

Apache Commons CSV

Software Dependability

AY 2023 - 2024

Team members



Francesco Paolo D'Antuono



Ivan Capobianco

Apache Commons CSV

Overview



Parsing and writing CSV files



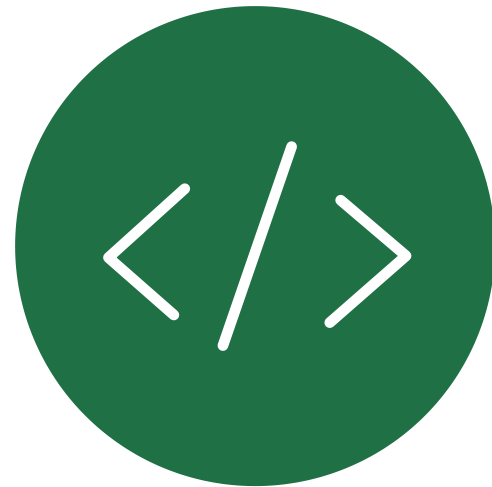
Flexible configuration



Open source

Apache Commons CSV

Size



37300

Lines of code



46

Classes

Apache Commons CSV



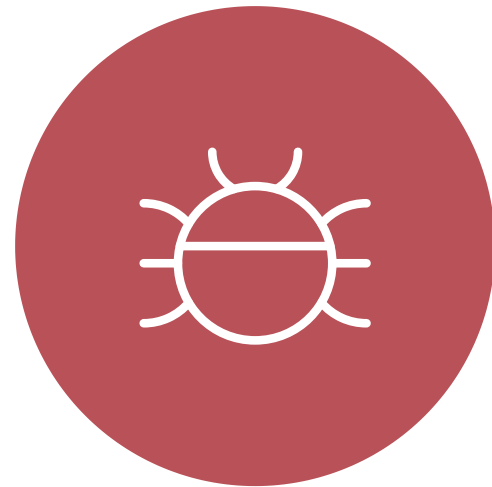
Our repository



Software Quality Analysis

SonarCloud

Analysis results



6

Bugs



658

Code smells

SonarCloud

Bugs



3 critical

- 3x multiple lambda expressions chained together that could raise the same exception

SonarCloud

Fix

Applied refactoring by rewriting the chained expressions making sure that only one method could raise the expected exception.

SonarCloud

Bugs



3 major

- 2x NullPointerException not handled
- return value not used or stored

SonarCloud

Fix

All reported major bugs were identified as false positives.

SonarCloud

Resolved bugs

High  3/3

Medium  3/3

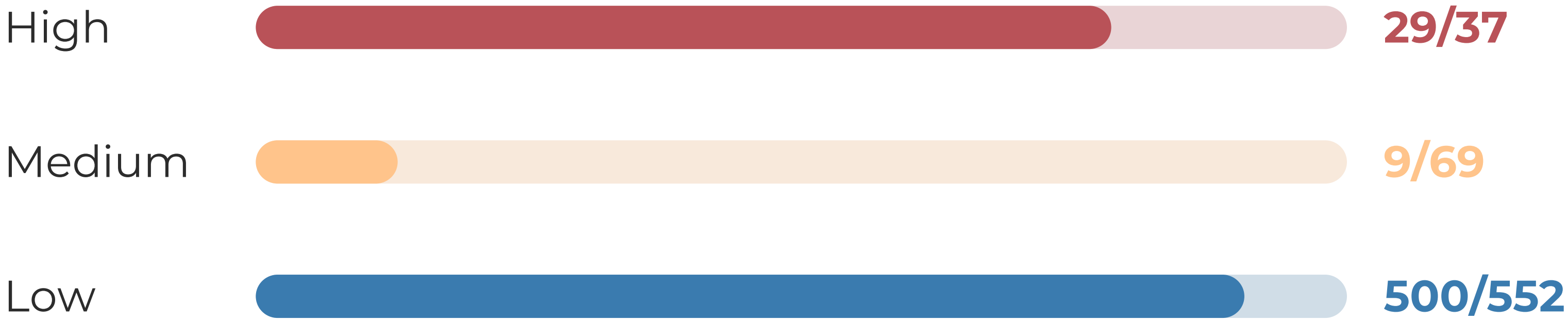
SonarCloud

Code smells

- Test case without assertions
- High cognitive complexity
- Naming scheme for constant values
- Unnecessary public modifiers

SonarCloud

Resolved code smells





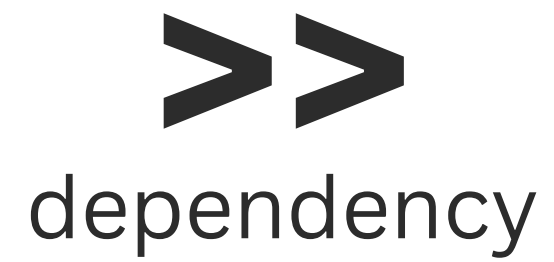
Code Coverage Analysis

JaCoCo

Configuration



Jacoco







































Maven

JaCoCo

Results

 [Apache Commons CSV](#) >  org.apache.commons.csv

org.apache.commons.csv

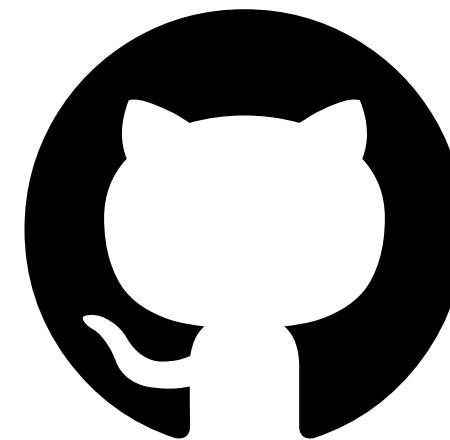
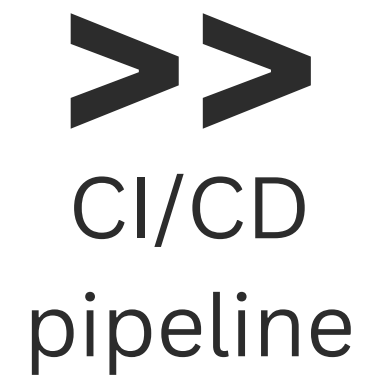
Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
 CSVParser		95%		95%	4	78	2	144	0	31	0	1
 CSVFormat		98%		98%	6	273	5	507	0	112	0	1
 Lexer		98%		97%	4	103	3	172	1	27	0	1
 CSVFormat.Builder		100%		100%	0	43	0	98	0	33	0	1
 CSVPrinter		100%		100%	0	42	0	85	0	18	0	1
 CSVRecord		100%		100%	0	39	0	47	0	23	0	1
 ExtendedBufferedReader		100%		96%	2	39	0	62	0	13	0	1
 CSVFormat.Predefined		100%		n/a	0	3	0	17	0	3	0	1
 CSVParser.CSVRecordIterator		100%		100%	0	11	0	17	0	5	0	1
 Token		100%		n/a	0	3	0	9	0	3	0	1
 QuoteMode		100%		n/a	0	1	0	6	0	1	0	1
 Token.Type		100%		n/a	0	1	0	6	0	1	0	1
 DuplicateHeaderMode		100%		n/a	0	1	0	4	0	1	0	1
 CSVParser.Headers		100%		n/a	0	1	0	4	0	1	0	1
 Constants		100%		n/a	0	1	0	2	0	1	0	1
Total	71 of 5.336	98%	16 of 718	97%	16	639	10	1.180	1	273	0	15

Codecov

Configuration



Codecov



GitHub

Codecov

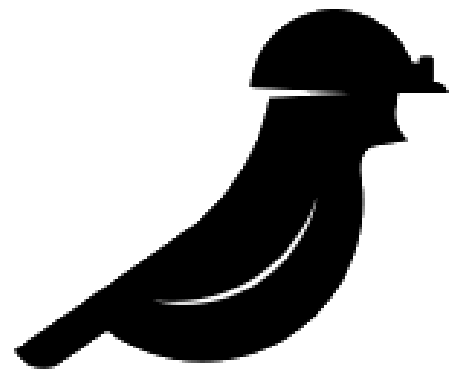
Results

Files	Tracked lines	Covered	Partial	↓ Missed	Coverage %
src/main/java/org/apache/commons/csv/CSVFormat.java	607	597	5	5	<div><div></div></div> 98.35%
src/main/java/org/apache/commons/csv/Lexer.java	169	164	2	3	<div><div></div></div> 97.04%
src/main/java/org/apache/commons/csv/CSVParser.java	158	153	3	2	<div><div></div></div> 96.84%
src/main/java/org/apache/commons/csv/QuoteMode.java	6	6	0	0	<div><div></div></div> 100.00%
src/main/java/org/apache/commons/csv/ExtendedBufferedReader.java	59	57	2	0	<div><div></div></div> 96.61%
src/main/java/org/apache/commons/csv/CSVRecord.java	47	47	0	0	<div><div></div></div> 100.00%
src/main/java/org/apache/commons/csv/Token.java	15	15	0	0	<div><div></div></div> 100.00%
src/main/java/org/apache/commons/csv/DuplicateHeaderMode.java	4	4	0	0	<div><div></div></div> 100.00%
src/main/java/org/apache/commons/csv/CSVPrinter.java	85	85	0	0	<div><div></div></div> 100.00%
src/main/java/org/apache/commons/csv/Constants.java	2	2	0	0	<div><div></div></div> 100.00%

Mutation Testing

PiTest

Configuration



PiTest

>>
test
dependency



Maven

PiTest

Configuration

- All stable mutators used
- Configured to use 8 threads

PiTest

Execution results

Pit Test Coverage Report

Package Summary

org.apache.commons.csv

Number of Classes	Line Coverage	Mutation Coverage	Test Strength
7	99% <div><div>1149/1159</div></div>	95% <div><div>689/723</div></div>	96% <div><div>689/718</div></div>

Breakdown by Class

Name	Line Coverage	Mutation Coverage	Test Strength
CSVFormat.java	99% <div><div>618/623</div></div>	94% <div><div>323/343</div></div>	95% <div><div>323/340</div></div>
CSVParser.java	99% <div><div>159/161</div></div>	100% <div><div>94/94</div></div>	100% <div><div>94/94</div></div>
CSVPrinter.java	100% <div><div>85/85</div></div>	100% <div><div>53/53</div></div>	100% <div><div>53/53</div></div>
CSVRecord.java	100% <div><div>47/47</div></div>	95% <div><div>41/43</div></div>	95% <div><div>41/43</div></div>
ExtendedBufferedReader.java	100% <div><div>62/62</div></div>	93% <div><div>55/59</div></div>	93% <div><div>55/59</div></div>
Lexer.java	98% <div><div>169/172</div></div>	94% <div><div>121/129</div></div>	95% <div><div>121/127</div></div>
Token.java	100% <div><div>9/9</div></div>	100% <div><div>2/2</div></div>	100% <div><div>2/2</div></div>



Performance Testing

Java Microbenchmark Harness (JMH)

Configuration



JMH



benchmark profile
dependency



Maven

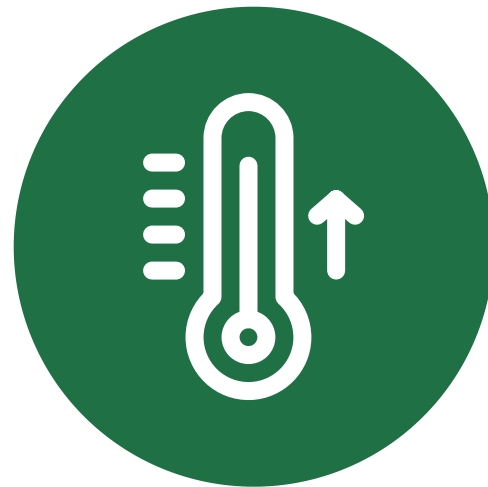
Java Microbenchmark Harness (JMH)

First execution parameters



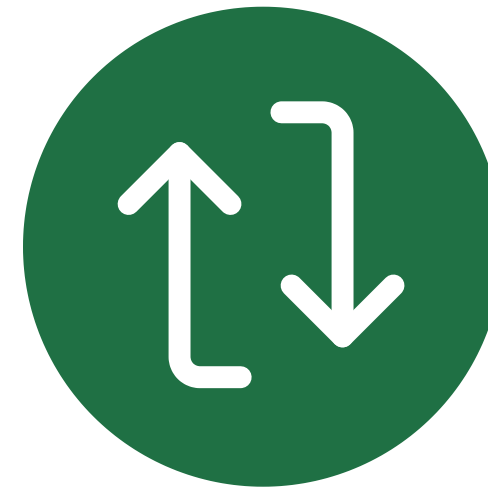
1

Fork



5

Warmups



20

Measurements

Java Microbenchmark Harness (JMH)

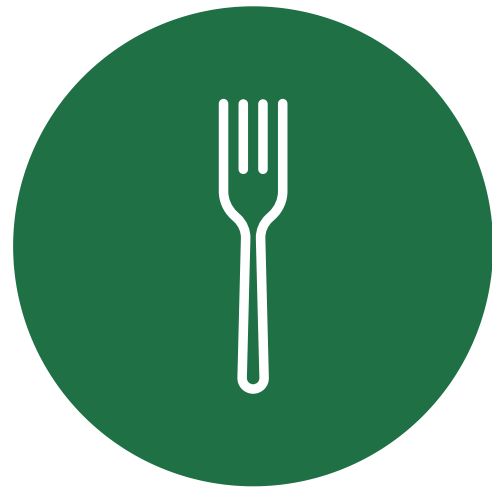
First execution results



Execution time taken not optimal

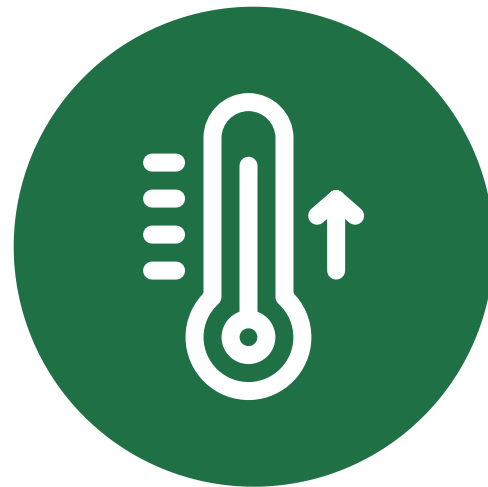
Java Microbenchmark Harness (JMH)

Second execution parameters



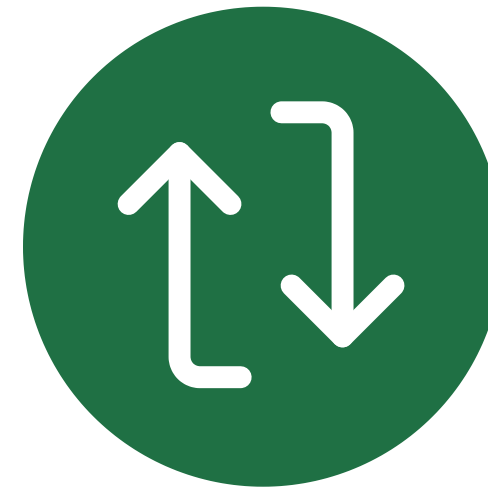
1

Fork



3

Warmups



10

Measurements

Java Microbenchmark Harness (JMH)

Second execution results



Improved execution time



Good performance results

Java Microbenchmark Harness (JMH)

Insights

Benchmark	Avg Time (ms)	Min Time (ms)	Max Time (ms)
parseCommonsCSV	1523.675	1337.792	1765.603
parseGenJavaCSV	1225.577	1124.541	1295.039
parseJavaCSV	1234.754	1073.294	1498.246
parseSkifeCSV	1462.125	1307.927	1628.038
parseSuperCSV	1381.573	1293.049	1526.181
read	157.744	151.640	166.647
scan	1140.819	1012.635	1293.638
split	750.084	688.006	851.519



Automatic Test Case Generation

EvoSuite

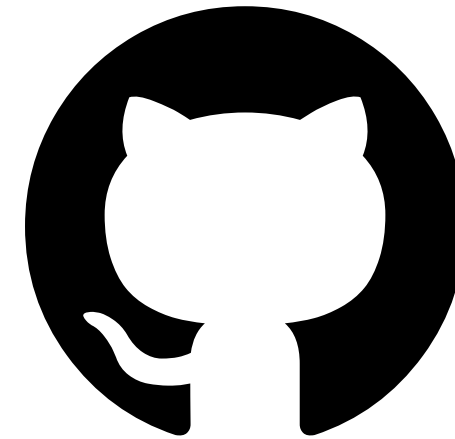
Configuration



EvoSuite



jar files added
manually from



GitHub

EvoSuite

Results



32 test cases successfully created



No bugs identified

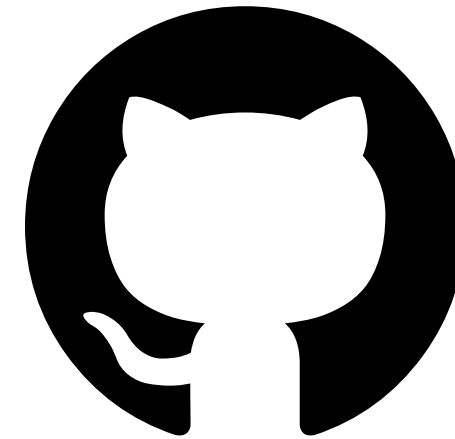
Randoop



Randoop



jar files added
manually from



GitHub

Randooop

Results



108 test cases successfully created



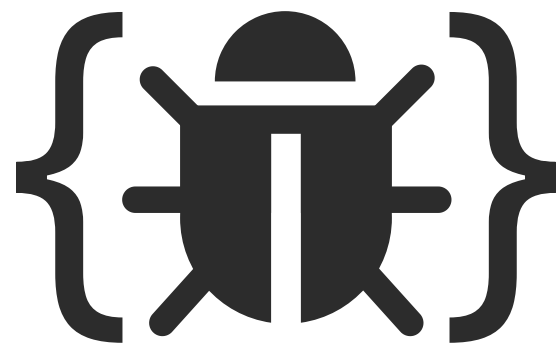
No bugs identified



Software Vulnerabilities

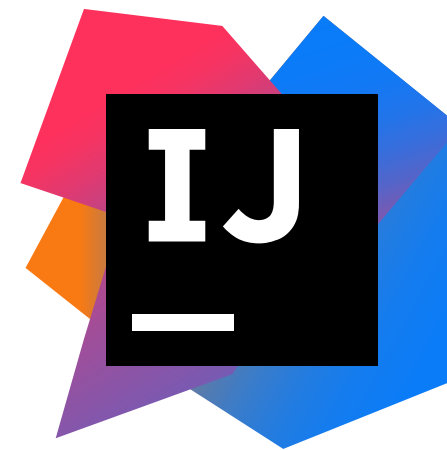
SpotBugs

Configuration



SpotBugs

>>
plugin



IntelliJ IDEA

SpotBugs

Identified vulnerabilities



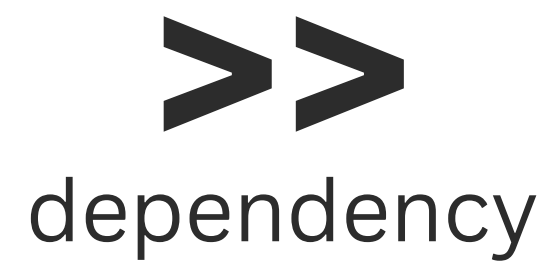
Server-Side Request Forgery (SSRF)

OWASP DC

Configuration



OWASP DC



Maven

OWASP DC

Identified vulnerabilities



No known vulnerabilities reported

Conclusions

Conclusions

- Robust and well-maintained project
- Medium/high code quality
- High code security
- High code performance



**Thank you for
your attention**