
Go	<ul style="list-style-type: none"> ● Key check rule set ● General check rule set ● Comprehensive check rule set
Python	<ul style="list-style-type: none"> ● Key check rule set ● General check rule set

Procedure





- Step 1** On the DevCloud homepage, choose **Services > CodeCheck**. The CodeCheck homepage is displayed.
- Step 2** Click the **Rule Sets** tab. The check rule set list page is displayed.

Rule Set	Language	Rules	Modified	Operation
cc	C	0	2022-09-15 20:06:22	✎ ☒ ⋮
C_	C	2	2022-08-19 17:07:58	✎ ☒ ⋮
JAVASCRIPT Default Rule Set	JAVASCRIPT	12	2022-09-23 15:00:28	✎ ☒ ⋮
Rule001	C	0	2022-09-19 18:15:45	✎ ☒ ⋮
cts-cms	C	0	2022-09-16 17:48:38	✎ ☒ ⋮
dddd	C	33	2022-08-19 16:57:48	✎ ☒ ⋮
gz111	C	0	2022-08-16 14:38:50	✎ ☒ ⋮
secsolar	C	0	2022-09-17 14:29:31	✎ ☒ ⋮
testctss	C	0	2022-09-15 20:13:39	✎ ☒ ⋮
wotest	C	6	2022-08-18 10:55:40	✎ ☒ ⋮

Add, modify, delete, view, copy, or set rule sets as required.

Table 5-2 Rule set operation description

Operation	Description
Add a rule set	On the Rule Sets tab page, click New . You can customize a rule set. For details, see 5.2 Customizing a CodeCheck Rule Set .

Operation	Description
Configure (Modifying) a rule set	Click  in the row where the rule set is located and choose Modify Basic Information . The Configure Rule Set page is displayed. You can select rule sets and set issue levels as required.
Copy a rule set	Click  in the row where the rule set is located and choose Copy . NOTE A rule set name must be unique.
Default rule set (under a project)	On the project details page, choose Code > Code Check > Rule Sets . On the Rule Sets tab page that is displayed, click  in the row where the rule set is located and select defaultSet to set it as the default check rule set of a CodeCheck task in the project.
Delete a rule set	To delete a rule set, click  in the row where the rule set is located and choose Delete . For details, see 5.4 Deleting a Custom CodeCheck Rule Set . NOTE <ul style="list-style-type: none">• System rule sets and rule sets in use cannot be deleted.• System rule sets can only be viewed and copied.
View a rule set	You can click a rule set name to view its details.
Configure a rule set	Go to a check details page, choose Settings > Rule Sets , select the created rule set, and set it as the check rule set for the task. For details, see 5.2 Customizing a CodeCheck Rule Set .

----End

5.2 Customizing a CodeCheck Rule Set

- In addition to default rule sets, you can also customize rule sets. After customization, use the custom rule set for a CodeCheck task.
- At least one rule must be set for each rule set template.

Prerequisites

Currently, only rule sets in a single language can be configured. That is, only check rules in the same language can be configured for one rule set.

Procedure

- Step 1** On the CodeCheck homepage, click the **Rule Sets** tab. The **Rule Sets** tab page is displayed by default.
- Step 2** Click **New** to add a custom rule set.

In the **Add Rule Set** dialog box that is displayed, set the rule set name, Language, and rule set base, and click **OK**.

Add Rule Set ✕

* Rule Set:
Rule01

* Language:
C

Replicated in:
Select rule sets

Remarks:


OK Cancel

Step 3 Configure check rules for the rule set.

Select a rule name and set the issue level as required.

Name	Tag	Issue Level
<input checked="" type="checkbox"/> G.FUU.1: [redacted]	codemars	Minor
<input type="checkbox"/> G.FUU.1f: [redacted]	codemars	
<input type="checkbox"/> G.INC.07: [redacted]	fixbotengine-cxx	
<input type="checkbox"/> G.INC.08: [redacted]	fixbotengine-cxx	

Step 4 Change the check rule set used by the CodeCheck task to the custom rule set.

Go to the CodeCheck details page, choose **Settings > Rule Sets**, enable  for the target language, and select the custom rule set configured for the target language.

Languages Included Check Parameter

JAVA javaNoCompile

Enable Rule Set

JAVA Default Rule Set Custom

Secrrella Custom

javaNoCompile Custom

javaset Custom

----End

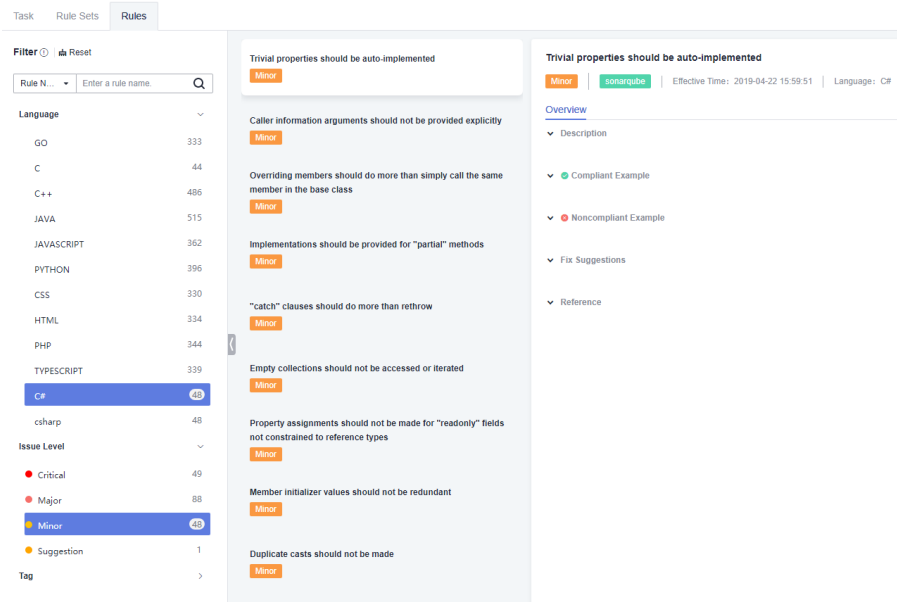
5.3 Viewing Rule Details

Rule details include the issue levels (such as **Minor**), compliant examples, noncompliant examples, and fix suggestions.

Procedure: Method 1

Step 1 Click the **Rules** tab on the CodeCheck homepage. The **Rules** tab page is displayed.

Step 2 Click a language type. The corresponding rule details are displayed on the right.



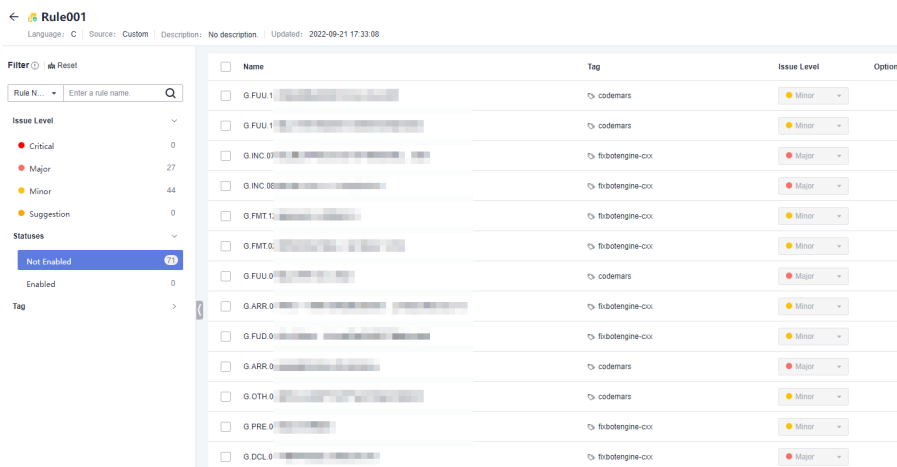
----End

Procedure: Method 2

Step 1 Click the **Rule Sets** tab on the CodeCheck homepage.

Step 2 Click a rule set name. The rule list page is displayed.

Step 3 Click a rule name. The rule details are displayed on the right.



----End


5.4 Deleting a Custom CodeCheck Rule Set




If a check rule set is associated with a CodeCheck task, the check rule set cannot be deleted. After deleting the CodeCheck task that uses the rule set or associating the CodeCheck task with another rule set, you can delete the custom rule set.

Prerequisites

- Only a creator can delete the created rule set.
- System rule sets and rule sets in use cannot be deleted.

Procedure

- Step 1** Click the **Rule Sets** tab. The check rule set list page is displayed.
- Step 2** If the check rule set is in use, delete the CodeCheck task that uses the rule set or use another rule set.
- Step 3** To delete a check rule set, click  in the row where the rule set is located and choose **Delete**.

Rule001	C	0	2022-09-21 17:33:08	  
hj	C	0	2022-09-06 17:18:46	   deleteSet

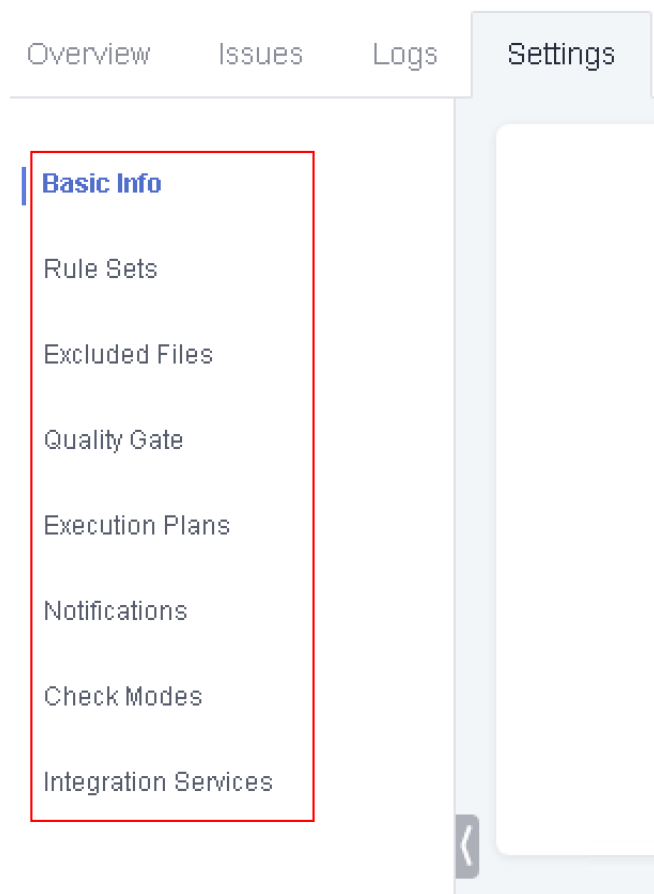
----End

6 (Optional) Configuring a Task

- [6.1 Overview](#)
- [6.2 Modifying or Deleting a Source Code Check Task](#)
- [6.3 Configuring Multi-Branch Check](#)
- [6.4 Configuring Paths of Ignored Files](#)
- [6.5 Configuring Quality Gates](#)
- [6.6 Configuring a Task Execution Plan](#)
- [6.7 Enabling and Disabling Notification](#)
- [6.8 Configuring a Check Mode](#)
- [6.9 Configuring an Automatic Check Task for Branch Merge](#)

6.1 Overview

You can modify or configure an existing check task. For details about how to configure a task, see [Table 6-1](#).

**Table 6-1** Configuring a CodeCheck task

Task Configuration	Operation Description
Basic info	You can modify task names, modify the default branch , delete a branch, and delete a task. For details, see 6.2 Modifying or Deleting a Source Code Check Task .
Rule Sets	Configure a check rule set for the task. For details, see 5.2 Customizing a CodeCheck Rule Set .
Excluded Files	For details, see 6.4 Configuring Paths of Ignored Files .
Quality Gate	For details, see 6.5 Configuring Quality Gates .
Execution Plans	For details, see 6.6 Configuring a Task Execution Plan .
Notifications	For details, see 6.7 Enabling and Disabling Notification .
Check Modes	For details, see 6.8 Configuring a Check Mode .
Integration Services	For details, see 6.9 Configuring an Automatic Check Task for Branch Merge .

6.2 Modifying or Deleting a Source Code Check Task

You can modify the basic information about a task as required, such as the check task name and default branch. You can also delete a check task.

Prerequisites

A task creator, project creator, or project administrator can modify or delete a check task.

Modifying the Name and Default Branch of a Check Task

- Step 1** Search for the target task on the CodeCheck homepage.
- Step 2** Click the task name in the row where the check task is located. The check details page is displayed.
- Step 3** Click the **Settings** tab and choose **Basic Info**. You can modify the task name and default branch.

The screenshot displays a configuration interface for a task. It is divided into four sections:

- Task Name:** A text input field containing a blurred name, followed by a blue **Rename** button.
- Default Branch:** A dropdown menu currently set to 'master', with a **Save** button to its right. Below the dropdown is the text: "The default branch affects task and task details display, as well as the branches to be checked periodically."
- Delete Task:** A **Delete Task** button. Below it is the text: "All data in the task will be deleted and cannot be recovered."
- Branch Deleted:** A dropdown menu, followed by a **Branch Deleted** button. Below the dropdown is the text: "Some branches of the task will be deleted."

----End

Deleting the Check Task

- Step 1** Search for the target task on the CodeCheck homepage.
- Step 2** Click a check task name. The check details page is displayed.
- Step 3** Click the **Settings** tab and choose **Basic Info**
- Step 4** Click **Delete Task** and complete deletion as prompted.

----End


6.3 Configuring Multi-Branch Check

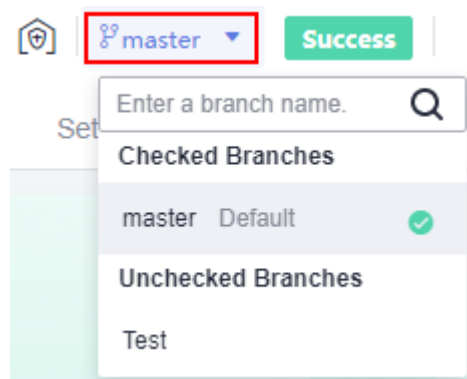
On the CodeCheck details page, you can switch among branches.

Prerequisites

Only CodeHub check tasks are supported. For details, see [4.2 Creating a CodeHub Source Code Check Task](#).

Procedure

Step 1 On the CodeCheck details page, the **Overview** tab page is displayed by default. You can click  to switch among different branches.



Step 2 After switching to the target branch, click **Start Check**.

----End

6.4 Configuring Paths of Ignored Files

You can configure the file scope by task.

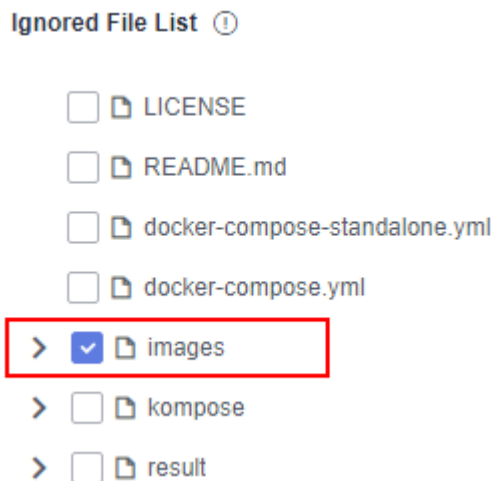
- If the file scope path is empty, that is, no file scope path is selected, all code of the selected repository branch is checked.
- If you specify a file path, the files in the specified path are ignored during a check.
- The file path cannot be changed.

Procedure

Step 1 Access the CodeCheck details page.

Step 2 Click the **Settings** tab and choose **Excluded Files**.

Step 3 Configure the ignored file list.



Step 4 After the file path is configured, run the check task. The check will take effect based on configurations. (For example, the **images** file is ignored as shown in the preceding figure).

----End

6.5 Configuring Quality Gates

Configure quality gate thresholds based on requirements. If a threshold is exceeded, the system displays a message indicating that the gate result is **Failed**.

Procedure

Step 1 Access the CodeCheck details page.

Step 2 Click the **Settings** tab and choose **Quality Gate**.

Step 3 Configure the gate switch and threshold based on quality requirements. Save configurations.

Item	Operator	Threshold	Enable
Critical	≤	15	<input checked="" type="checkbox"/>
Major	≤	6	<input checked="" type="checkbox"/>
Minor	≤	210	<input checked="" type="checkbox"/>
Suggestion	≤	0	<input type="checkbox"/>

Step 4 Click **Start Check** and view the check result.

----End

6.6 Configuring a Task Execution Plan

Configure an execution plan of a check task as required.

Procedure

- Step 1** Access the CodeCheck details page.
- Step 2** Click the **Settings** tab page and choose **Execution Plans**.
- Step 3** Enable **Scheduled Check**.

You can configure a day or time segment in a week to execute a check task.

Scheduled Check ⓘ

Time :

Mon Tue Wed Thu Fri Sat Sun 00:00

Save

----End

6.7 Enabling and Disabling Notification

You can configure notification for **Task Checked** and **Delete Task** as required.

Currently, **Notify** and **Email** are supported.

Procedure

- Step 1** Access the CodeCheck details page.
- Step 2** Click the **Settings** tab and choose **Notifications**.
- Step 3** Enable or disable the notification status of an event type as required.

Currently, **Task Checked** and **Delete Task** are supported.

You can select **Notify** or **Email**.

Event	Notify ⓘ	Email ⓘ
Task Checked	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Delete Task	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

----End

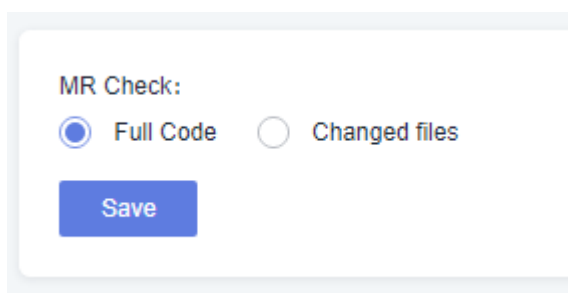
6.8 Configuring a Check Mode

You can configure a check mode as required. The **MR Check** mode has **Full Code** and **Changed files** options.

Step 1 Access the CodeCheck details page.

Step 2 Click the **Settings** tab and choose **Check Modes**.

By default, **Full Code** is selected. Change the mode as required and save the configuration.



----End

6.9 Configuring an Automatic Check Task for Branch Merge

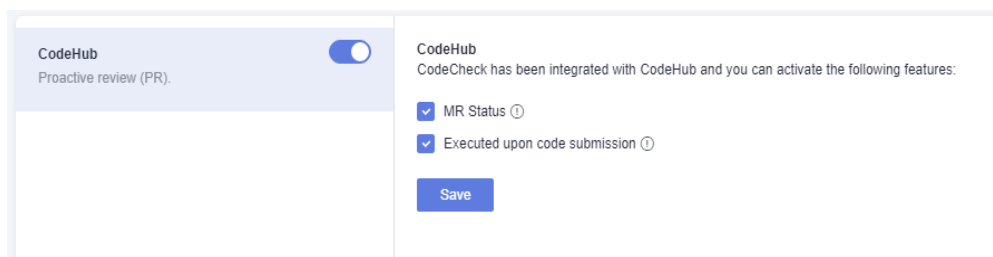
If branch merging exists in CodeHub, you can configure an automatic check task in CodeCheck and generate a job. That is, if branch merging exists in a code repository, a new code check job is generated.

Procedure

Step 1 Access the CodeCheck details page.

Step 2 Click the **Settings** tab and choose **Integration Services**. Enable **CodeHub**. The CodeHub integrated functions are displayed.

Step 3 Select **MR Status** and **Executed upon code submission** as required.




Step 4 Click **Save**. After the file is saved, a success message is displayed.

 **NOTE**



After **MR Status** is selected, source and target branches are automatically checked upon a code commit. After **Executed upon code submission** is selected, the check task is automatically executed upon a successful code commit.

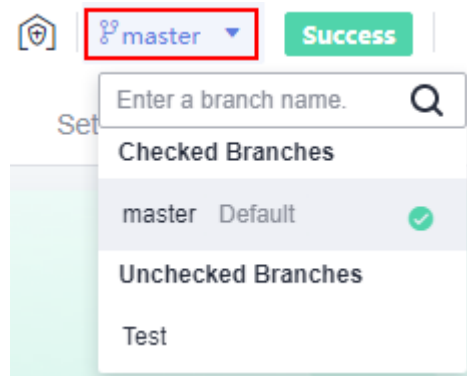
Step 5 If a branch is merged in the code repository of the task, CodeCheck automatically generates a code check job.

1. Access the check task details page.
2. Click  and create an MR in CodeHub.

 **NOTE**

For details about how to merge branches, see [CodeHub User Guide](#). After the branch is merged, only differences between code branches are checked.

3. Click  master . You can view the historical records of all MRs from the drop-down menu.



----End

7 Running the Task

- [7.1 Running a Check Task](#)
- [7.2 Viewing Check Details](#)
- [7.3 Viewing Code Issues](#)

7.1 Running a Check Task

You can perform a check task to identify defects in the source code in time.

Prerequisites


The check task has been created with related permissions.

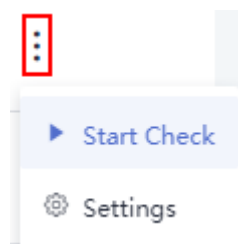
Procedure

- Step 1** Search for the target check task in the lower part of the CodeCheck homepage.
- Step 2** Click **Start Check** in the row where the code check task is located.



NOTE

If the code check task has been run, click  in the row where the task is located and choose **Start Check** to run the task again.



Step 3 Wait for a while as prompted. After the running is complete, the check details page is displayed. By default, the **Overview** tab page is displayed. For details, see [7.2 Viewing Check Details](#).

----End

7.2 Viewing Check Details

Check details include overview, code defects, check logs, and task configurations.

Prerequisites

[The check task is complete](#) and you have related permissions.

Viewing Check Details

Step 1 Search for the target task on the **Task** tab of the CodeCheck homepage.

Task	Quality Gate	Issue	Last Check	Operation
groovy		0 Version Issues	629 Lines of Codes	3 weeks ago
JavaMaven_3566	Passed	0 Version Issues	5 Lines of Codes	2 weeks ago
test12_28		Start Check		
test12_27		Start Check		
test12_26		Start Check		

Step 2 Click the task name link to view the check details.

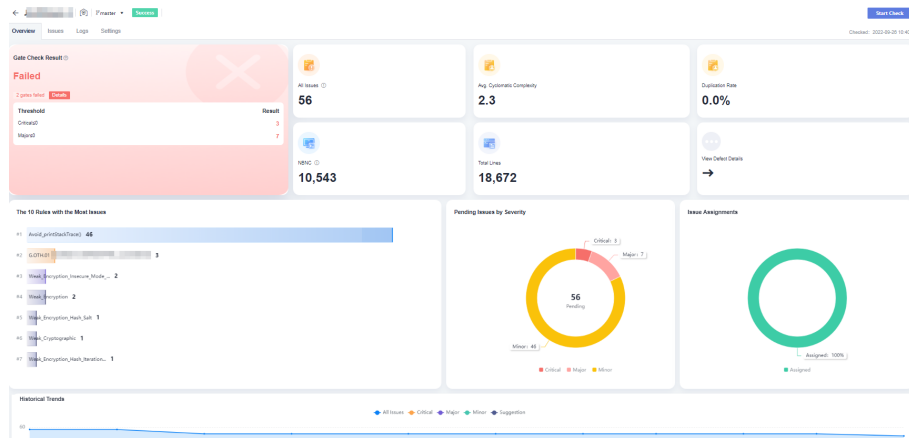
1. Overview: For details, see [Viewing the Task Overview](#).
2. Issues: You can view and fix each code issue. For details, see [7.3 Viewing Code Issues](#).
3. Logs: You can view check logs, including the execution history, step logs, and check parameters. For details, see [Viewing Check Logs](#).
4. Settings: For details, see [6.1 Overview](#).

----End

Viewing the Task Overview

After a check task is complete, the check result is displayed in charts.

On the check details page, the **Overview** tab page is displayed by default. You can view the task overview information.



7.3 Viewing Code Issues

After the check task is complete, you can view code issues and perform the following operations.

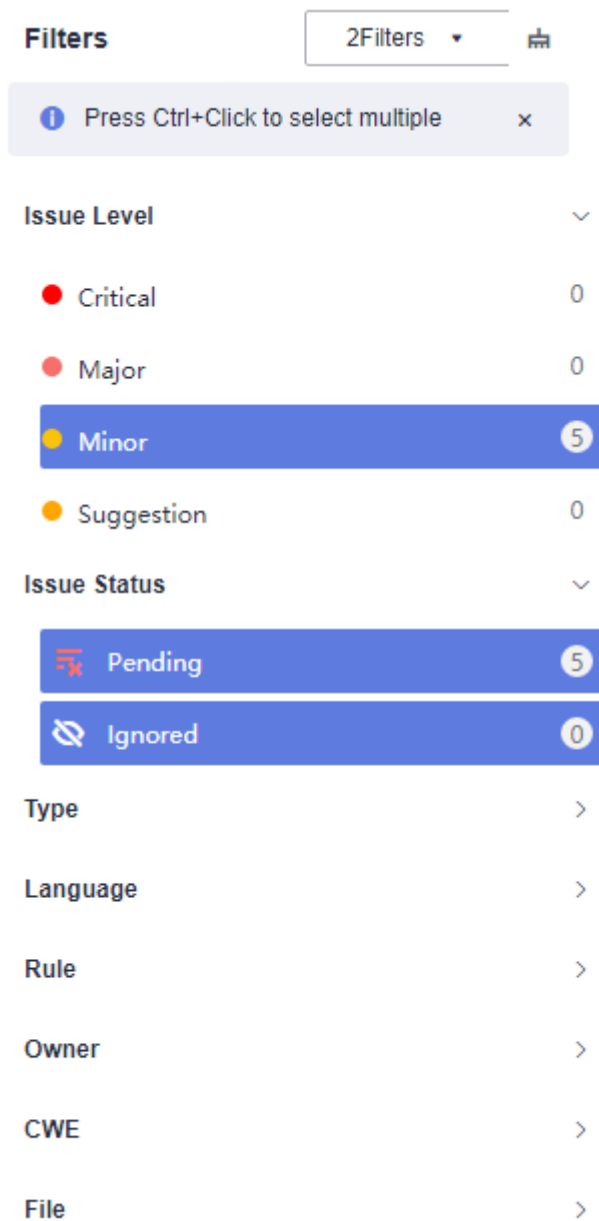
- Filter code issues based on different conditions.
- View fix suggestions for each issue.
- Assign issue owners.
- Perform batch operations on code issues.

Viewing Code Issues

Step 1 Access the check task details page.

Step 2 Click the **Issues** tab.

Filters on the left allow you to filter issues by level, status, and finding time.


**Table 7-1** Operation description for code issues

Operation	Description
Issue Level	Options include Critical , Major , Minor , and Suggestion .
Issue Status	Options include Pending and Ignored .
Type	You can filter new and all issues.
Rule	Statistics are collected based on matched rules so that users can fix the same type of issues at a time.

Operation	Description
File	Statistics are collected by source file.
Language	Statistics are collected by programming language.
Owner	Assign issue owners.
Apply to All	Click Apply to All in the upper right corner to mask all code issues or transfer all issue owners. NOTE If the number of code issues is 0, this parameter is not displayed.
Export All	Click Export All in the upper right corner of the page to export all code issues.
More	Click More in the upper right corner of the page and choose Export Masked to export all masked MR issues.

Step 3 Click **Help** to view overview information, including the issue description, compliant examples, noncompliant examples, and fix suggestions.

Step 4 Fix issues following suggestions.

Step 5 If an issue does not need to be fixed, click  to ignore the issue.

----End

Assigning Issue Owners

During check, new issues are automatically assigned to the last committer of the faulty code line.


NOTE

If the code committer is not in the code repository user list, the issues will not be allocated.

Step 1 Access the check task details page.

Step 2 Click the **Issues** tab. Use **Issue Level** and **Issue Status** filters as required.

Issue Level	▼
● Critical	0
● Major	0
● Minor	5
● Suggestion	0
Issue Status	▼
🚧 Pending	5
🚫 Ignored	0

Step 3 Click  in the issue details to set the issue owner.

----End

Performing Batch Operations for Code Issues

Perform batch operations on code issues.

Step 1 Access the check task details page.

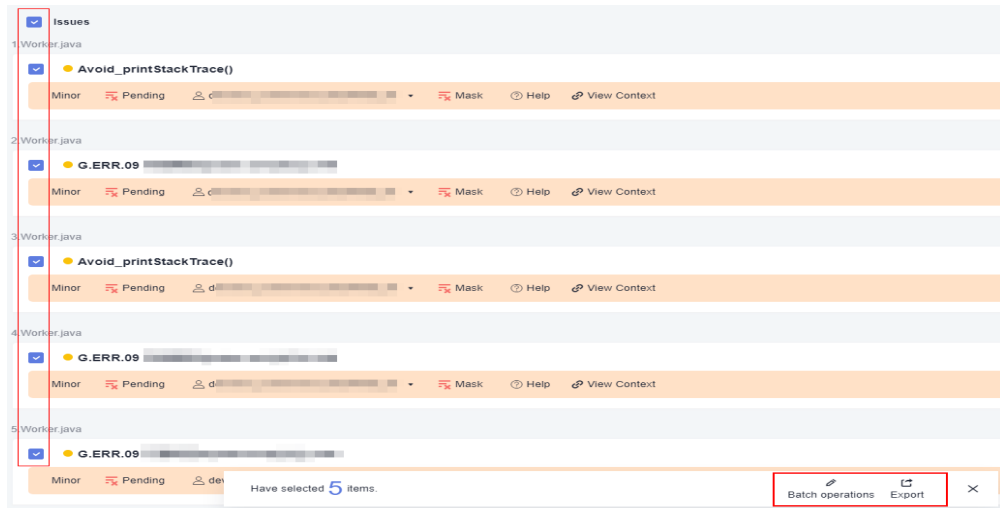
Step 2 Click the **Issues** tab.

Step 3 In the filter on the left of the page, filter code issues to be exported, for example by **Issue Level** or **Issue Status**.

Step 4 (Optional) Configure specified code issues in batches.

Select the target issue. **Batch operations** and **Export** are displayed in the lower part of the page.

- Batch operations: You can apply for blocking all code issues or transferring all issue owners.
- Export: You can export code issues.




----End

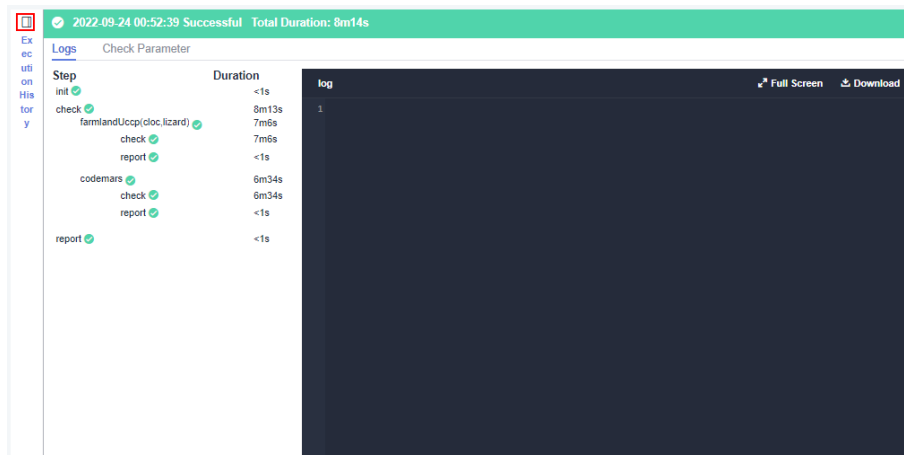
Viewing Check Logs

Step 1 Access the check task details page.

Step 2 Click the **Logs** tab.

Log details include execution history, step logs, and check parameters.

Step 3 Click  in the upper left corner to view logs in all states.



----End