

# **FUJITSU Software**

## **Agile+ Relief J V1.1.1**



# **Command Manual**

B1WD-3618-01ENZ0(00)  
January 2023

# Preface

---

The Agile<sup>®</sup> Relief J is the source program described by Java language, and an application program that supports program review by parsing the class file with compiling result of source program.

This manual describes the commands provided by Agile<sup>®</sup> Relief J.

For cautions of Code check and relevant content of tool cooperation, please refer to "Agile<sup>®</sup> Relief J Code check Operation Guide".

## Trademarks

- Microsoft, Windows, Windows Server are the registered trademarks or trademarks of Microsoft Corporation in the United States, and other countries
- Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.
- Interstage is the registered trademark of Fujitsu.
- Interstage Studio is a product of Fujitsu.
- The name FindBugs™ and the FindBugs logo are trademarked by The University of Maryland
- Android is the registered trademarks or trademarks of Google Inc.
- The names of other products and services referred to in this document are trademarks of their respective developers and manufacturers.

All rights reserved, Copyright(C) 2009-2023 FUJITSU LIMITED.

# Contents

---

Chapter 1 pgrjava Command.....	1
Chapter 2 Output Format.....	3
Chapter 3 Usage Examples.....	5
Chapter 4 Operating Environment Error.....	6

# Chapter 1 pgrjava Command

The Agile<sup>®</sup> Relief J provides the pgrjava command that executes Code check function through command line.

Before using the pgrjava command, please build and create the class file in advance.

The pgrjava command executes Code check after the Java source and class file have been input, and outputs message of rule violation as a result file.

This chapter describes the command format, options, and return code of the pgrjava command.

## Command Format

The execution format of pgrjava command is as follows.

```
"%PGRJ_HOME%\pgrjava.bat" [-XmxMaximumHeapSize] [-a ClassPathList] [-af ClassPathListFile]
[-r CheckRuleDefinitionFile] [-e {Cp1252|UTF-8}] [-qm] [-am AndroidManifestFile] [-ar
AndroidResourcePathList] [-arf AndroidResourcePathListFile] SourceFolder ClassFolder ResultFile
```

\* The environment variable PGRJ\_HOME is set when the storage folder of Agile<sup>®</sup> Relief J is being installed.

## Option

The option of the pgrjava command is as follows.

### SourceFolder

Specify with full path for the folder in which the source file to be checked is stored.

The specified folder must have the same structure as the package name.

When the folder name contains a space, it must be enclosed with quotation marks.

### ClassFolder

Specify with full path for the folder in which the class file (class file) to be checked is stored.

The specified folder must have the same structure as the package name.

When the folder name contains a space, it must be enclosed with quotation marks.

### ResultFile

Specify with full path for the name of the file that outputs the check results.

When the folder name contains a space, it must be enclosed with quotation marks.

The result file is output in CSV format. For details, please refer to "Chapter 2 Output Format".

### -XmxMaximumHeapSize

Specify the maximum heap size for code check. When this option is omitted, it is considered that 512M was specified.

It is necessary to specify the specification of the maximum heap size by using the number of byte and the character m or M that shows Mega Byte.

Example) -Xmx512m or -Xmx512M

### -a ClassPathList

Specify with full path for the class library that uses the check target.

Multiple class paths can be specified. Please separate the specification by path delimiter ";"

When the class library, class path, or class path list contains path delimiter ";" or space, they must be enclosed with quotation marks.

In addition, if the number of characters in the command line exceeds the limit, please use the -af is option.

### -af ClassPathListFile

Specify with full path for the file that describes the class path list.

When the file name contains a space, it must be enclosed with quotation marks.

The class library is described in full path in the format of one file (class library) in one line.

**-r CheckRuleDefinitionFile**

Specify the check rule definition file with full path.

When the file name contains a space, it must be enclosed with quotation marks.

When this option is omitted, the check rule definition file (rule.pgrj) under the folder that is set in the environment variable PGRJ\_HOME will be used.

**-e {Cp1252|UTF-8}**

Specify the code system of the Java source code of the check target.

When this option is omitted, the default code of the system is considered to be specified.

**-qm**

This is the output format for the analysis result consolidation mode of the Agile+ Relief. Specify this option to use the analysis result consolidation mode of the Agile+ Relief. For the analysis result consolidation mode of the Agile+ Relief, please refer to "2.3.2 Procedure of using under analysis result consolidation mode" in " Agile+ Relief Manual".

**-am AndroidManifestFile**

Specify the AndroidManifest.xml that is the manifest file of Android applications with full path.

When the file name contains a space, it must be enclosed with quotation marks.

This option is necessary for checking the following rules. For details of each rule, please refer to the "Rule Detail Manual"

pgj20001, pgj20218, pgj20403, pgj20404, pgj20405, pgj20406, pgj20409, pgj20410

**-ar AndroidResourcePathList**

Specify the resource folder of Android applications with full path.

The resource folder refers to a folder that stores any of the following folders.

`layout` folder / `layout-screen size` folder / `menu` folder / `xml` folder

Multiple resource folder paths can be specified. Please separate the specification by path delimiter ";".

When the resource folder path contains path delimiter ";" or space, they must be enclosed with quotation marks.

This option is needed for pgj20001 checking. For details of each rule, please refer to "Rule Detail Manual".

**-arf AndroidResourcePathListFile**

Specify with full path for the file that describes the Android resource path list.

When the resource folder path contains a space, it must be enclosed with quotation marks.

The resource folder path is described in full path in the format of one folder path in one line.

**Return Code**

The return codes of the pgrjava command are as follows.

Return Code	Result Category	Content
0	Normally terminated	
1	Operating environment error	The memory is insufficient etc.
2	User operation error	Option specification error etc.
3	Result file output error	No writing privilege or the capacity is insufficient etc.

The operating environment error is output to result file, and other errors are output to console.

# Chapter 2 Output Format

This chapter describes the format of result file that is output by the pgrjava command.

In the result file, the messages are output as the result of Code check.

## Output example

```
SJIS
source file name, line number, group, messageID, message
C:\Project2\src\sample\student\Main.java,30,S: Reliability, pgj10242, the method invocation that must
be noticed when in use exists. java.lang.System.exit(int)
C:\Project2\src\sample\student\Main.java,88,W: Efficiency, pgj10322, an inefficient method is
invoked. java.util.ArrayList.ArrayList()
C:\Project2\src\sample\student\Main.java,94,S: Reliability,pgj10239, there is no invocation of
java.io.BufferedReader.close() in the finally block.
C:\Project2\src\sample\student\Main.java,94,W: Functionality, pgj10405, the invocation with
possibility of file injection exists.
C:\Project2\src\sample\student\Main.java,97,S: Maintainability/Readability,pgj10061, the embedded
assignment has been used.
C:\Project2\src\sample\student\Main.java,97,W: Maintainability/Readability,pgj10068, literal is not
described on the left of comparison expression.
C:\Project2\src\sample\student\Main.java,99,W: Maintainability/Readability,pgj10068, literal is not
described on the left of comparison expression.
C:\Project2\src\sample\student\Main.java,115,S: Reliability,pgj10216,viloate to "Finally block of
try-catch must describe the processing".
```

The following table describes the output format of result file.

## Format of Result File (CSV Format)

The result file of pgrjava command is output in the following formats.

Line No.	Content	Remarks
First line	SJIS	Code system of file (fixed)
Second line	Header part	Header of message
Third line ~	Message	Content of message

### - Message

The message line is output in the following formats.

Column	Header Name	Description
1	source file name	File name of the pgr object
2	line number	Line number of the pgr object
3	group	Group of the message (combination of importance degree and category)
4	messageID	Rule ID of message
5	message	Message text

### - Group

The column of group is output in the following format.

Importance degree: category

\* ":" and importance degree are single-byte characters.

### Importance Degree

Importance degree of the rule. Any of the following values is output.

Value	Content
!	Operating environment error
S	Most important rule to be evaluated
W	Important rule to be evaluated

### Category

Rule category. Any of the following values will be output.

Value	Content
Reliability	The rule that checks the description that is likely to trigger bugs such as complex control structure and the incorrect use of API
Maintainability/Readability	The rule that checks the naming rules of class and method, as well as violation of coding rules
Efficiency	The rule that checks the operation of string and file, as well as the part, such as access to DB and XML that may cause performance deterioration
Functionality	The rule that checks the security vulnerability of SQL injection and cross site scripting (XSS) etc.
Portability	The rule that checks the part with problems during the migration to other platforms such as OS-dependent code
User rules	Fixed check rule created by users
Other	It is used during cooperation of FindBugs and PMD. It is the rule of category that is newly added in FindBugs and PMD.

## Chapter 3 Usage Examples

The examples of using the pgrjava command are as follows.

### Example 1) When the source folder is different from the class folder

```
C:\Product\Project1
+- src
  +- package
    +- Test1.java
+- bin
  +- package
    +- Test1.class
```

Command

```
"%PGRJ_HOME%\pgrjava.bat" C:\Product\Project1\src C:\Product\Project1\bin C:\Product\Project1\result1.csv
```

### Example 2) When the source folder is the same as the class folder

```
C:\Product\Project1
+- package
  +- Test1.java
  +- Test1.class
```

Command

```
"%PGRJ_HOME%\pgrjava.bat" C:\Product\Project1 C:\Product\Project1 C:\Product\Project1\result2.csv
```

### Example 3) When there are multiple source folders and class folders

```
C:\Product\Project1
+- subproject1
  +- java
    +- src1
      +- package
        +- java_source
    +- src2
      +- package
        +- java_source
  +- bin1
    ...
  +- bin2
    ...
```

Command

```
"%PGRJ_HOME%\pgrjava.bat" C:\Product\Project1\subproject1\java\src1 C:\Product\Project1\subproject1\java\bin1 C:\Product\Project1\result3_1.csv
"%PGRJ_HOME%\pgrjava.bat" C:\Product\Project1\subproject1\java\src2 C:\Product\Project1\subproject1\java\bin2 C:\Product\Project1\result3_2.csv
```



# Chapter 4 Operating Environment Error

The following lists shows the meaning and responsive actions for operating environment errors that are notified by Agile+ Relief J.

Table 4.1 List of Operating Environment Errors

Code	Message Character String	Meaning	Responsive Action
ERR001	Error occurred during check operation.	Exception occurred in Agile+ Relief J.	Please contact the inquiry window of our company.
ERR002	Insufficient memory.	The memory was not enough during the execution of Agile+ Relief J.	<ul style="list-style-type: none"> <li>- Please specify the maximum heap size for code check by using the optional -Xmx, and execute again.</li> <li>- Please reduce the number of sources being targeted for code check, and execute again.</li> </ul>
ERR003	Agile+ Relief J installation environment error.	Installation environment of Agile+ Relief J may be destroyed.	Please re-install Agile+ Relief J.
ERR004	JDK environment setting (JAVA_HOME) error.	<ul style="list-style-type: none"> <li>- JDK is not installed.</li> <li>- The environment variable JAVA_HOME is not set or the setting includes an error.</li> </ul>	<ul style="list-style-type: none"> <li>- Please install JDK.</li> <li>- Please set the environment variable JAVA_HOME.</li> </ul>
ERR100	Failed to input file.	<ul style="list-style-type: none"> <li>- No privilege to read the Java source file.</li> <li>- The specified encoding is different from the encoding of the Java source file.</li> </ul>	<ul style="list-style-type: none"> <li>- Please confirm the existence of the file, access privilege and network, and make the file into readable state.</li> <li>- Please change the encoding of the file to the encoding that is supported by Agile+ Relief J.</li> </ul>
ERR101	Failed to parse the file.	A syntax error is detected in Java source codes.	Please correct the syntax error, and execute code check again.
ERR102	The class library that is necessary for parsing is not found.	Specification of class library is insufficient.	Please specify all the necessary class libraries.
ERR103	The file to be checked is not found.	<ul style="list-style-type: none"> <li>- The specified file does not exist.</li> <li>- The file to be checked under the specified project does not exist.</li> <li>- The specified project folder does not exist.</li> </ul>	Please confirm the existence of the file and the folder.
ERR104	Class file is not found.	Class file is not found.	Please build the Java source code as a targeted for check, and create the class file.
ERR105	Failed to input/output work file.	<ul style="list-style-type: none"> <li>- There is no privilege for reading/writing.</li> <li>- The disk capacity is insufficient.</li> <li>- The network is cut off.</li> </ul>	Please confirm the access privilege of the target folder and disk capacity, and change it into readable/writable state.
ERR106	Class file may not be the latest version. Please build the Java source code as a targeted for check, and create the class file.	Class file may not be the latest version.	Please build the Java source code as a targeted for check, and create the class file.

ERR200	Option specification error.	An inappropriate option has been specified.	Please check the option, and specify a correct one.
ERR201	The file specified with "-af" option is not found.	The specified file does not exist.	Please specify a correct file path.
ERR203	Failed to output the check result.	- There is no privilege for reading/writing. - The disk capacity is insufficient. - The network is cut off.	Please confirm the access privilege of the target folder and disk capacity, and change it into readable/writable state.
ERR204	Encoding specification error.	The specified encoding is not supported.	Please specify the encoding that is supported by Agile+ Relief J.
ERR205	The format of check rule definition file is incorrect.	- code=001 The specified path does not exist. - code=002 A higher version of a check rule definition file or a non-check rule definition file is specified.	Please specify a correct "Check rule definition file".
ERR206	The file specified with "-am" option is not found.	The AndroidManifest.xml does not exist.	Please specify a correct file path.
ERR207	The folder specified with "-ar" option is not found.	The specified folder does not exist.	Please specify a correct folder path.
ERR208	The file specified with "-arf" option is not found.	The specified file does not exist.	Please specify a correct file path.
ERR209	The folder specified with "-arf" option is not found.	The specified folder does not exist.	Please specify a correct folder path.
ERR500	Error occurred during check operation of the cooperation tool.  Please check the error details at console.	By executing FindBugs or PMD, an error occurred.	Please confirm the console and handle the error according to the error message output by the cooperation tool.
ERR501	Failed to start the cooperation tool.	Definition of "plugins.xml" has an error.	Please confirm the definition of "plugins.xml".
ERR502	"plugins.xml" definition error.	Format of "plugins.xml" has an error.	Please confirm the definition of "plugins.xml".
ERR900	Error has been detected in the license management system.  Please handle it according to "Messages and disposal of license management system" of "Installation Guide".		